DECEMBER 1982 LIDS-R-1267

PROCEEDINGS OF THE

5TH MIT/ONR WORKSHOP ON C^3 SYSTEMS

HELD AT

NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA
AUGUST 23 TO 27, 1982

EDITED BY

MICHAEL ATHANS
ELIZABETH R. DUCOT
ALEXANDER H. LEVIS
ROBERT R. TENNEY

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

SPONSORED BY

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LABORATORY FOR INFORMATION AND DECISION SYSTEMS
CAMBRIDGE, MASSACHUSETTS

WITH SUPPORT FROM

OFFICE OF NAVAL RESEARCH

CONTRACT ONR/N00014-77-C-0532(NR041-519)

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)* 5TH MIT/ONR WORKSHOP ON C^3 SYSTEMS		5. TYPE OF REPORT & PERIOD COVERED INTERIM
		6. PERFORMING ORG. REPORT NUMBER LIDS-R-1267
7. AUTHOR(s) Editors Michael Athans Elizabeth R. Ducot Alexander H. Levis Robert R. Tenney		8. CONTRACT OR GRANT NUMBER(s) ONR-N00014-77-C-0532
9. PERFORMING ORGANIZATION NAME AND ADDRESS Laboratory for Information and Decision Systems Massachusetts Institute of Technology Cambridge, MAssachusetts 02139		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NR 041-519
11. CONTROLLING OFFICE NAME AND ADDRESS Office of Naval Research Arlingtom, Virginia 22217		12. REPORT DATE December 1982
		13. NUMBER OF PAGES xii + 309
14. MONITORING AGENCY NAME & ADDRESS*(if different from Controlling Office)*		15. SECURITY CLASS. *(of this report)* UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT *(of this Report)* Approved for public release: Distribution unlimited		
17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)* COMMAND, CONTROL, AND COMMUNICATIONS		
20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)* This report contains printed manuscripts of papers presented at the Workshop by several authors.		

DD FORM 1 JAN 73 1473

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

FOREWORD

The MIT Laboratory for Information and Decision Systems is pleased to publish the Proceedings of the 5th MIT/ONR Workshop on C^3 Systems. The workshop was held at the Naval Postgraduate School, Monterey, California from August 23 to 27, 1982.

The goal of this annual workshop is to provide an unclassified forum for paper presentation and discussion in the vital field of Command, Control, and Communications (C^3) systems. These workshops were initiated in the summer of 1978 in the hope that intellectual interactions between industrial, government, and academic researchers, with common interests in the theoretical and technological aspects of Command and Control, would provide a much needed acceleration in our basic understanding of the truly interdisciplinary issues and challenging problems encountered in present and future military C^3 systems. To the best of our knowledge, these workshops provide the only open forum for presentation of new concepts, models, and results and an opportunity for critical discussions.

The papers contained in this volume were submitted by the authors following their presentation at the workshop. The authors were encouraged to take into account the comments and discussions pertaining to their paper that took place at the workshop. Unfortunately, several authors did not submit typed manuscripts for inclusion in this Proceedings. Thus, there are differences between the Table of Contents and the Final Workshop Program (which follows the technical papers). A list of the 178 attendees is included at the end of this volume.

There were significant differences between the 5th Workshop and prior ones. Following suggestions from prior year attendees, the workshop was compressed into a single week, thus necessitating for the first time to have two parallel (and competing) sessions during the afternoon and less time for extensive discussion. However, it was felt that this price was worth paying, given the fact that in prior years the great majority of the attendees attended only a single week of the workshop. We plan to follow the single week format in the foreseeable future.

A greater amount of editorial control was exercised during and after the workshop. For the first time, the entire collection of papers can be found in a single volume. This was accomplished by requiring that the manuscripts be typed on special mats, and imposing page charges to discourage excessive length.

From an intellectual point of view the workshop contained a wider interdisciplinary mix of papers. For example, a larger number of papers was devoted to human factors and cognitive psychology issues, reflecting the fact that human decision makers are very much an integral part of C^3 systems! In addition, the number of papers that utilized artificial intelligence concepts was increased. Furthermore, we eliminated all papers that presented subliminal "sales pitches".

This year's workshop was also characterized by a greater degree of intellectual unity. Faithful workshop attendees (about thirty of them) have adopted common methodologies and definitions. There is a greater appreciation of the important C^3 operational problems faced by the military, and many theoretical papers presented during the current workshop indeed reflect a greater awareness by the theoreticians of real C^3 problems. On the other hand, C^3 technologists are becoming increasingly aware of the state of the art in theory and algorithms; there is a clear trend that is developing for seeking more systematic and less ad-hoc approaches to the analysis and design of military C^3 systems. We are still a long way from having a "C^2 science" or "C^3 theory", but the signs are highly encouraging, as can be evidenced by several papers in these proceedings.

I would like to take this opportunity to thank the workshop attendees and authors of the contributed papers for making the workshop a success. I would like to thank my MIT colleagues Prof. Robert R. Tenney, Dr. Alexander H. Levis, and Ms. Elizabeth R. Ducot for their contributions to the planning of the workshop, paper review, and chairing the sessions. Special thanks are due to Dr. Stuart Brodsky (ONR) and Admiral William Meyers (R.Adm., USN, ret.) for organizing the classified ONR sponsored session. I greatfully acknowledge the Naval Postgraduate School for hosting the workshop and Commander Leon Gardner (Lt. Com., USN) in particular for helping with local arrangements. My personal thanks go to Dr. Joel S. Lawson (Naval Electronic Systems Command), also known as the godfather of the C^3 Mafia, for his moral and intellectual support of the workshop since its inception. Last, but certainly not least, I am indebted to Ms. Lisa M. Babine of MIT that took full responsibility for organizing the Workshop and made everything run smoothly according to doctrine.

Michael Athans
Chairman, Workshop Operating Committee
December 1982

TABLE OF CONTENTS

DATA BASES AND DECISIONS

Dr. Joel S. Lawson, Jr.

Naval Electronic Systems Command (NAVELEX 06T), Department of the Navy,
Washington, DC 20363

Abstract. This paper proposes that the popularly used term "data base" should be replaced by "knowledge base" when used in connection with C2 systems. The different kinds of knowledge contained in one are described and related to the decision process. A functional decomposition of command decisions is proposed and an example of the effect of delays on the assessment of the situation is computed, using a methodology described by Akst (1982).

INTRODUCTION

This paper is motivated by a particular difficulty in the development of C3 theory. We have not yet been able to carry out a complete set of calculations which show how a C3 system will, or should behave. This I believe is due to our present inability to represent the decision-making process in a convenient analytic form. The performance of sensors, data processing techniques, and the weapons employed all have simple models, adequate for simple analyses, but we do not have equivalent models for the "decision process."

Only when we have the ability to predict, at least to an order of magnitude, the damage inflicted on the enemy and the losses suffered by the C3 system's forces can we claim to have a theory of C3. Even if we could only do it for a "set piece" battle, we would have a predicted result which could then be tested in a war game or field exercise.

But to do this we must be able to follow analytically the effect of a change in the C3 system's environment from its detection through its processing and assessment to the response it evokes and the results of that response. At the present time we seem to get bogged down when we try to "compute" the decision-making function in the loop.

It is the thesis of this paper that much of this difficulty is due to a lack of precision in our vocabulary when we discuss "data bases" and to having not examined the "decision-making" process in sufficient detail to partition it into convenient and manageable sub-processes.

In the following sections we will first examine briefly some basic models of a C3 system to set the stage and then examine the role of data bases in a C3 system. This examination will set the background for a taxonomy of decision-making and suggest modeling techniques to permit calculations of the through-put and time delays attributable to this function.

C3 SYSTEM MODELS

Probably the most basic model of a C3 system is that shown in Fig. 1. The environment is sensed, the data processed and the resulting "perception" compared with some desired state of the environment, a decision is made about what to do, and the forces directed to take some action. This represents a very much simplified model but it illustrates the main functions which the system must perform.

A more complex model, indicating a superior C3 system's need to deal with possible conflicts between subordinates is shown in Fig. 2. In an analogous model of the three commanders involved, Athans has proposed a model in which each commander is envisioned to have a "mental model" of how the other commanders will behave. This reduces their need to explicitly "coordinate" their activities.

A model similar to Fig. 1 is used by Carol Fox and her group at Johns Hopkins University/Applied Physics Laboratory but breaks down the processing, comparison, and decision functions into elements whose names at least are more in keeping with Navy terminology. This will later form a useful beginning in our search for an improved taxonomy of decision making.

Harmon and Brandenburg of the Naval Ocean Systems Center in San Diego have proposed a very different sort of model, shown in Fig. 3. They view a C3 system as an "information" processing system consisting of four kinds of elements: sensing nodes, action nodes, command nodes, and communications links. The system contains, and deals with two kinds of "information," knowledge stored at command nodes and generated by sensor nodes, and messages which are knowledge in transit from one node to another. This model gives rise to some interesting Measures of Effectiveness (MoEs)to which we shall return later. It also lets us reserve the term "information" for the meaning Shannon ascribed to it, that which reduces uncertainty. (That's why the word "information" was used in quotes above.) And note that the information content of a message depends on the knowledge base of the recipient, not the sender. This has several interesting implications which are left to the reader's imagination.

"DATA BASES" IN C3 SYSTEMS

Two observations are worth noting about data bases in C3 systems. First, it is one of the ironies of the C3 world that any commander you talk to can give you a list of what he wants in his "data base," but, although there is some overlap and isomorphism, no two lists are alike. The elements they ask for range from broad generalities to excruciating detail. Examples, taken from three different lists of "requirements" include such items as: "enemy force disposition and indications of enemy intentions," "the location and capabilities of the opposition," and "Soviet Bloc air units including geographic display of position, type, regimental subordination and climb rate at sea level."

The other interesting observation is that more conventional control systems, such as the thermostat which controls a room's temperature or an aircraft autopilot, don't seem to have any "data bases." And in the computer science sense of the term, they don't have any. In the lexicon of a computer programmer, a data base is "a collection of interrelated data stored together without harmful redundancy...to serve one or more applications in an optimal fashion"(Martin). The term, borrowed from the earliest applications of computers to C2 problems, seems to have been imprecisely extended to cover a great deal more than its original meaning.

The difficulty here is semantic. Most control systems do, in fact, have some kind of memory, such as the dial setting on the thermostat. They also have "rules" which they follow; in the case of the thermostat, turn on the furnace if the temperature falls below the commanded (remembered) level. These rules or algorithms are often "wired in" and we tend to overlook them.

In some of the listings of data base requirements we find similar "rule information," such as "rules of engagement," or "no attack zones." This suggests that what we are really talking about most of the time (in the C3 arena) is a "knowledge base," not a data base. One or more data bases may be part of the knowledge base, but it encompasses more than just data. Harmon and Brandenburg anticipated this in their model and call it a knowledge structure.

In the next section we will consider what such a knowledge base should contain, based on the models we described earlier. And the sorts of "decisions" which it must support.

CONTENTS OF A KNOWLEDGE BASE

Let us start by defining what we mean by the "knowledge base" of a C3 system. Because of the ambiguities of the English language, and the tendency of dictionaries to use definitions which define "knowledge" in terms of "information," and vice-versa, the best we can do is to propose an arbitrary set of definitions and then hope to make them consistent through usage. Therefore we define:

The knowledge base of a C3 system is the entirety of the knowledge, information, or data held in some form in the system's memory, be it maps, books, computer memory, documents, human memory, or switch positions.

Using this definition, it is obvious that such things as the assigned mission, rules of engagement, and tables of assigned frequencies are part of the knowledge base. In fact, everything the system "knows" is explicitly included.

This leads to the question of what kinds of things the system knows. Is there a taxonomy of the system's knowledge which will allow us to decompose it into more manageable classes?

If we consider the thermostat example cited above, we see that there are at least three different kinds of knowledge present even in that simple a device. First, there is a "rule": if it gets cold, turn on the furnace. Second, there is a "condition": the dial setting which defines "cold." And finally there is "data": the present temperature as measured by the position of a bi-metallic strip. Without doing too much damage to conventional English usage, these terms can be adopted as the generalized names of three qualitatively different kinds of knowledge which appear in a C3 system's knowledge base. For reasons which will become apparent later, it will be convenient to have at least two more classes of knowledge. For now, let us call one class "models" which basically are mental shorthand for how things work. They may be the mental models of other team members that Athans proposes or complex sets of

rules which have been reduced to a simple input-output syllogism for convenience.
The other class we will call "narratives" and consign to it everything (such as intelligence estimates) which does not reasonably fit in one of the other classes.

As an aside, it is interesting to note that in the Russian literature on "military cybernetics," a great deal of attention is paid to algorithms (or rules), the part they play in control systems, and the conditions under which they are to be invoked. With the U.S. emphasis on initiative, we often overlook the fact that we, too, have a great many rules and algorithms in our C3 system and concentrate on the "data."

In order to provide a frame of reference for these "classes" of knowledge, let us consider a very stylized decision task. Drawing an analogy with computer programming, we express the decision rule as:

$$\text{If A} \quad \text{Then B} \quad \text{Else C.} \qquad (1)$$

In a simple task, "A" is then the condition, and B and C are pre-planned responses. In a more complex task, "A" might be a more complicated expression, such as:

$$A = (H + J) \cdot (K + L) \qquad (2)$$

where A has been written as a Boolean expression of the elements H, J, K, and L. We will return to this formulation of a decision later and explore some of the possibilities it opens up. For now, it is sufficient to note that H, J, K, and L might very well all be "data elements." For instance, if they were the classes of ships in a group, A might read: "A carrier or a heavy cruiser and a destroyer or a frigate." Note that this decision rule may be either directive, If A then take action B; or inferential, if A then B is probably true.

Thus we have a conceptual framework and vocabulary for our "classes of knowledge." Data refers to physical observations or measurements of the real world, including such things as charts and histories. Conditions are relationships among data elements. Rules are the basis for many of the decisions that the system makes. And Models are mental or other constructs which provide an output for each input without our having to go through a laborious calculation--a form of mental shorthand. (Things which don't fit in these categories we will call simply "narrative," and duck the issue of what they are.)

KINDS OF C3 DECISIONS

With this characterization of the contents of a knowledge base, we can now turn our attention to the decisions which must be made by a C3 system. Using Fig. 1 as our starting point, we see the very first decision is made by the sensing function. It must solve the question of whether or not to declare a target present at some point. A great deal of effort has been expended on how to make this sort of decision, which involves such things as false alarm rates, missed targets, and countermeasures.

Similarly, the processing function must decide if the report is valid and consistent with other data being reported or already known. This set of decisions has also been the subject of much research, e.g., Kalman filters, etc.

And then we begin to enter the realm of cognitive psychology, the region where the human decision maker, the commander, enters the picture. This is the area where our computational capability starts to break down. Let us call these decisions "command decisions" to distinguish them from the "system decisions" such as declaring a target. By this artifice we will separate out the "important military decisions" from the more routine ones which must be made to keep the system running. Note that some of the system decisions may be being made by humans, e.g., radar or sonar operators. And they may require a great deal of experience and judgment, but they do not directly influence the system's response to a stimulus. It is this criterion which we will use to separate "command decisons" from "system decisions," recognizing that the command decision is usually built on the basis of many system decisions, and is generally dependent on their validity for its own.

With these semantic matters out of the way, we can now address the taxonomy of command decisions and how they may depend on and be influenced by the C3 system's knowledge base.

THE COMPONENTS OF A COMMAND DECISION

As mentioned earlier, in Fox's model, the "Compare" and "Decide" functions of Fig. 1 are subdivided into classification, evaluation, planning and decision. These are familiar Navy terms and serve to make her model more understandable by Naval officers. But they still are not sufficiently "process oriented" to provide a basis for understanding how a decision is made. To analyze the process, let us go through an example decision, step by step, and see what takes place. As we do so, we will introduce more precise definitions for some of the terms we will use to discuss this activity of a C3 system

We choose as the boundary or interface between "system decisions" and "command decisions" something which we will call the "perception" of the environment. That is to say, the perception is the best picture of the external physical world that the system can currently form from the data (physical observations) available. Note that at this point, no "meaning" is ascribed to the picture. It is simply a representation of the real world as best as it is known. Obviously the picture may be incomplete, inaccurate, or contain non-existent (decoy) objects. But it is the best the commander has to work with.

The next logical step in a command decision is to ascribe an interpretation to this picture. Although the Navy often refers to this as "evaluation," a better term would be "assessment," which avoids any taint of value judgment. An assessment then is the (present) meaning which the commander ascribes to his perception. In his assessment the commander may ignore objects he believes are decoys or add objects which he believes his sensors overlooked. It is essentially his view of what the world is really like. And we will reserve the term evaluation for a later step in the process.

Based on his assessment, the commander will then form some hypothesis about the future state of the world. That is, what is likely to develop if he does nothing.

This is a ticklish area, and may be a distinction without a difference. But it seems reasonable that there is a difference between the assessment of the present state of the environment and a prediction or hypothesis as to what that portends for the future. For instance, the situation might be correctly perceived and assessed as meaning that an attack was imminent, but the hypothesis that it was coming from the south could be very much mistaken. Similarly, one assessment may lead to two or more hypotheses or several perception and assessment paths may lead to the same hypothesis or predicted future. Therefore, it seems prudent to include in our vocabulary separate terms for these different concepts.

Having perceived the environment, formed an assessment of it, and generated one or more hypotheses about the future, the commander's next step is to develop one or more options about what he might do to "respond" to the situation from which he then has to select one to execute.

In some cases these options may be developed "on the fly" in response to a change in the environment. In others they will have been already developed as contingency plans. It is through this mechanism that the existence of rules and conditions in the system's knowledge base takes account of the "planning" that is done before a military mission is undertaken.

In either case, the available options are then evaluated and one selected for execution. The basis for this evaluation will generally vary from one situation to another. Avoidance of loss and inflicting maximum damage on the enemy are two possible criteria, but by no means the only ones. Another which is often used is to attempt to learn more about the situation, to reduce the uncertainty in either the assessment of the situation or the hypotheses of what it portends. And it is this criterion which often leads to the commander's adopting what might well be called the universal option: "do nothing." But in any event, he has assessed his perception of the world, estimated the likely future(s), examined his possible actions, and chosen one. These are the essential steps of a command decision.

This taxonomy is far different from that of classical decision theory. There is no well-structured problem. The choices are poorly defined and the "utilities" are often situation and personality dependent. And there are possibilities for error at every stage. The perception may be in error or distorted. The assessment may be faulty, the hypotheses may be wrong or the "correct" one may have been overlooked. Options may be overlooked or poorly formulated. And finally, the wrong option can be selected for all the right reasons. It makes one wonder that a commander can ever be successful!

However, this dissection of a command decision has two interesting properties. One is that at each step reasonable men who have agreed on the previous steps may disagree on the next one. Second, the sources of error or "noise" in each stage are qualitatively different, and tend to have different sorts of effects on the outcome.

This model of the decision process is clearly an extension of the SHOR (Stimulus-Hypothesis-Option-Response) paradigm proposed by Wohl, and by an analogy might be called the PAHOES (Perception-Assessment-Hypothesis-Option-Evaluation-Selection) paradigm. Because of its finer structure, I believe it can provide a better basis for model building than does the SHOR paradigm. In addition, it will allow us to use some of the MoEs proposed by Harmon and Brandenburg as diagnostic tools within the framework of Athans' model.

A MODEL OF DECISION MAKERS

A useful model (which deals with only one elementary decision) has been proposed by Boettcher. This model is represented graphically in Fig. 4 and consists of a two-stage process. In the first stage, the input variable X is transformed by some hypothesis to an intermediate variable Z. Then Z is processed by an algorithm into the output decision Y. Crudely one might think of the first stage as generating an understanding of the input, X, and the second stage as processing this understanding into an action or choice.

Using this model, Boettcher has defined the total activity of the decision process as:

$$G = G_C + G_N + G_B + G_T \qquad (3)$$

Here G_T is the throughput, that is the activity which is reflected in the outside world in the form of the variable Y. G_B is the "blockage," that fraction of the activity which results in no output. In a physical system, this could be interpreted as the dissipative losses. G_N is the noise-like activity which can presumably lead to a randomization of the output, either by distorting the intermediate variable Z or leading to selection of a less than optimum algorithm. And finally, G_C represents the coordination which goes on internal to the decision process.

By using an information theoretic approach, Boettcher provides a method of calculating numerical values for the components of this activity. In particular, for binary decisions, the number of bits required for each component can be calculated directly

Thus Boettcher's model seems to have several useful properties for carrying out computations about command decisions. It provides for blockage or rejection of information, it has a dissipative term in its internal coordination, and permits the introduction of rules, conditions, data, and noise into the decision process. Furthermore, it is an elementary model, which can be chained together in serial fashion to deal with first assessments, then hypothesis forming, followed by option generation, etc.

By adding to this basic model a bypass around the decision maker, L(X), which is guaranteed to make the right choice, we could measure the effect on the outcome of changes in the contents of the knowledge base available to the commander. This suggests that there are an interesting series of experiments which could be done in some war gaming facility. Additionally, Boettcher and Levis have extended this model to two and three man teams of decision makers, so the sort of situations portrayed in Fig. 2 and 3 can now be accommodated.

MEASURES OF PERFORMANCE SUGGESTED BY THESE MODELS

Granted that these models have some validity in that they represent in a gross way the various processes which are invoked in the making of a command decision, how can they be employed?

First and foremost, they could be used to predict the behavior of a command control system, if we can make some reasonable estimates of some of the quantities which enter into them. In particular, the time required for each step in the PAHOES model is probably estimable for at least some simple scenarios. And Athans' model of the expert team of experts gives us a mechanism for describing how a team coordinates its activities, which allows us to estimate the internal coordination required in Levis and Boettcher's extended model.

Secondly, they should suggest MoPs and MoEs that could be used to compare the performance of different C3 system arrangements. For instance, Harmon and Brandenburg propose as a measure of C3 system performance the difference in the knowledge bases at the various command nodes. If we think of each of our team of decision makers as a separate command node, then we see that some part of their knowledge base is shared, but as Athans has pointed out, one DM's mental model of another is not as detailed as his model of himself. This knowledge difference (ΔK) may or may not give rise to inferior system performance, depending on how seriously the lack of detail affects the behavior of the team, either by conflicting actions on the part of the DMs or by requiring excessive coordination and communications.

Another other cause of ΔK is the transit time of messages within the system. For example, one node might be charged with generating an overall assessment of the situation, based on the perceptions provided by separate perceivers of the air, surface, and subsurface worlds. If it takes very long to form its assessment, the result may be sufficiently distorted to lead to the formulation of seriously flawed hypothesis.

George Akst of the Center for Naval Analyses has proposed an interesting model of this effect. In examining the impact of automation on the sensing, processing and perception forming stages of the Marine Corps C3 system, he hypothesized that the present assessment of the battlefield was related to the previous assessment in the following way.

If there were N objects (e.g., tanks) detected on the battlefield at time $(t-t_d)$ and the probability of detection of the surveillance system is P_d, then the number of objects which will be assessed to be present at time t is given by:

$$\tilde{N}(t) = P_d N(t-t_d)+(1-P_d)\tilde{N}(t-1) \qquad (5)$$

where t_d is the system delay and $\tilde{N}(t-1)$ is the number which were assessed to be there at the last step. The derivation of this equation involves some simple substituting in Bayes rule and uses a discrete "time step" model of the sensing process to represent a useful approximation of the effects of delay. Fig. 5, taken from

Akst's paper is a comparison of a possible "real world" with the assessment of it presented to the commander.

Obviously, a commander working with such a delayed assessment might very well make less than optimum choices.

An interesting property of ΔK as an MoP is that it is something which is in principle measurable. One can imagine equipping a team of observers with cameras and recording devices and positioning them in each command node. If they took synchronized pictures of the geographic plots and status boards in the various command centers throughout an exercise or wargame, it would then be possible to determine roughly how different the knowledge bases were. And a few pointed questions asked every 15 minutes or half-hour could illuminate the question of how alike the assessments and hypotheses were.

Time delays between and through nodes are, of course, also measurable by inserting synthetic data or messages at various points in the system. As suggested by Akst's model, a useful gross measure of some of these delays might be the relative delay between real world and assessed events.

CONCLUSIONS

The existence of some relatively simple models of decision making, and in particular distributed team decision making which allow for the introduction of rules and conditions as well as "data" suggests that by adopting the concept of a "knowledge base" as herein proposed, we may be able to compute something about command decisions. In fact, Akst's work provides a concrete example.

Furthermore, the PAHOES paradigm provides a means of examining the effect of different kinds of errors or deficiencies in the knowledge base on a command decision. And it suggests several possible experiments which could be undertaken in war games or exercises.

Finally, the fact that the information content of a message depends on the knowledge base of the recipient suggests that an important class of messages might be those which describe to the others the present knowledge base of each team member.

In summary, it appears that we should be able to break the computational bottleneck about the "decision process" and attain at least a crude theory of C2 in the near future.

REFERENCES

Akst, G., Klein, S., and Simmons, D. (1982). Evaluating Tactical C2 Systems--A Three-Tiered Approach. 49th MORS Symposium, Albuquerque, NM June 1982

Athans, M. (1981). The Expert Team of Experts Approach to C2 Organizations. Proceedings of 4th MIT/ONR Workshop on C3 Systems, San Diego, California, Jun 1981.

Boettcher, Kevin. (1981). An Information Theoretic Model of Decision Making. Masters Thesis, MIT Cambridge, MA.

Harmon, S. Y., and Brandenberg, R. L. (1981). Concepts for Description and Evaluation of Military C3 Systems. Proceedings of the Control and Decision Conference, San Diego, California, Dec 1981.

Levis, A., and Boettcher, K. (1982). On Modeling Teams of Interacting Decision Makers with Bounded Rationality. Proceedings of the IFAC/IFIP/IEA Conference on Analysis, Design and Evaluation of Man Machine Systems. Baden-Baden, Federal Republic of Germany, Sep 1982

Martin, James. (1975). Computer Data Base Organization. Prentice-Hall, Inc., Englewood Cliffs, NJ.

Wohl, J. (1979). Battle Management Decisions. MITRE Corp. Rpt. M79-233, Bedford, MA, Dec 1979.

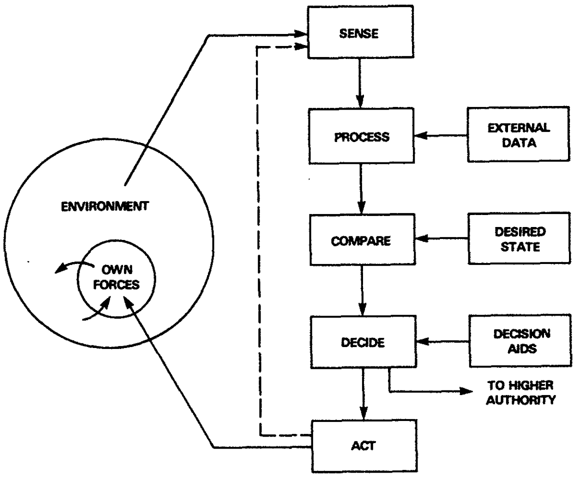


FIG 1. BASIC MODEL OF C2 PROCESS

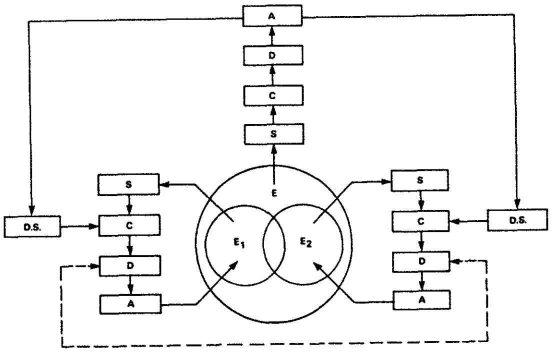


FIG 2. HIERARCHY OF C2 PROCESSES

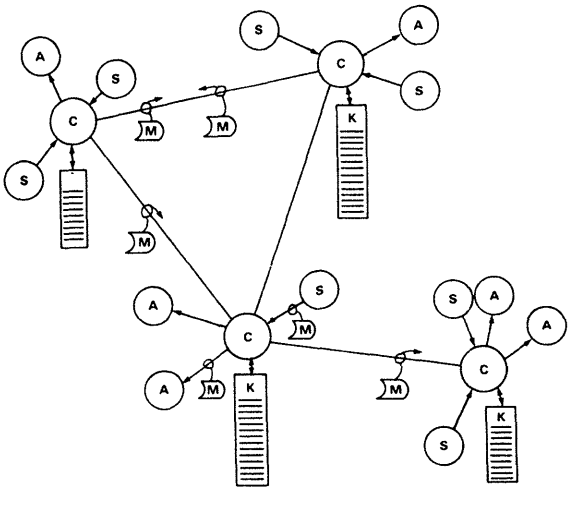


FIG 3. HARMON-BRANDENBURG MODEL OF C2

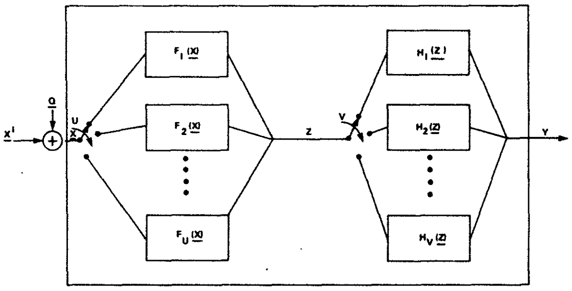


FIG 4. MODEL OF A DECISION MAKER AFTER BOETTCHER

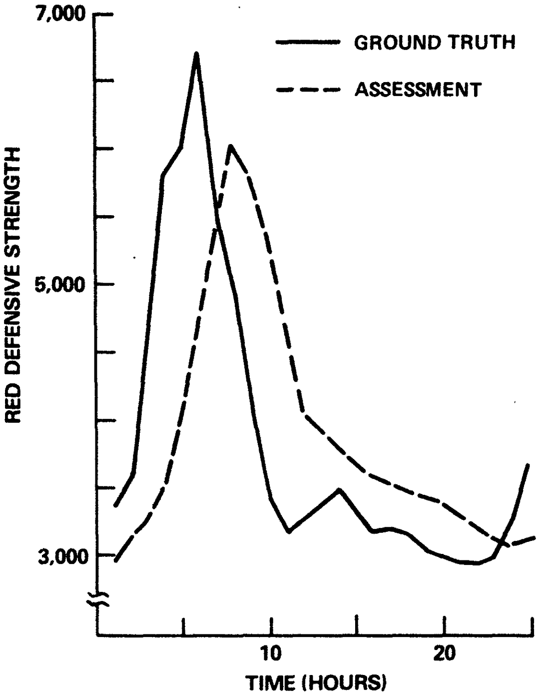


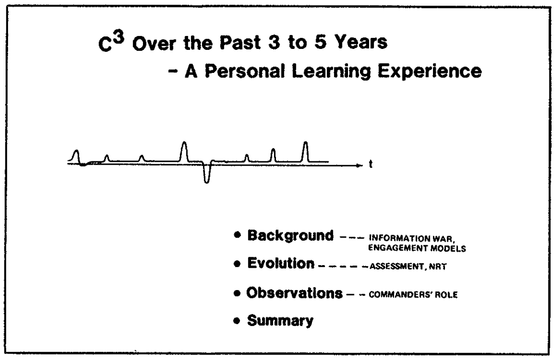
FIG 5. EFFECT OF DELAYED ASSESSMENT (AFTER AKST)

C^3 OVER THE PAST 3 TO 5 YEARS —A PERSONAL LEARNING EXPERIENCE

T. P. Rona

Boeing Aerospace Company, P.O. Box 3999, Seattle, Washington 98124

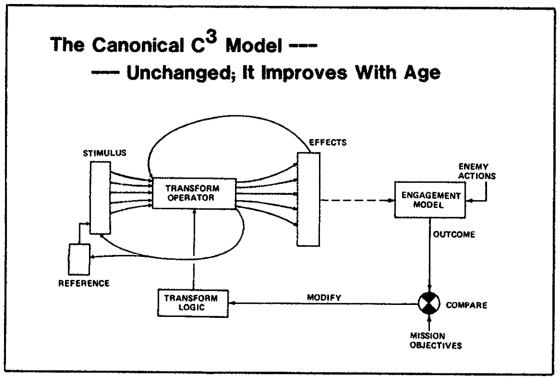
Abstract. Five or so years ago, the focus of my attention was on the vulnerability of C^3 systems, exposed as they are to the interference and exploitation by the enemy, according to the (increasingly accepted) concepts of the information war. Since then, in addition to many self-inflicted difficulties associated with C^3 system design and evaluation, the importance of the non-real-time (NRT) message channels is being recognized. Our perception of the commander's role may be modified accordingly.



C^3 Over the Past 3 to 5 Years
- A Personal Learning Experience

- **Background** --- INFORMATION WAR, ENGAGEMENT MODELS
- **Evolution** ------ASSESSMENT, NRT
- **Observations** -- COMMANDERS' ROLE
- **Summary**

Fig. 1



The Canonical C^3 Model —
— Unchanged; It Improves With Age

Fig. 2

This talk is spontaneous; it has been prepared while I was listening to my fellow-speakers. I hope that by explaining the nature of my own learning process as we go along, several messages will intrigue the audience to the point of becoming accepted. Learning, in general, is a sporadic and haphazard accumulation of knowledge; it is even more so in this instance (Fig. 1). I often get an inspiration (positive peaks), then almost immediately my good friends and colleagues proceed to demolish the same; I end up in a negative state of utter confusion and dismay...then slowly, ever so slowly, some consensus emerges and the integrated effect is gradually trending toward the positive. In what follows, I will try to summarize the background and explain the reasons for, and the nature of, the evolution in my thinking. Based on a few observations of the talks presented earlier here, I make bold to proceed to some conclusions.

This canonical model (Fig. 2) is actually 5 years old; it was first produced for a slim document, "Conceptual Framework for Military C^3 Assessment" prepared for Mr. A. W. Marshall from OSD Net Assessment.[1] My knowledgeable friends convinced me that this is a cybernetic model; if such designation makes it more respectable, so be it. Whatever its theoretical virtues, it does adequately reflect the emphasis I would like to convey: The hardware-intensive elements (those which are usually scrutinized at considerable length, and rightly so, in view of their voracious consumption of budgets and engineering talent), the communication links, the sensors, and the associated platforms are barely apparent. On the other hand, the logical functions are explicit: Any C^3 operation must have a set of <u>stimuli</u> which are apt to modify the state of the <u>system</u>; these are conveyed to a decision process, functionally equivalent to a <u>transform operator</u> (XFO) which, in some mysterious way, derives a set of <u>effectors</u>, i.e., messages that cause modifications in the military forces or operations managed by the C^3 system. A few important details: The stimulus set must be measured with respect to some <u>reference</u> set (calibration, or previous "state"); the <u>transform</u> operator (XFO) must act in conformance with some

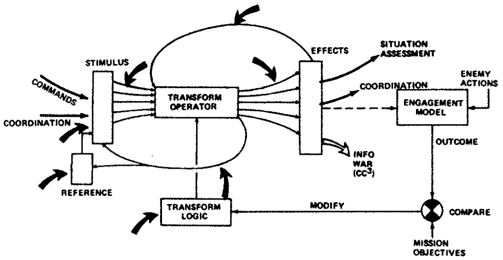
---

[1] Available to participants of the ONR/MIT Workshops upon request to the Boeing Aerospace Company, Attention: Mrs. D. P. Linder, Mail Stop 84-56, P.O. Box 3999, Seattle, Washington, 98124.

previously established transform logic; the effector set is reconveyed to the transform operator as a stimulus in the form of feedback (what has happened as a consequence of my actions?). One of the subtle but important effectors is that which selects, controls, updates the stimulus and the reference sets. Finally, and lest we forget it, a mechanism must be provided, no matter how rudimentary or embryonic, that will compare the results of an engagement model (using the effectors generated by the C^3 system in the context of some hypothetical enemy actions) to those required by the set of mission objectives. If this comparison is unsatisfactory (given that the combat and combat support resources are available), then the transform logic must be iteratively modified until the desired result is achieved. In more simple terms, the "commander" must in some way learn the "tricks of the trade"; the decisions must be conducive to the desired results, if not "optimally," at least with a reasonable probability of success.

All this is old material; not a single concept is new. It has been debated ad nauseam in recent years. It is thus quite surprising that some of the essential details are still lost from sight in many of the discussions presented here. Let me quickly review these (Fig. 2A):

The Canonical C^3 Model —
— Unchanged; It Improves With Age



C^3 is not only a support function –
it is a combatant

Fig. 2A

- The stimulus set comprises command messages from higher authority, as well as coordination messages from parallel commands and status messages from subordinates.
- Conversely, the effector set comprises situation assessment and/or status messages to higher authority, as well as coordination messages to other units at the same hierarchical level.
- The effector set explicitly covers counter-C^3 (CC^3) messages aimed at degrading the enemy command and control operations; conversely, the stimulus set should pay particular attention to information obtained from the exploitation of the enemy C^3 messages (e.g., by means of ELINT, COMINT, SIGINT, and other arcane-sounding techniques).
- Both stimulus and effector sets must deal with the well-being of the C^3 system itself: Its status and integrity must be verified at all times, but more particularly following overt or covert attacks by the enemy. Actions must be on hand to repair, reconstitute, verify, etc.
- Finally, the enemy may attack, again overtly or covertly, by means of sabotage or electronic warfare, just about any point of the C^3 system. THE MAIN PURPOSE AND MERIT OF THE CANONICAL C^3 MODEL ARE TO PORTRAY EXPLICITLY THE MANY POSSIBLE ENTRY POINTS FOR ENEMY ACTION. Among these, the stimuli and the communication links are well-known and obvious, but the enemy can also spoof our feedback links, can deceive our sensor control and reference setting links. In the same vein, the transform logic can be misled ("misimprinted"), causing the commander to follow a logic not appropriate to the real situation.

It is important to note that many of these enemy actions may take place long before the actual overt hostilities. Unless we ceaselessly watch for such actions, they may wreak havoc with our C^3 performance without our being even aware of them.

To sum up: We should stop looking at C^3 as just another combat support system--it is a true combatant on its own right; it can attack, and it must be protected against enemy attacks.

Rona's Axiom

- "Wars are fought against the enemy"

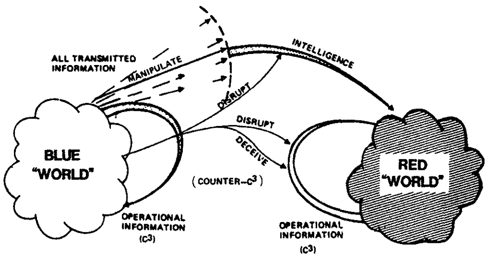
Corollary

- The enemy is the most important element in the C^3 design and operation
  - Contributes essential stimuli
  - Attempts to destroy, degrade and to **exploit** our C^3
- He/she should appear explicitly on our charts

Fig. 3

The axiom in Fig. 3 is just to reemphasize the preceding points. There is no purpose in understanding the exquisite refinements of C^3 theory if the malicious ever-presence of the enemy is being ignored or misconstrued. Whenever a C^3-related logic diagram is shown, I urge you to give at least some recognition to enemy being explicitly present, both in the attack mode (trying to degrade or to destroy our own C^3 operation) and in the exploitation mode (trying to understand and use to advantage the information gathered from our C^3).

The Information War Concept



— essential aspect of warfare

Fig. 4

Figure 4 is, again, a good old musty chart (A.D. 1975), but quite to the point. Blue wants to disrupt or at least deceive (spoof) the Red C^3 message traffic; it would also like to disrupt or deceive the Red intelligence channels. It turns out to be more effective to mix, in the latter case, the true and the deceptive messages, resulting in manipulation of the Red intelligence. The C^3 systems play an essential role in waging this type of offensive warfare. Protection of our own C^3 against similar attempts by the enemy is no less essential.

Evolution

- **Before (Ca. 1977)**

  The enemy is the cause of most of the C^3 problems

- **After (1982 +)**

  The enemy may help, but we mostly do it to ourselves

  Deficiencies in
  - System and operational concept document
  - Functional analysis
  - Equipment, software, procurement
  - Congruence C^3 ⟷ **Military Structure**
  - System level trades
    - Human performance vs. automation
    - Reversionary modes, tests

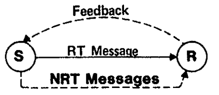
  Explicit consideration of NRT message flow

Fig. 5

In the original perception, I assumed that we are competent in defining, designing, and deploying the C^3 system, so I placed the emphasis on the role of the enemy and on the various aspects of vulnerability of the C^3 systems to enemy action. The converse, i.e., the opportunity for attacking the enemy C^3, was also repeatedly emphasized.

It is now quite apparent that many of our C^3-related difficulties are of our own doing. The enemy may help, but his role is just another contribution. In almost all the efforts aimed at assessing the "value" or "effectiveness" of a C^3 system, we encounter deficiencies in the documentation of system and operational concepts. Functional analysis often stops at the essential, e.g., what are the decision rules at the "fusion" and situation assessment levels? The most important question, i.e., what is the logic underlying the "commander's" decisions, is often left unanswered. Case studies, quite naturally, are exceedingly difficult to conduct; victorious generals are reluctant to admit the role of their own luck or of the blunders of their adversaries; defeated generals do not, as a rule, carry the weight of authority. Also conspicuously missing are the trade studies that would back up some of the most essential system-level design decisions, e.g., the levels of desirable automation vs. obtainable human performance, the cost and feasibility of reversionary modes planned and incorporated into the C^3 system, the level and organization of testing, etc. When relatively new mission aspects are being considered (e.g., military space operations), often the military structure lags behind the assigned mission responsibility, thus the congruence between C^3 and military organization is often (much) less than adequate. The so-called non-real-time[2] (NRT) information channels are by now recognized as a vital part of C^3 system design.

Nature and Impacts of NRT



- **NRT** Present in all communication channels—
  - Include planned (structured) portion
    e.g. indoctrination, instruction, training, code-books, keys, ---- programs, settings
  - But also unplanned (unstructured) portions
    e.g. psycho/physiological interpretations, pattern recognition, conceptual associations, emotions, stress reactions, allegiances
- **Performance and vulnerability of NRT must be explicitly entered in C^3 design**

Fig. 6

If a source S wants to communicate with a receptor R, the real-time message must be preceded by a non-real-time message that explains to the receptor the rules of interpretation of the message (Fig. 6). This principle is very general; it applies to communications between humans, between humans and machines (and vice-versa), and, of course, between machines, robots, computers, etc. The time domain is also very large; such non-real-time "precommunications" can take place a few minutes or even seconds before the real-time message (e.g., a permissive arming code sent a few seconds before an actual launch command). They encompass mission briefings, training programs, basic training, language skills, all the way to the cultural and even genetic inheritance of (human) communicators. The comprehension of speech by a young infant is built up bit by bit through the hundreds of thousands of learning episodes reinforced by the reward/penalty "pay-offs" of the socialization process.

---

[2] Perhaps, according to some real-time advice, the expression "pre-real-time" may be more desirable and descriptive.

The NRT channels operate at all levels of the C^3 hierarchy; they include a planned (structured) portion, such as indoctrination, instruction, training, code books, crypto keys for humans; programs, instructions, threshold or frequency settings for machines. The unplanned (unstructured) portion may assume paramount importance in the NRT traffic; it can lead to distortions, confusion and delays, but it can also offer unexpected resources of resilience, self-repair, and initiative. The unstructured portion is characteristic of humans[3]; it comprises the psycho-physiological interpretations and reactions, the pattern recognition and conceptual associations, emotions, stress reactions---even the loyalty and allegiance-related motivations.

The consequences for the C^3 system design and performance are as important as they are (or should be) obvious:

o The existence of NRT message traffic interacts with the requirements and performance associated with the real-time messages. Thus, competent preplanning (typically the SIOP, "canned" instructions, etc.,) materially reduces the need for wide-band emergency communications; the personal familiarity of the subordinates with the manners and voice patterns of the commander decreases the need for time-consuming authentication procedures, etc. Conversely, the absence of competent NRT messages places additional, and often unnecessary, burdens on the real-time "action links."

o Conceptually, the NRT links are just as vulnerable to disruption and deception as are the real-time links. The enemy, knowing this, may attack the NRT channels or may attempt to "exploit" them in the sense discussed earlier. The survivability, the vulnerability to countermeasures, and the security against hostile exploitation of the NRT message structure must be examined, just as are these same parameters for the real-time domain communications.

It is theoretically interesting to describe the commander as a senior authority figure, endowed with suprahuman insight and wisdom, pondering the situation and making the decisive moves that will eventually lead to victory. In reality, the nature of military command operations is quite different. Figure 7 illustrates one half (the "stimulus world") of the canonical model described above.[4] The "raw," undigested stimuli never (well, hardly ever) reach the top commander directly; if they do, they are invariably momentous and catastrophic in their consequences. In normal operation, structured messages, as well as sensory or intelligence inputs reach subordinates (humans or machines) whose responsibility is to verify, filter, interpret, synthesize and/or otherwise "process" the raw data. These subordinates are exactly that, subordinate decision makers responding to some stimuli, transforming them according to a more-or-less stringently prescribed logic into coalesced "messages" (effectors) for the use of their next superior echelons or into coordination messages for the use of their peers. Note that this holds true for the humble noncommissioned officer watching his radar screen; it is just as true for the interpreter participating in the interrogation of some prisoner; it remains true for the most sophisticated "situation analysts" at higher headquarters; and it is true for the Commanding Officer's chief of staff who (at least theoretically) is responsible for summarizing the battle situation, or even the progress of a theater-wide campaign, and for advising about the likely outcomes of alternative actions. Every single one among these decision nodes, including the top box labeled "XFO" (transform operator), CAN BE FULLY REPRESENTED BY A MINIATURE VERSION OF THE CANONICAL MODEL, complete with stimuli, transform operator, transform logic, references, and effectors. Communication can be real short (if the operation is contained within one facility) or real extended and complex (if remote distances are involved). Whenever humans are present, the NRT inputs, both structured and unstructured, invariably introduce their undesired, as well as their highly desirable, characteristics. This is true for the senior commander in the system, as well as for all of his subordinates.

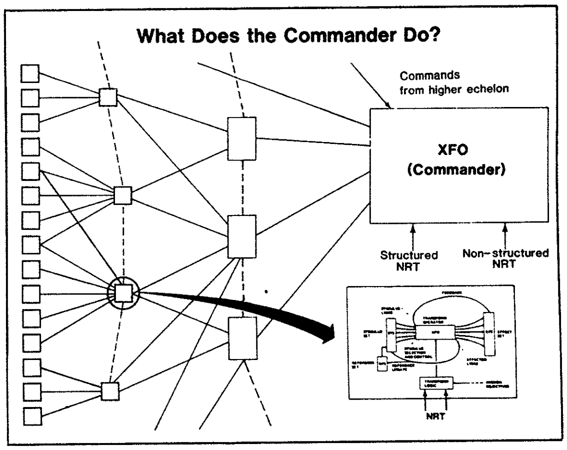


Fig. 7

This brings us to one of the new perceptions (it should have been obvious all along to expert C^3 designers): (Fig. 8) The commanders must, even and especially under combat conditions, spend a substantial portion of their time to continually ensure the integrity of the C^3 system that was originally intended to help in the management of the available combat resources.

---

[3] As the complexity of electronic information handling machinery increases, the possibility of machines developing idiosyncrasies and psychoses should not be underestimated. If so, the NRT inputs to and by machines may become relevant, but let us hope that such possibility is, for the time being, beyond our time horizon of concern.

[4] Similar remarks would apply to the "effector world," the other half of the canonical model.

... But What Does He/She Really Do?

- Creates, understands, approves S—XFO—E structure
- Assigns people (directly or indirectly)
  Select, fit, train, exercise, calibrate, replace
- Exercise, calibrate system (people, organization, procedures, equipment)
- Protects, repairs, reconstitutes
- Grow, adapt . . . . start over again

Conclusions: – NRT message world is key . . .
—"Normative" characteristics excessively rare

– Decision aids must evolve from lower strata

Buzzer-Dial-Screen-Map-Videogames-Human

Fig. 8

The commander must (ideally) create, understand, and approve (at least tacitly) each element and the connections between the elementary decision nodes. People must be assigned (directly or through intermediaries), the commander must select, fit, train, exercise, calibrate...if necessary, replace, even in the heat of the battle, unsatisfactory or unavailable personnel (Fig. 8). The C^3 system, including people, organization, procedures, equipment, must be continuously exercised and calibrated, preferably under realistic conditions, while preserving its basic security---surely not a routine or easy task! In case of battle damage or following compromise of security features, the commander must see to the protection of the remaining elements, then must proceed to the repair or even the reconstitution of the essential C^3 structure. Even in peacetime, the commander must accommodate the relentless onslaught of new technology; the C^3 system must be able to grow, to accommodate changes without loss of operational capability---not even for a few hours.

The conclusions are by now self-evident: As C^3 systems grow in complexity and technical sophistication, they become less transparent to the top commander. To discharge his responsibility and to preserve his own sense of confidence, he must spend a nontrivial fraction of his time to verify and to calibrate the "response" of the system to various combinations of stimuli, including emergencies. The presence of the NRT message world makes such calibrations highly dependent on the individual humans involved---including the top commander himself. To the extent that artificial decision aids are being conceived, these should first address the lower levels; the job of the top commander is, in my view, not yet ready for machine-aided decisions.

Note at the bottom of the Fig.: One can imagine a whole sequence of decision aids of increasing complexity, addressing increasingly higher-level decision nodes. A simple buzzer can alert a sentry to a breach of security; dials establish the quantitative nature of critical stimuli; maps and screens can be used to depict increasingly complex "orders of battle"---it is easy to imagine symbolism derived from modern videogames to portray the progress of battle at higher levels of abstraction. I happen to be partial to the view that for a long time to come the top commander will prefer to use the "matched filter" of the well-calibrated brains of his own trusted and carefully nurtured immediate staff.

Summary

A. Please introduce the enemy in the full C^3 picture

(Beyond the stimuli and survivability)

B. Design (minimal) engagement simulators ⟶ test, checkout, empirical growth of "XFO"s

C. Rediscover and introduce **explicitly** the non-real-time message flow

(Trade against real time message traffic requirements)

Fig. 9

In summary (Fig. 9):

A. Let us hereby resolve that henceforth the role of the enemy will be at least suggested or mentioned whenever the C^3 system performance or design requirements are being discussed.

B. The understanding of logical operations at the intermediate hierarchical level leaves much to be desired. These are not as philosophically complex as those of the top command nodes, neither are they as simple or even trivial as some of the signal preprocessors. A typical example may be the tactical "fusion center," perhaps cued by intelligence inputs. Designing engagement simulators to exercise such decision nodes, preferably without alerting the personnel involved, may go a long way to instill confidence in the performance of the whole system.

C. Extend onto the NRT message flow the same painstaking scrutiny that is now being lavished on the quasi-real-time operation of the C^3 systems. Inquire about the gains achievable in the perfomance levels and cost reduction of the real-time channels if an increasing proportion of the communication and decision burden can be shifted to the NRT domain.

D. When contemplating the development of "concept-oriented" decision aids (cf. Dr. J. S. Lawson's presentation), start by the intermediate levels (middle level and senior staffs, rather than the top commander and his personal staff). With the rapid advance of computer-assisted "video games," the technology for structuring multisensory representations of relatively complex engagements may not be far beyond the currently envisioned state of the art.

CONCEPTS AND THOUGHTS CONCERNING CONTROL STRATEGY FOR CONDUCTING INFORMATION WARFARE

DANIEL SCHUTZER
CHIEF OF NAVAL OPERATIONS
DEPARTMENT OF THE NAVY
WASHINGTON, D.C.

ABSTRACT

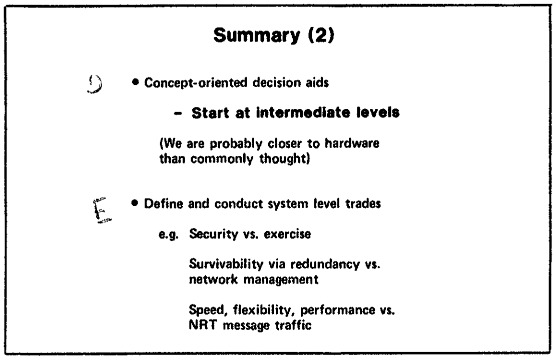
This paper introduces a methodology from which the various functions of communications, surveillance and reconnaissance, intelligence, cover and deception, electronic warfare, communications security, operations security, command and control can be managed under a single unified framework. This includes a control strategy that can be used to manage the above information manipulative functions from a total information warfare viewpoint. This strategy involves attempting to maximize a measure of the combined states of own forces knowledge and opponent forces ignorance through application of these information manipulative functions. Finally several principles are introduced and discussed in relation to this information warfare control strategy.

INTRODUCTION

In the last decade, we have experienced some trends that, in the authors opinion, may serve to dramatically alter naval warfare in rather fundamental ways. It is believed that to realize the full benefit of these trends and changes, the various information-manipulative functions of communications, surveillance and reconnaissance, intelligence, cover and deception, electronic warfare, communications and operations security, and command and control need to be viewed and managed from a single unified framework.

Today we are on the threshold of major new innovations and trends which will result in an order of magnitude increase in both the tempo (rate of change or action) and range lethality (action-at-a-distance) normally associated with naval warfare. Satellites which are capable of communicating sensed intelligence, literally, at the speed of light, hover the sky. C2 platforms in space, and weapons that can be launched from space, are becoming technically viable. Missiles and torpedo technologies have progressed to the point where unmanned vehicles can be launched at intercontinental ranges and supersonic speeds. They can fly ballistic profiles; they can skim the oceans surface to avoid detection; and they can execute pop-up terminal maneuvers. They possess sophisticated on-board sensors and processors providing a significant degree of autonomy. Sophisticated RPV's that extend a platforms sensor range against targets "over-the-horizon" are also becoming technically viable.

The capability of our major combatants for self-protection can also be expected to increase dramatically. All trends indicate significant growth in the size, armour, and defensive capabilities of our major combatants. These ships will literally become floating fortresses. They already possess an arsenal of sophisticated defensive warning radars, ESM, missiles, EW, E-O countermeasures and off-board decoys to defend themselves



Summary (2)

D • Concept-oriented decision aids

- Start at intermediate levels

(We are probably closer to hardware than commonly thought)

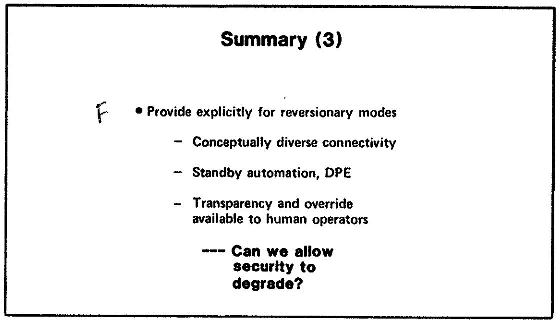
E • Define and conduct system level trades

e.g. Security vs. exercise

Survivability via redundancy vs. network management

Speed, flexibility, performance vs. NRT message traffic

Fig. 10



Summary (3)

F • Provide explicitly for reversionary modes

- Conceptually diverse connectivity
- Standby automation, DPE
- Transparency and override available to human operators

--- Can we allow security to degrade?

Fig. 11

E. System level trades must be defined and studied well before committing to the development of increasingly complex C^3 systems. A few examples are:

- o System security vs. exercise and proficiency levels
- o Survivability (obtained by means of additional redundancy) vs. increasing complexity in network management
- o Reliability, flexibility, and execution delays vs. NRT message traffic.

F. The topic of reversionary operating modes was just barely touched upon. But I want to emphasize that highly automated systems are prone to catastrophic breakdowns when they are needed the most. C^3 systems must have conceptually diverse connectivities (several types of comlinks, not susceptible to simultaneous failure due to the same cause); they must have a high degree of redundancy in their data processing equipment (DPE) including data bases; they must be reasonably transparent to, and subject to override by, human operators. Whether or not security can be allowed to lapse (to some limited extent) under emergency conditions is one of the trickiest questions to be answered by the C^3 system designer.

against incoming attack. In the future we can expect still greater capability in these areas of defense as well as the employment of such new technologies as laser and directed-energy particle beams. Future enemy raids will have to traverse layers of defensive zones or barriers before they can reach their target. Clearly the offense must be prepared to make the most of the element of surprise and concealment if his attack is to succeed. Even so, the offense must be prepared to suffer unprecedented high attrition rates.

When one lives in a environment of continual world-wide surveillance, achievement of the element of surprise takes on a new dimension. Techniques for the concealment from, and the coordinated deception of, the enemys surveillance systems and warning indicators need to be developed, perfected, and exercised.

All of these considerations highlight the main need for the Navy to significantly improve its ability to assess, to evaluate, and to coordinate, manage and control its forces and to possess timely and accurate data concerning the enemy forces capabilities, characteristics, modus operandi and intentions.

This requires organizing and integrating the various information and signal collection, dissemination, management and control disciplines and processing techniques into a cooperative team of "expert" modules or assistants that can work together effectively to accomplish the objectives of "putting metal on the target", orchestrating a cover and deception strategy, and of evaluating a situation and making sound decisions under tight time constraints. This necessitates a unified framework from which to manage and coordinate the various information-manipulative functions. This paper proposes one such framework.

FRAMEWORK

Consider the various information-manipulative functions of command, control and communications, surveillance, reconnaissance and intelligence, cover and deception and electonic warfare, communications security and operations security. Each of these functions have a role in direct support of war fighting. Some of them have the function of maximizing the information available to own forces, whereas some of them have the function of minimizing the information available to hostile forces.

For the cases of information exchange and active (or non-cooperative) search it is speculated that the resulting increase in the state of knowledge should increase in direct proportion to the current lack of knowledge, or ignorance, concerning the current situation. The more there is to know, the more there is to gain by actively searching, collecting and exchanging information about the situation amongst the various concerned and cooperating parties. This relationship is expressed by the following equation:

Case 1 - Exchange of Information and Active Search

$$\dot{u} = -au \qquad (1)$$

Where

u = entropy, or ignorance

$\dot{u} = \frac{du}{dt}$ = change in entropy

a = positive constant

Information may also be collected and exchanged by means which exploits such enemy actions and activities as their communications, radar emissions or patterns or behavior, and/or which is dependent for its success on the enemys ignorance concerning the nature, location, means, or method of the collector and receiver. For these cases the increase gained in knowledge seems to be directly related to both our current knowledge and to the enemies current ignorance of the situation. The more knowledgeable we are, and the more ignorant the enemy is, of the current situation the more effective we will be in our covert collection attempts.

This relationship is expressed by the following equation:

Case 2 - Covert and Cooperative Collection

$$\dot{u} = -b(1-u)v \qquad (2)$$

Where

u = entropy of friendly
v = entropy of enemy
$\dot{u}$ = change in entropy of friendly
b = positive constant

Cover and deception and electronic warfare are techniques for interfering with and inhibiting the enemies information collection and exchange activities with the goal of deceiving, confusing, and degrading the enemies knowledge of the situation. For these cases the success we achieve in degrading the enemies knowledge seems to directly related to both our own and the enemies current knowledge of the situation. The more we know, the more effectively we can manipulate and inhibit the enemies attempts at information collection and exchange. The more the enemy knows, the more he has to lose through our use of cover and deception and electronic warfare.

This relationship is expressed by the following equation:

Case 3 - Active jamming and deception

$$\dot{v} = c(1-u)(1-v) \qquad (3)$$

Where u = entropy or ignorance of friendly forces,

v = entropy or ignorance of enemy forces,

$\dot{v}$ = change in entropy of enemy forces

c = positive constant.

Communications security and operations security measures serve to make the enemys job of covert intelligence collection more difficult by denying him key sources of data. This can be reflected by an appropriate adjustment of the constant coefficient, b of equation (2) downward. The net result is that b becomes a smaller number thus requiring a greater exertion of effort on the collectors part to increase his knowledge of the situation.

Although these various information-manipulative functions all have the common objective of increasing the friendly forces knowledge of the situation while degrading and spoofing the enemy with respect to his knowledge of the situation, when viewed in isolation from each other they could well be employed at cross-purposes and end-up substantially negating each other.

For example, exercising emissions control prohibits the employment of active search; and jamming an enemies communications prevents covert collection and exploitation of these communications.

For purposes of this paper it is assumed that the respective designs of the systems that support these various information-manipulative functions is coordinated and that the systems would be capable of working together whenever it makes sense to do so. For example, it is assumed that friendly forces has designed its EW and communications systems so that employment of the former will not jam the latter unnecessarily.

It should be noted that even for the case when no information-manipulative activity is occurring, the information states do not remain static, and without an active effort at collecting and exchanging information concerning the evolving situation, ones state of knowledge will degrade and ones ignorance will grow in direct proportion to ones past state of knowledge.

This relationship is expressed by the following equations:

Case 4 - Normal growth in uncertainty

$$\dot{u} = d(1-u) \; (4)$$

Where u = entropy or ignorance,

$\dot{u}$ = change in entropy,

d = positive constant.

When viewed in this manner, the strategy illustrated in figure 1 emerges as a reasonable information control scheme. The objective of this control strategy is to maximize a measure of the combined state of knowledge of own forces and state of ignorance of opponent forces through judicious application of the information manipulative functions previously enumerated. Friendly sensor measurements, intercepts of enemy communications and sensor radiations, and human intelligence reports are received, combined, and interpreted to form two pictures; one representing our own state of knowledge of the situation, and the other representing our best estimate of the opponents state of knowledge. These pictures include what units are located where, their course, and likely destination and intentions along with indications of the confidence assocsiated with these estimates.

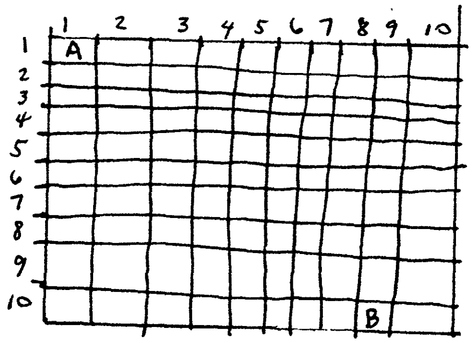
ESTIMATION OF THE STATES OF KNOWLEDGE AND IGNORANCE

The estimate of the state of ignorance would be proportional to the sum of uncertainties associated with a units identity, location, course of destination. Misinformation would correspond to a case where ones uncertainty is low, but where an error bias has been introduced causing a serious misidentification or mislocation. This will result in a larger entropy than for the state of total ignorance.

As an illustration, consider Figure 2 below. In this example there are ten location cells and two ships (course, and destination type information have been omitted for simplicity). Each location cell has three terms; namely:

$$q_j(A)\ln p_j(A/A) - q_j(B)\ln p_j(B/B) - q_j(\bar{A},\bar{B})\ln p_j(\bar{A},\bar{B}/\bar{A},\bar{B}) \quad (5)$$

FIGURE 2
ESTIMATION OF ENTROPY



Where $q_j(A)$ = probability that ship A is in location cell j;

$q_j(B)$ = probability that ship B is in location cell j;

$q_j(\bar{A},\bar{B})$ = probability that neither ship B or ship A is in location cell j;

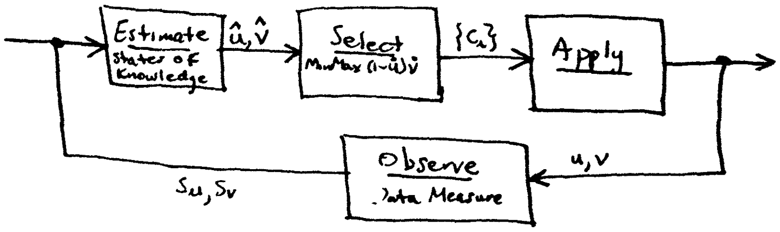
A and B are mutually exclusive (ships A and B cannot be in same location cell at the same time);

$p_j(A/A)$ = probability that, given ship A is at location cell j, ship A is detected;

$p_j(B/B)$ = probability that, given ship B is at location cell j, ship B is detected;

$p_j(\bar{A},\bar{B}/\bar{A},\bar{B})$ = probability that, given neither ship A or B is at location cell j, no ship is detected;

Figure 1 - Information Warfare Control Strategy



If as shown in Figure 2, ship A is actually at location cell (1,1), or Row 1 and column 9; then all the q terms are zero with the exception of

$$q_{(1,1)}(A)=1,\ q_{(10,9)}(B)=1$$

and $q_{(n,m)}(\bar{A},\bar{B})=1$
for all n and m except for (n=1, m=1) and (n=10, m=9).

Therefore, for example (2), the entropy reduces to the sum of 100 terms; namely

$$-\ln p_{(1,1)}(A/A)-\ln p_{(10,9)}(B/B)$$
$$-\sum_{\substack{n,m\neq\\ 1,1 \text{ and } 10,9}}\ln p_{(n,m)}(\bar{A},\bar{B}/\bar{A},\bar{B})$$

For the case of total knowledge,

$$p_{(1,1)}(A/A)=p_{(10,9)}(B/B)=\underset{\substack{n,m\neq\\ 1,1 \text{ and } 10,9}}{p_{(n,m)}}(\bar{A},\bar{B}/\bar{A},\bar{B})=1$$

and entropy = 0

For the case of total ignorance,

$$p_{(n,m)}(A/A)=p_{(n,m)}(B/B)=p_{(n,m)}(\bar{A},\bar{B}/\bar{A},\bar{B})=\frac{1}{3}$$

and entropy $= 100\ln 3$

For the case of complete misinformation,

$p_{(1,1)}(A/A)=p_{(10,9)}(B/B)=0$, as well as selected $p_{(n,m)}(\bar{A},\bar{B}/\bar{A},\bar{B})$ terms, and entropy $=\infty$.

INFORMATION WARFARE CONTROL STRATEGY

Information warfare is a two sided contest. It is assumed that both parties will seek to select that combination of information manipulative functions which serves their best interests. Accordingly, friendly should select that set of information manipulative functions which results in a maximum objective function assuming that the opponent will always select a corresponding set of information manipulative functions designed to minimize the same objective function. This is a worst case approach. It might prove desirable to provide the decision maker with other possible control strategies and their possible outcomes; such as one based upon a best case analysis and another based upon a most likely (derived from historical precedence) enemy response.

Of course the end objective is to prevail in battle, maximizing the enemies losses and minimizing ones own losses. It will be shown in a later section that the amount of knowledge, or information, a combatant possesses directly influences his combat effectiveness. Accordingly, a suggested objective function is $(1-\dot{u})\dot{v}$, the product of own forces increase in his "state of knowledge" and the opponents increase in his "state of ignorance". This objective function increases when friendlies knowledge of the situation is increased and/or the opponents ignorance of the situation is increased. It is recommended that prior to the selection for execution of any set of information manipulative functions, a projection of battle outcome be determined which includes the likelihood of enemy initiation of conflict and the predicted losses and associated risks. This battle outcome prediction should be presented along with recommended information manipulation control strategies so that a course of action may be determined which includes consideration of the initiation of conflict as well as pure information warfare actions.

CONTROL STRATEGY EXAMPLES

The control strategy just enumerated is illustrated by two examples.

First Example: Suppose one wished to choose between two candidate information manipulative control actions (active surveillance and covert collection) in response to an opponent who is currently practicing covert collection. The problem is formulated as a choice between the following two cases:

CASE 1: Friendly, active surveillance. Enemy, covert collection.

Friendly entropy $\dot{u}=d(1-u)-au$

Enemy entropy $\dot{v}=d(1-v)-b(1-v)u$

CASE 1: Objective function =
$$[(1-d)+(d+a)u](d-bu)(1-v)$$

CASE 2: Friendly, covert collection.
Enemy, covert collection

$$\dot{u} = d(1-u) - b(1-u)v$$
$$\dot{v} = d(1-v) - b(1-v)u$$

CASE 2 Objective functions =

$$[(1-d) + bv + du - buv](d-bu)(1-v)$$

Then case 1, active surveillance, should be selected over case 2, covert collection, when

$$\frac{a}{b}\frac{u}{1-u} \geq v, \text{ or } \frac{a}{b}\frac{u}{v} \geq (1-u)$$

Active surveillance should be chosen when our own state of knowledge is less than an adjusted ratio of own ignorance to opponent ignorance.

SECOND EXAMPLE:

A second example concerns the selection of covert collection versus friendly jamming. This problem is formulated below:

Case 1: Friendly, covert collection. Enemy, jams.

$$\dot{u} = d(1-u) - b(1-u)v + c(1-u)(1-v)$$
$$\dot{v} = d(1-v)$$

Case 1 $= [1 - d(1-u) + b(1-u) - c(1-u)(1-v)]\, d(1-v)$

Case 2: Friendly, jams. Enemy, does nothing.

$$\dot{u} = d(1-u)$$
$$\dot{v} = d(1-v) + c(1-u)(1-v)$$

Case 2 $= [1 - d(1-u)][d + c(1-u)](1-v)$

Then case 1 should be selected when

$$u \leq \left(\frac{b}{c} - \frac{1}{d}\right) + v$$

Friendly should jam rather than perform covert collection when his ignorance exceeds an adjusted measure of the opponents ignorance.

KNOWLEDGE AND COMBAT EFFECTIVENESS

The amount of knowledge, or information, possessed by a combatant influences his combat effectiveness. This influence can be represented as an adjustment of parameters in the differential equations that govern the combat situation which, in turn, determines the sensitivity of the engagement outcome to the number of units involved. These equations, are discussed in great depth in the Lanchester-combat-theory literature, are summarized in Figure 3 below, where x = number of friendly units and y = number of enemy units:

LANCHESTER-COMBAT AND INFORMATION WARFARE DUALITY

Comparison of the differential equations in Figure 3 with the information control differential equations (1) through (4) formulated earlier reveal striking similarities in form. Accordingly, many of the solutions and mathematical techniques developed as part of Lanchester-combat-theory are directly applicable to the solution of problems in information warfare control theory.

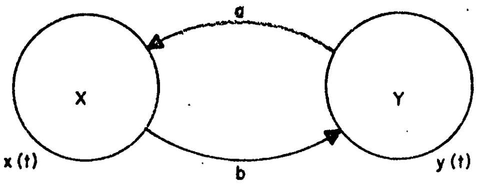
The Helmbold-type combat equations below, represents a general combat model which contains many of the classic homogeneous-force combat models as a special case.

$$\dot{x} = -a(t)\left(\frac{x}{y}\right)^{1-w_y} y \quad \text{with } x(0) = x_0$$

$$\dot{y} = -b(t)\left(\frac{y}{x}\right)^{1-w_x} x \quad \text{with } y(0) = y_0 \tag{7}$$

COUPLING BETWEEN INFORMATION MEASURES AND COMBAT THEORY EQUATIONS

The degree to which a combatants performance behaves more like an aimed fire engagement or more like an area fire engagement is a function of the knowledge possessed concerning the combat situation. The more complete the information a combatant possesses with respect to the combat situation, the more closely that combatants performance will follow the laws of aimed fire. In this case the combatants performance approaches a value proportional to the square of the number of units in the engagement, whereas, in the case of area fire, the performance is proportional to the number of units. In short, the relative effectiveness of the units



ATTRITION PROCESS	DIFFERENTIAL EQUATIONS	STATE EQUATION
Aimed Fire vs Aimed Fire	$\frac{dx}{dt} = -ay$ $\frac{dy}{dt} = -bx$	LANCHESTER (1914) $b(x_0^2 - x^2) = a(y_0^2 - y^2)$ SQUARE LAW
Area Fire vs Area Fire	$\frac{dx}{dt} = -axy$ $\frac{dy}{dt} = -bxy$	LANCHESTER (1914) $b(x_0 - x) = a(y_0 - y)$ LINEAR LAW
Aimed Fire vs Area Fire	$\frac{dx}{dt} = -ay$ $\frac{dy}{dt} = -bxy$	BRACKNEY (1959) $\frac{b}{2}(x_0^2 - x^2) = a(y_0 - y)$ MIXED LAW
Operational Losses vs Operational Losses *	$\frac{dx}{dt} = -ax$ $\frac{dy}{dt} = -by$	PETERSON (1953) $b \ln \frac{x_0}{x} = a \ln \frac{y_0}{y}$ LOGARITHMIC LAW
Aimed Fire plus Operational Losses *	$\frac{dx}{dt} = -ay - \beta x$ $\frac{dy}{dt} = -bx - ay$	MORSE and KIMBALL (1951) (GENERALLY VERY COMPLICATED)

Figure 3 VARIOUS FUNCTIONAL FORMS FOR ATTRITION RATES THAT HAVE BEEN CONSIDERED IN THE LANCHESTER-COMBAT-THEORY LITERATURE.

employed by two opposing forces in an engagement situation is directly related to their relative state of knowledge, or information, governing the combat situation. They can be expressed by means of the Helmbold equation, equation where it is postulated that perfect knowledge would correspond to aimed fire, $W_y = 1$ for friendly and $W_x = 1$ for the enemy. Total ignorance would correspond to area fire, or $W_y = \frac{1}{2}$ for friendly and $W_x = \frac{1}{2}$ for the enemy. Misinformation would correspond to worse performance than area fire, or to the cases $W_y < \frac{1}{2}$ for friendly and $W_x < \frac{1}{2}$ for the enemy. Total misinformation could be represented by $W_y = 0$ for the friendly, and $W_x = 0$ for the enemy. Under total misinformation, a combatant fires on himself and equation 7 reduces to:

$$\begin{aligned} \dot{x} &= -ax \\ \dot{y} &= -by \end{aligned} \qquad (8)$$

In this case the more units a combatant possesses the larger is the combatants losses; the combatants attrition rate being proportional to the number of units he possesses.

The Helmbold-type combat represented by equations 5 and 6, can be slightly modified by adding an additional term for "operational" losses; i.e., losses not due to enemy action (e.g. losses due to sickness, accidents, or fraticide due to misinformation). If we add terms for such losses, then equations 7 and 8 becomes

$$\begin{aligned} \dot{x} &= -a(t)\left(\frac{x}{y}\right)^{1-w_y} y - \beta(t)\, x \\ \dot{y} &= -b(t)\left(\frac{y}{x}\right)^{1-w_x} x - \alpha(t)\, y \end{aligned} \qquad (9)$$

These relationships can be depicted graphically in Figures 4, 5. Here the parameters d and e (as opposed to W_x and W_y) represent the coupling of the combatants state of information to the combat equations; e.g. d=1 corresponds to total knowledge, d= 1/2 to complete ignorance, and d=0 to total misinformation.

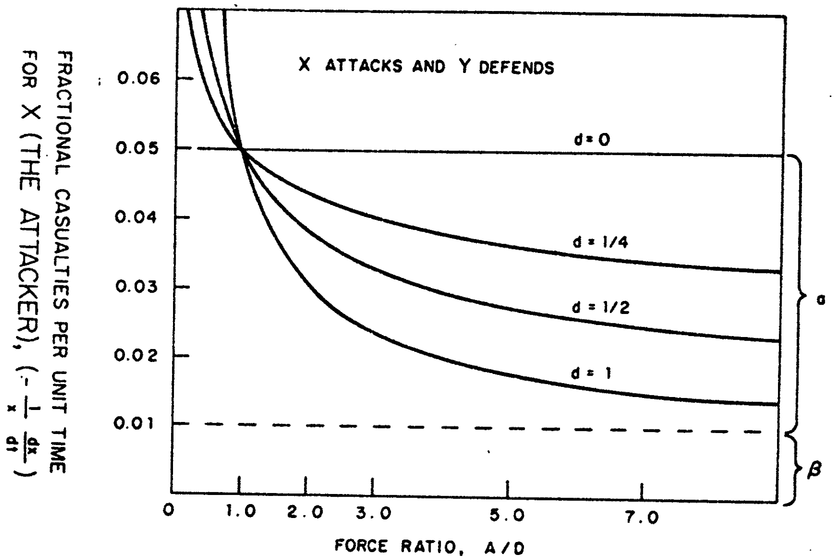


Figure 4
RELATION BETWEEN X's FRACTIONAL CASUALTY RATE AND THE FORCE RATIO FOR THE MODEL $\frac{dx}{dt} = -a\cdot\left(\frac{x}{y}\right)^{1-d}\cdot y - \beta x$ WHEN X ATTACKS.

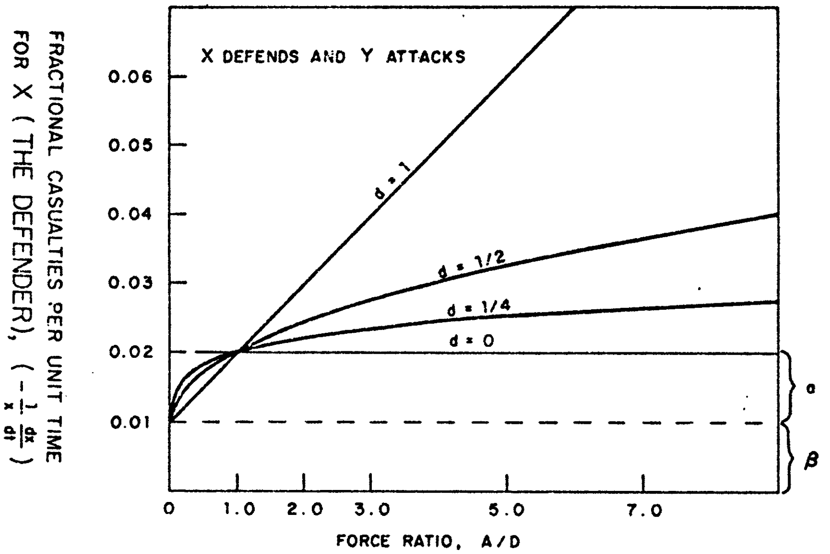


Figure 5
RELATION BETWEEN X's FRACTIONAL CASUALTY RATE AND THE FORCE RATIO FOR THE MODEL $\frac{dx}{dt} = -a\cdot\left(\frac{x}{y}\right)^{1-d}\cdot y - \beta x$ WHEN X DEFENDS.

PRINCIPLES OF INFORMATION WARFARE CONTROL

By examining the differential equations which govern information warfare and combat theory and their postualted coupling, several principles emerge. They are summarized below:

There is a threshold effect with respect to combat information. Below a given threshold, added information has little effect on combat outcome and above a critical threshold added information is superfluous. There are time windows within which an information advantage is critical and outside of which the same information advantage becomes irrelevant. Since it is not possible to maintain an information advantage indefinitely, it is necessary to possess a good understanding of the battle dynamics and to key the timing of the information warfare measures taken to the battle dynamics. All information items are not equal in value. The value of a particular item of information is context and situational-dependent. These principles have intuitive appeal. They are illustrated below with some simple examples.

THRESHOLD EFFECT

Consider a battle situation where at the onset of conflict friendly and enemy both posses an equal number (25 units) of equivalent type units. The value of information in this example is assumed to be reflected in the number of friendly units that can engage in aimed fire rather than area fire. The effect of increasing friendlies information advantage, under these assumptions, is illustrated below.

FIGURE 8 - THRESHOLD EFFECT

FRIENDLY Aimed/Area	ENEMY	REMAINING ASSETS Friendly/Enemy
0/25	25	0/0
1/24	25	0/0
2/23		2/0
3/22		8/0
4/21		12/0
5/20		20/0
6/19		20/0
⋮	↓	↓
25/25	25	20/0

It can be seen from Figure 8 that once the information advantage reaches a level where at least 4 units are engaged in aimed fire, further increases in the information advantage has no effect on battle outcome; it is time to strike. Similarly there is a lower level threshold effect in that increasing the information advantage from no units to one unit engaged in aimed fire has no effect on battle outcome. The information advantage has to increase to the point where two or more units are engaged in aimed fire before there is an effect on battle outcome.

TIME WINDOWS

To illustrate that there are time windows or points in a battle when an information advantage is more critical than at other times or points in the battle sequence, consider the following simple examples. In the first example, both sides are assumed to have 25 units each. The attack is either initiated when friendly possesses a high information advantage (23 units in aimed fire mode and 2 units in the area fire mode) or when friendly has much lower information advantage (2 units in an aimed fire mode and 23 units in an area fire mode). The timing of the attack relative to the information advantage is critical to outcome of the battle. If the battle is initiated when in the high information advantage mode, friendly will completely destroy the enemy with 12 of his own units remaining (almost half his force). For the low information advantage case, friendly will completely destroy the enemy, but only two of his units will remain; a bitter price to pay for victory.

The second example, makes use of Figures 5 and 6 which relate fractional casualty rate to force ratio for the Helmbold-type combat modified to take "operational" losses into account. These curves are computed with typical values assumed for the attrition rate coefficients a and b. The parameter d, if you recall the previous discussion, is related to the combatants informations state; where d=1 represents complete knowledge, d=1/2 represents total ignorance and d=0 represents complete misinformation. Consider, the following two stage battle, each stage of equal duration, ten time units. Initially the force ratio is 50 attacking units to 25 defending units, or A/D = 2. For this example two cases are studied. In the first case the attacker is given an information advantage over the defender (d-1 for attacker, d=1/2 for defender) in the first stage of the battle, and no information advantage (d=1/2 for both attacker and defender) during the second stage of the battle. For this case it can be shown from Figures 5 and 6 that the two stage battle results in a force ratio after the entire battle of 1.32 (with A/D = 1.77) resulting as an intermediate force ratio after the first stage).

In the second case, there is no information advantage during the first stage of the battle but the attacker achieves an information advantage during the second phase of the battle. For this case it can be shown that the battle results in a force ratio of A/D = 1.2 (with A/D = 1.63 resulting as the intermediate force ratio after the first stage).

It is seen for this example that there is an distinct preference for starting with an information advantage at the beginning of the battle rather than achieving this advantage during the course (in this case at the middle) of the battle.

DYNAMIC NATURE

The differential equations, 1 through 4, introduced in this paper model information warfare as a time-varying process. For example, equation 4 reflects that fact that if no information manipulating functions are actively applied, the information state will not remain static but will degrade exponetially with time. This situation is compounded by the introduction of uncontrollable enemy information warfare actions. Taking all of these considerations into account, it is clearly not prudent to assume that an information advantage can be maintained indefinitely.

FUTURE RESEARCH AREAS

It should be noted that many research issues and areas of investigation remain to be pursued. Some of these issues are briefly summarized below.

COUPLING TO LANCHESTERS-COMBAT THEORY EQUATIONS

Much more research needs to be done with respect to examining the best means of modeling the coupling of the state of information (entropy measure) to the combat equations. For example, is the mapping between the information entropy and the combat equation parameters (d,e,) a linear or a non-linear relationship? Is there some measure of the state of information that is better to use than the entropy? Can the assumed coupling between information state and combat equations be experimentally verified? Are the assumptions regarding the contribution of misinformation to the coupling parameters valid? How do we account for the context-sensitive aspect of the value of a particular item of information? These and other such issues need to be addressed in any future work.

SENSITIVITY OF STRATEGY TO OBJECTIVE FUNCTIONS

The choice of the objective function is a particularly important consideration. Alternative objective functions should be identified and evaluated. The sensitivity of the various factors (i.e., the choice of objective function, of the candidate control strategies to be considered, and of the optimization alternatives themselves) to the effectiveness and performance of the proposed information warfare system needs to be more thoroughly studied and addressed.

EXPERIMENTAL VALIDATION

The information warfare concepts presented in this paper, the exact form of the differential equations which govern the information warfare process, and the coupling of information measures to the equations of combat needs to be experimentally validated and refirmed.

COMPUTATIONAL COMPLEXITY

There is a real issue with respect to the viability and the affordability of practically implementing the algorithuns, hardware devices, and software programs required to compute entropy information state estimates, to generate and evaluate candidate information warfare control strategies, and to display and present these results to the decision-maker.

REFERENCES

1. Force-on-Force Attrition Modelling, James G. Taylor, Jan 1980, Military Applications Section, Operations Research Society of America

2. A Dynamic Model for C3 Information Incorporating the Effects of Counter -C3 P. Moose, Dec 1980, NPS-62-81-025

PR

HELBAT/ACE FIRE SUPPORT CONTROL RESEARCH FACILITY

B. L. Reichard

US Army Ballistic Research Laboratory, Aberdeen Proving Ground, Maryland

Abstract. The HELBAT (Human Engineering Laboratory Battalion Artillery Test) series of field exercises has provided an understanding of field artillery fire support system operations as well as evaluations of promising new materiel and operations concepts. The major thrust of HELBAT 8 was the demonstration and evaluation of new flexible artillery command and control concepts. Such concepts were successfully demonstrated but in so doing, so was the true complexity of the artillery fire support control problem area. While HELBAT 8 was being planned, Ballistic Research Laboratory (BRL) members of the HELBAT Working Group initiated a major work effort that would permit the live play of artillery fire support control functions in a computer-controlled, laboratory environment. Through the exploitation of newly developed interactive operating systems (software), a real-time, multiplayer simulator technology, called ACE for Artillery Control Experiment, was conceived and is now evolving. At the March 1982 HELBAT Executive Committee Meeting, it was agreed that the ACE and HELBAT activities should be joined to create a research or test bed facility with which a combination of laboratory and field exercises could be run.

During the past decade the HELBAT (Human Engineering Laboratory Battalion Artillery Test) series of field exercises has provided a fundamental understanding of field artillery fire support system operations as well as cursory evaluations of promising new materiel and operations concepts. New concepts that were born out of HELBAT exercises include: Closed-Loop Fire Control Technique, wherein unguided area-effects projectiles can be effectively delivered on moving target complexes; Battery Computer System (BCS); M109-based Ammunition Resupply Vehicle; Fire Support Team Digital Message Device (FIST DMD); and on-board gyro-based fire control. More importantly, however, HELBAT serves in general as a learning tool in the area of artillery system research and development. The success of HELBAT can largely be accredited to (1) the joint TRADOC-DARCOM management scheme by which HELBAT exerices are planned and executed, wherein an Executive Committee (EXCOM) provides general direction and a Working Group does the planning and manages the execution and (2) the fact that both baseline and new concepts are studied in a live-fire, total system operations context.

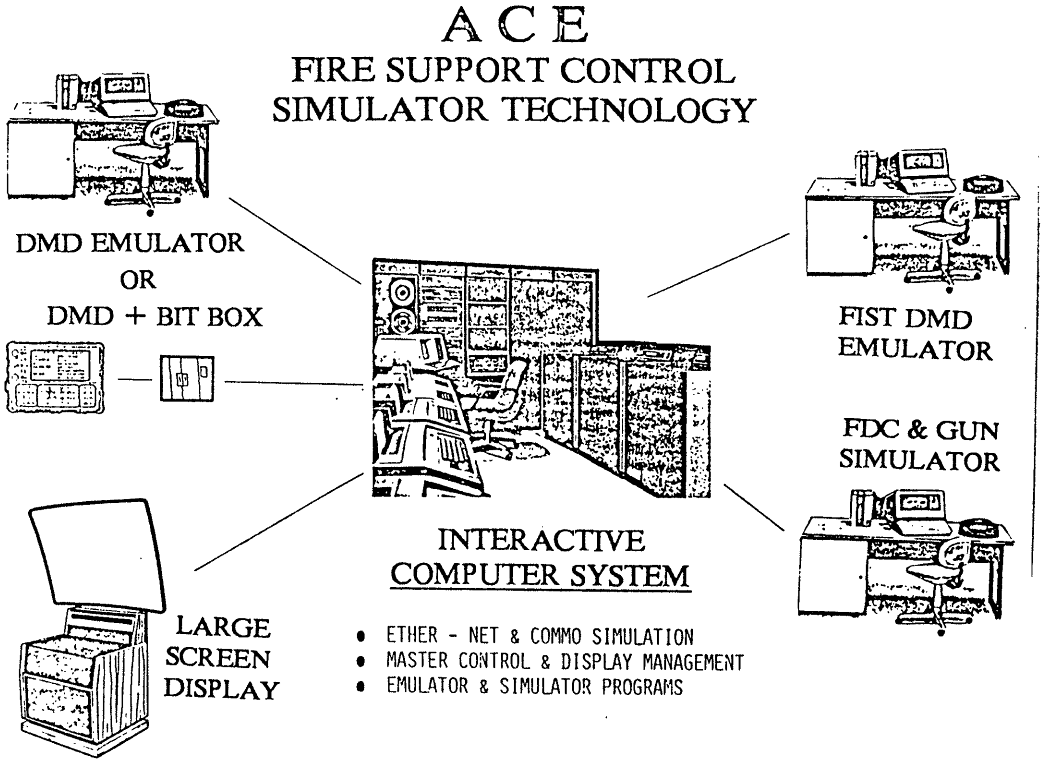
In HELBAT 8 (September-November 1981), the baseline system, against which new concepts are compared, was changed from the voice-manual-FADAC system to the newly fielded tactical ADP TACFIRE system, and the major thrust was the demonstration and evaluation of new flexible artillery command and control concepts, which are designed to permit fuller exploitation of ADP technology than simply the automation of manual procedures. Such concepts were successfully demonstrated but in so doing, so was the true complexity of the artillery fire support control problem area. Total artillery system operations now must include far more than the old-fashioned FO-FDC-gun components. Even with the smallest integral artillery unit (the battalion), many "players," radio nets, fire missions, and data messages must be dealt with in real time, and with the full exploitation of ADP, even the traditional functions of many artillery components may be changed drastically. In other words, HELBAT-8 work pointed out a need for intensive, controlled experimentation in artillery fire support control.

While HELBAT 8 was being planned, Ballistic Research Laboratory (BRL) members of the HELBAT Working Group initiated a major work effort that would permit the live play of artillery fire support contol functions in a computer-controlled, laboratory environment. Through the exploitation of newly developed interactive operating systems (software), a real-time, multiplayer simulator technology, called ACE for Artillery Control Experiment, was conceived and is now evolving. With the ACE concept, components of the fire support control ADP system can be played a number of ways: (1) devices can be emulated on low-cost, commercial video computer terminals; (2) devices or functions can be simulated in interactive computer programs; or (3) actual tactical equipment, fielded or experimental, can be accommodated through the use of the ACE Bit Box device, which interfaces any equipment employing the TACFIRE message protocol and format to a wide variety of commercial computers on which ACE can run. A particular ACE setup can be configured with any combination or number of these components as is needed for the desired application or the organization and operation to be played. Those fire support control components that are not actively played and inputs external to the organization structure being studied can be represented by scenario-based, time-ordered TACFIRE messages read into ACE from a centrally controlled magnetic tape or disc memory unit or by a DMD operator with cue cards. ACE components are interconnected by a program called Ether, which simulates radio nets and characterizes communications from perfect to a selected, degraded probability level of successful data communications for each net. A Master Control and Display Management Program provides for computer control of a particular experiment and permits experimenters to monitor real-time message flow on a large-screen TV or other suitable monitor or printer to instantly extract data such as decision time for a particular player. A sample ACE setup is depicted in Fig. 1.

At the March 1982 HELBAT Executive Committee Meeting, it was agreed that the ACE and HELBAT activities should be joined to create a research or test bed facility with which a combination of laboratory and field exercises could be run under the joint TRADOC-DARCOM HELBAT management scheme. Further, an ACE/CPX (command post exercise) Facility Subcommittee was appointed to provide for near-term, joint guidance in the development of the facility. The facility will be located in a newly built HEL building and will use ACE software provided by BRL and computer hardware and mock-up artillery facilities provided by HEL. Through radio links, laboratory-based exercises can include field elements such as mobile command post vehicle, howitzer, and ammunition handling test beds, as shown in Fig. 2.

This facility will not eliminate the need for live HELBAT field exercises, but it can be used to perform, for example, time and motion studies of the total artillery fire support system, and alternatively, to study selected individual components thereof in a total operations context, thereby identifying field data needs and aiding in the planning and preparation of efficient HELBAT field exercises in the complex total fire support control area. The extreme flexibility of this type of evolving test bed facility is obvious, with a range of applications too broad to cover here. With the development of interactive scenario data bases, for example, active single-thread players, such as a FIST HQ with only one active FO, can be tactically loaded to simulate the actions and reactions of subordinates and higher echelon players, thus eliminating the need to field large numbers of personnel. Flexibility is also enhanced by the ability to mix simulated and real (live) players, which can even include a remote player in another part of the country interconnected to the facility via commercial telephone lines. Using an appropriate mix of generic, developmental, and standard artillery equipment, general research areas, such as decision and control theory and operator interface technology, can be studied, and new hardware, software, and "skinware" technology application concepts from such research can be explored. As a technology in itself, the HELBAT/ACE Research Facility concept could also be utilized and further developed as an automated CPX facility at Ft. Sill for field artillery training in the new tactical ADP world; in addition, this CPX facility could be used by the combat developer and trainer to investigate alternative operations and organization concepts. The Field Artillery School is now considering this application.

Specific study exercises to be run in the HELBAT/ACE Research Facility will be planned and executed by the HELBAT Working Group under the general direction of the HELBAT EXCOM. In the near term, validation experiments are being planned to determine whether operations in the laboratory facility can duplicate selected fire missions accomplished in the field at HELBAT 8. Some of the first actual study exercises will probably include: adding tactical scenario loading to the players used in fire missions run at HELBAT 8 without tactical loading and running HELBAT-8 fire missions with degraded communications. These exercises will initially be limited, of course, by hardware and software capabilities of the facility. As additional facility components, such as TACFIRE and generic terminals for battalion fire support elements, are developed or acquired, however, all HELBAT-8 type fire missions can be run in the facility; then a program to add other fire support functions, such as fire support planning, should perhaps be considered. Use of the HELBAT/ACE Research Facility will also be planned in joint laboratory-field exercises run to explore field artillery concept work areas identified at the last HELBAT EXCOM Meeting, namely: improved data communications performance of FM push-to-talk radios; use of the air observer and elevated platforms as high technology target acquisition devices; artillery use of an NBC-protected command post vehicle; and further advancement of on-board weapon computers, e.g., on howitzer test beds.



- ETHER - NET & COMMO SIMULATION
- MASTER CONTROL & DISPLAY MANAGEMENT
- EMULATOR & SIMULATOR PROGRAMS

Fig. 1

HELBAT/ACE
FIRE SUPPORT CONTROL
RESEARCH FACILITY

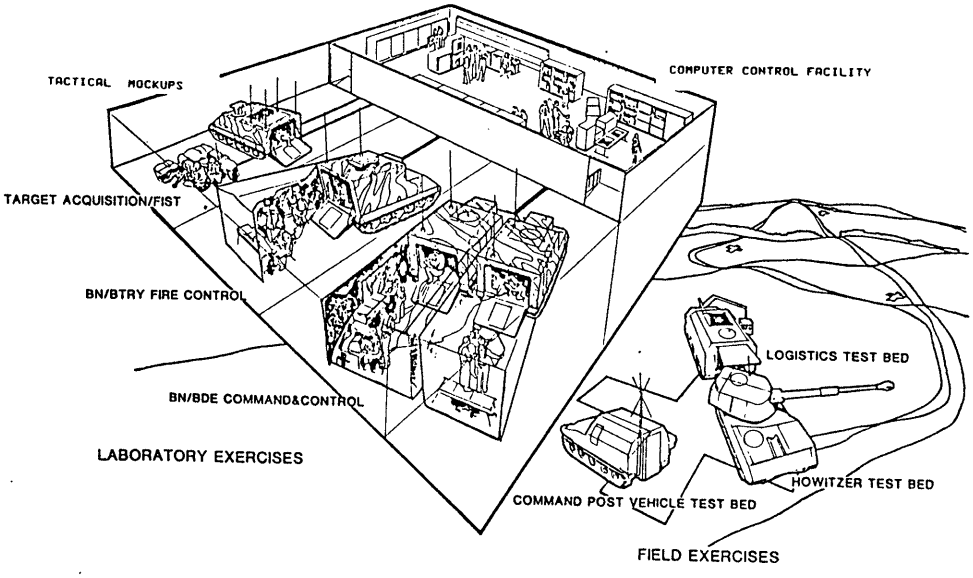


Fig. 2

A TOTAL SYSTEM APPROACH TO COMMAND SYSTEM DESIGN

P.D. Morgan

Scicon Consultancy International Ltd.,
49 Berners Street,
London W1P 4AQ, UK

Abstract A top-down analysis of the command and control of individual ships and of groups of ships is described. This analysis is organised in terms of; command system operation, command system structure, command system functions and the information system. This analysis is used to develop a command system design methodology in an implementation independant form and to make explicit the assumptions upon which it is based.

INTRODUCTION

This paper examines the command of single ships and groups of ships with the objective of deriving a design methodology based on explicitly identified premises and appropriate to a variety of applications and levels of technology.

The paper is organised in two sections; the first develops general command system concepts from a set of basic premises, and the second extends these concepts in terms of naval C^2.

GENERAL SYSTEM CONCEPTS

Premises

The Command System is that combination of men and machines which is used by the Commander in the planning, executing and monitoring of activities to achieve his objectives; as such it includes both subordinate commanders and the Commander's planning staff. The Commander is effectively external to his Command System when he configures it and sets objectives for it; and he forms a part of it when he works with it to achieve these objectives. In its operational timescale (fractions of a second up to days) the Command System must have the following attributes:-

- it must be a Relatively Closed System(1) which interacts with its environment through a limited number of well defined channels;.
- it must display Ideal - Seeking Behaviour in the development of new improved responses and in their evaluation on the basis of experience;
- its operation must be definable in terms of a Closed Loop Control Process;
- its organisation must be distinguishable in terms of system structure and system functions.

Command System Operation

The operation of the Command System is modelled by the closed-loop control process shown in Figure 1, the elements of the process are:

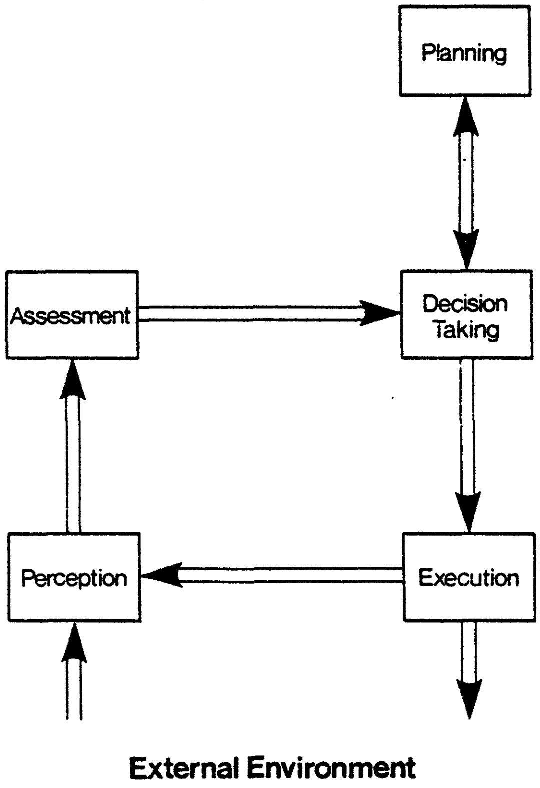


Figure1 Command System Operation

- Perception, the gathering of relevant information about the external environment and about actions being executed;
- Assessment, the comparison of the observed and predicted situations to determine discrepancies and the consequent corrective actions required by the plan;
- Decision Taking, the use of integrated outputs from Assessment to determine between the options of continuing with the present plan, selecting an alternative or initiating new planning activities;
- Execution, the implementation of planned activities;
- Planning, the development, assessment and selection of plans; this represents a further closed loop process operating an information abstracted from the main loop.

Comand System Structure

The majority of Command System's activities do not require this ideal-seeking capability in the time-scales of interest; consequently the command system is organised in the form of an hierarchical structure consisting of ideal-seeking, multigoal-seeking and goal-seeking elements. At the lowest level exist numerous goal-seeking elements consisting of simple feed-back control systems with specified predefined control rule sets; these are directed and tasked by a limited number of multi-goal seeking elements, each of which has a range of control rule sets plus a higher level rule set for selecting the appropriate control rule set; at the top is the ideal-seeking element which has rule sets for developing and evaluating plans on the basis of theory and experience, and for formulating goals for the multi-goal seeking elements.

These elements are distinguishable in terms of speed and complexity of response; in order to reflect their ranging speeds of response they are refered to an Medium Term (ideal-seeking), Current (multi-goal-seeking) and Immediate (goal-seeking).

The extension of the approach to cover the requirements of both ship and group command leads to the development five level Command System Structure shown in Table 1.

Immediate activities do not occur at group level as resource control is always a ship function.

The vertical arrows in Figure 2 show the command relationships between the various Command System Categories, and the diagonal arrows show potential command relationships. The apparent conflict between Categories 2 and 3 for the direction of Category 4 arises because ship resources (long range sensor and weapon systems) may be allocated for centralised Current direction within the group, this allocation may be overriden by the ship Commander in an emergency.

TABLE 1 Command System Structure

Command System Category	Command System Functions
Category 1 (Medium Term)	Group command and coordination of Current activities.
Category 2 (Current)	Direction of specific classes of group operations
Category 3 (Medium Term)	Ship command and coordination of Current activities
Category 4 (Current)	Direction of specific classes of ship operations.
Category 5 (Immediate)	Control of individual resources

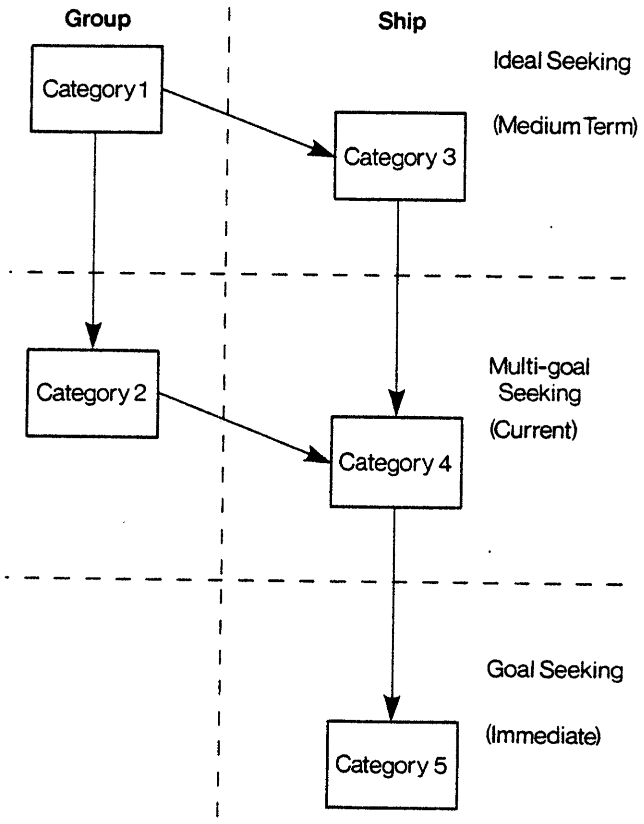


Figure 2 Command System Structure

Command System Function

The premise that the Command System is Relatively Closed results in the definition of two classes of Command System Functions; these are Task Definition functions concerned with interactions with the external environment, and Resource Management functions which are involved in maintaining and adapting the internal status of the Command System and its resources.

Information Requirements

Examination of the Command System operation in terms of Command System Structure and Functions leads to the identification of five classes of information requirements:-

- Immediate World information consisting of the positions, movements and emissions of all observable objects; this information has currency ranging from fractions of a second up to minutes;
- Current World information in which the objects under observation are categorised in terms of hostility, type and identity, in this case the currency ranges from the tens of seconds to hours;
- Medium Term World information corresponds to the assessed situation containing objects and groups of objects which have been evaluated in terms of significance and intent; the currency of this information ranges from tens of minutes up to days;
- Support information relating to the

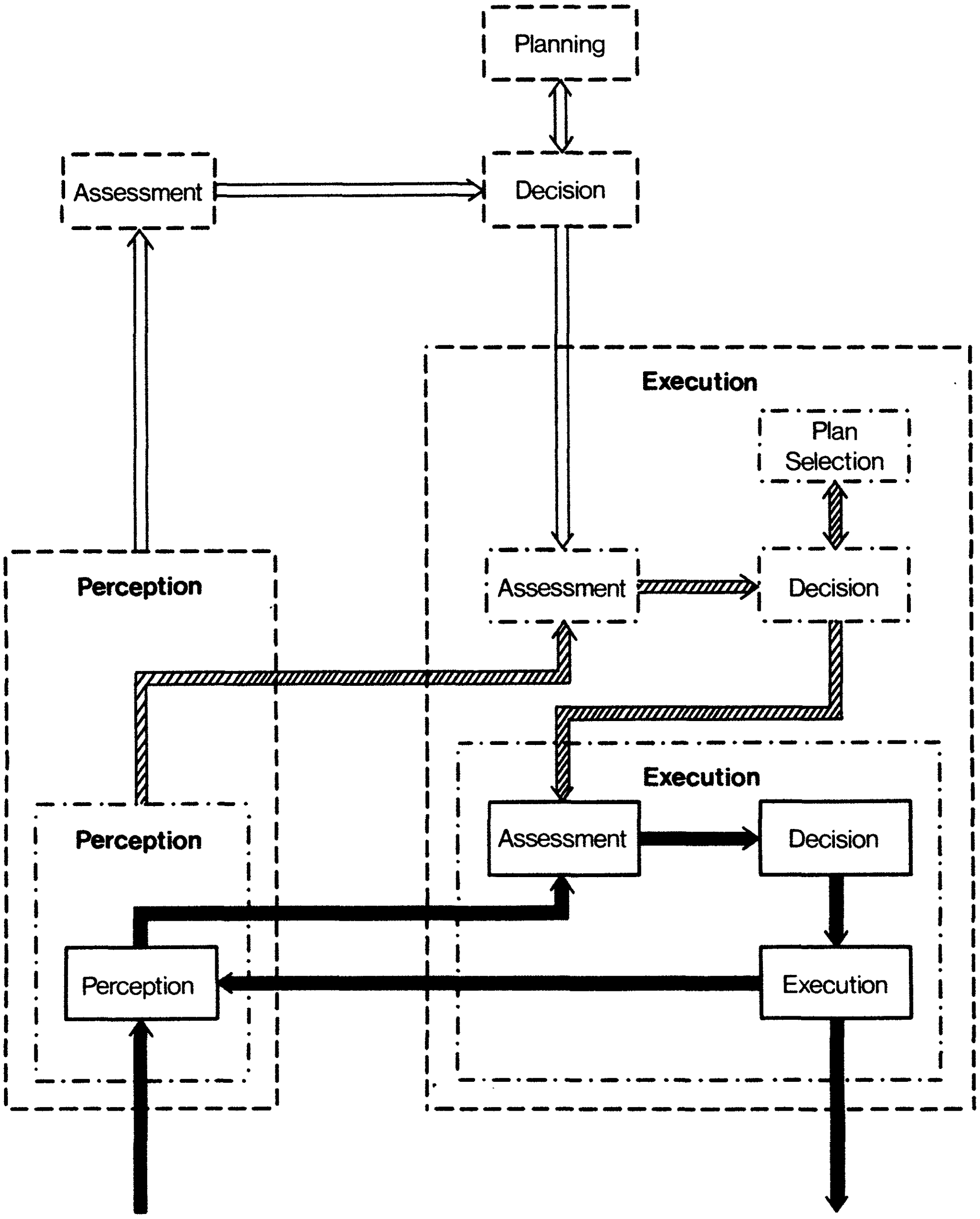


Figure 3 Nested Command System Operation

status of the Command System, its elements and the resources available through it;

- Encyclopeadic information on equipment parameters, Standard Operational Procedures, Orders of Battle, etc.

COMMAND SYSTEM DEFINITION

Command System Operation

Integrating the Command System Structure definition with the model of the Command System operation leads to the extension of the model in terms of nested processes as shown in Figure 3, this figure addresses the nesting of the Execution phase of the process; a subsidiary feature of this figure is that it illustrates the nesting of the Perception phase which is necessary to ensure each Category sets appropriate goals for its subordinates.

This model embodies two sets of assumptions:

- each cycle operates within a stable internal environment in that its objectives change slowly in comparison to the rate of change of the environment within which it is working to achieve those objectives;
- each Category can interact with its subordinates by defining and modifying objectives, as shown in this model, and by controlling the criteria used to select particular plans for achieving those objectives, this latter interaction is not shown in this model.

The conditions under which this model is appropriate have been examined in terms of discrete control processes, the conditions correspond to an order of magnitude difference between the cycle times of successive categories and to an equivalent relationship in terms of information coverage and resolution; these results correspond with those developed by Dr Castanon(2).

In the event that the cycle times of two Command System Functions are similar in magnitude, two options are available, either the two functions may be separated by the definition of distinct areas of responsibility with some overall coordination (c.f. the separation of Missile Engagement Zones and Combat Air Patrol zones), or the functions may be amalgamated by ensuring that they share common goals and plans. In the alternative case where the cycle times differ by much more than an order of magnitude the available options are: the introduction of an intermediate function to bridge the gap, or the direction of the subordinate function through the definition of the criteria for the selection of particular plans and the provision of generalised goals (c.f the control of 'Close In Weapon Systems' which carry out independant threat evaluation and sequencing).

Command System Structure

The separation of functions into Task Definition and Resource Management functions results in a corresponding extension of the Command System Structure (Figure 2) to support the two hierarchies of functions in their fixed relationships. The direction of the Resource Management functions by the Task Definition functions is controlled by resource allocation matrices which define the priority which each Task Definition function has in setting tasks for each resource (sensor, weapon system, etc.). These matrices are set up by the group and ship Commanders in accordance with their objectives and their appreciation of the situation.

This formal organisation of Task Definition and Resource Management functions is necessary to permit both the effective management of the resources and the flexible deployment of them to meet the perceived requirements.

Command System Functions

The Resource Management functions result from the grouping of resources in terms of common operating environments, potential mutual interference and contribution to internal Command System requirements (e.g. Communications); examples would be the grouping of Electronic Support Measures equipments with sensors, and the organisation of sensors, Electronic Counter Measures and communications under an overall Emission Management function. Thus the Resource Management functions are predetermined by the ship fit and the make-up of the group, and remain fixed within the time-scales of interest.

The Task Definition functions are not predetermined in this manner; they reflect the Commander's objectives, both offensive and defensive, his assessment of the situation and the capabilities of his resources. The conventional divisions into Anti-Air Warfare, Anti-Submarine Warfare and Anti-Surface Vessel Warfare provides a basis for analysis and scaling of system operation; however any specific mix of objectives, resources and threats may lead the Commander to employ a significantly different task division (e.g. the use of ASW Frigates to provides to provide short range air defence in an environment with a high level air threat and a low-level submarine threat).

Information Structure

The information requirements of the Command System are met by the provision of a range of group and ship World Models, Support Models and Encyclopaedia (Figure 4); the maintenance and use of these model froms the subject of a separate paper(3).

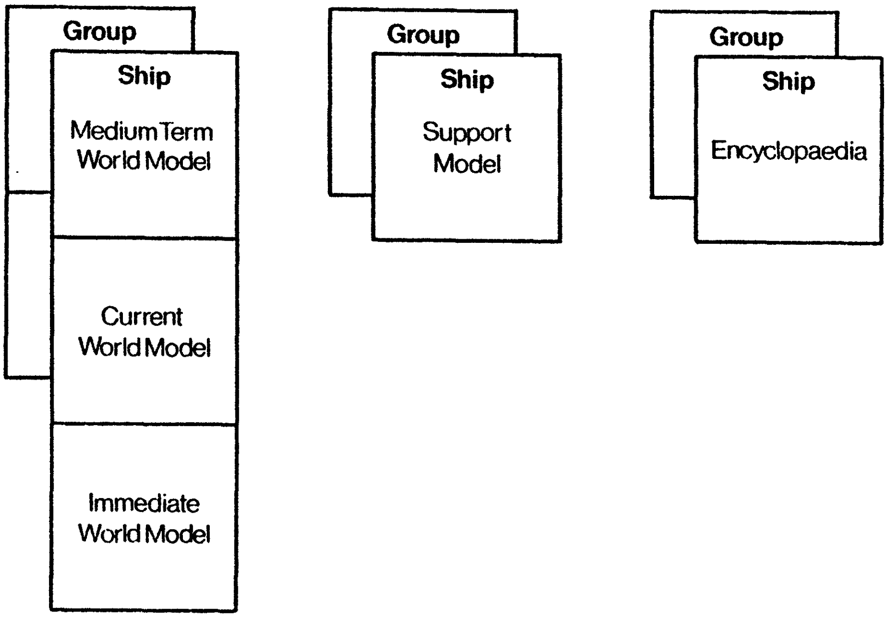


Figure 4 Information Organisation

The World and Support Models provide dynamic representations of processes occuring in the external and internal environments of the Command System; the structuring in terms of group and ship, and in terms Medium Term, Current and Immediate is done to maintain conformity with identified classes of information requirements.

CONCLUSIONS

The objectives of this high level approach were to provide a self-consistant approach to command system design in an implementation independant form, to make explicit the assumptions on which it is based, and to provide the basis for more detailed investigations. These objectives have been largely achieved, although much of the analysis has been qualitative rather than quantitative, a limitation which was accepted as inevitable given the breadth of the topic.

ACKNOWLEDGEMENT

The paper arose from work carried out with the support of Procurement Executive, Ministry of Defence by the C^3 Group of companies (Ferranti Computer Systems Ltd., PA Management Consultantants Ltd., Plessey Radar Ltd., Scicon Consultancy International Ltd., Software Sciences Ltd., and System Designers Ltd).

REFERENCE

1. Klir J. and Valach M.; 'Cybernetic Modelling; Prague, 1965.
2. Castanon, D.A.; 'Games with Uncertain Goals'; Proceedings of the 4th MIT/ONR Workshop on Command and Control; 1981.
3. Morgan P.D; Information Management for Command and Control'; Proceedings of the 5th MIT/ONR Workshop on Command and Control; 1982.

INFORMATION MANAGEMENT FOR COMMAND AND CONTROL

P.D. Morgan

Scicon Consultancy International Ltd
49 Berners Street.,
London W1P 4AQ, UK.

Abstract The related problems of information capture and information dissemination are examined within the context of a naval command system. Information capture is examined in terms of the conversion of sensor data into command information, the production of an hierarchy of world models, and the adaptive control of the surveillance operations. Information dissemination is discussed in terms of Volumes of Interest and Information Resolutions which provide the basis selecting information for distribution to individual users.

INTRODUCTION

The acquisition, organisation and dissemination of information are critical tasks within any command system. The continuing extension of the naval combat horizon and the consequent increases in the quantity and complexity of information to be handled threatens to overwhelm the existing Action Information Organisations.

The problem was addressed as part of a study of naval command systems(1); the concepts discussed in this paper forms a part of the overall command system methodology.

The factors taken into account in this study are:-

- the characteristics of the inputs, which range from tracks and bearing lines input by own sensors, to position and intended movement reports, sighting reports and intelligence summaries received in the form of Signal Message Traffic; these inputs vary in format, timeliness, accuracy and levels of prior processing;
- the need to validate information and guard against the propagation of erroneous information, whether arising from human or equipment failure or from enemy counter-measures;
- the need to be responsive to user requirements for timely and relevant information and for the exclusion of irrelevancies;
- the necessity of aiming for economy in the use of processing power and storage.

INFORMATION COLLECTION

In order to set up an information system rather than a data system it is necessary to distinguish between data, information and knowledge, particularly as these terms are user and task dependant. The basic premises upon which this paper is based are:

- information is the result of using knowledge to set data into an appropriate context;
- information at one level corresponds to data at higher level and to knowledge at a lower level

The process of converting sensor data into command information is represented by the three stage model shown in Figure 1; in this model no distinction is drawn between information collection at different sites, consequently no Data Links or Signal Message Traffic are shown. The three stage within the model are:

- the validation of data and the formation of tracks through the correlation of successive inputs from single sensors and the association of data from different sensors;
- the classification of these tracks on the basis of their aggregate characteristics to determine hostility, type and identity;
- the interpretation of the situation on the basis of the disposition and movements of the various classes of tracks.

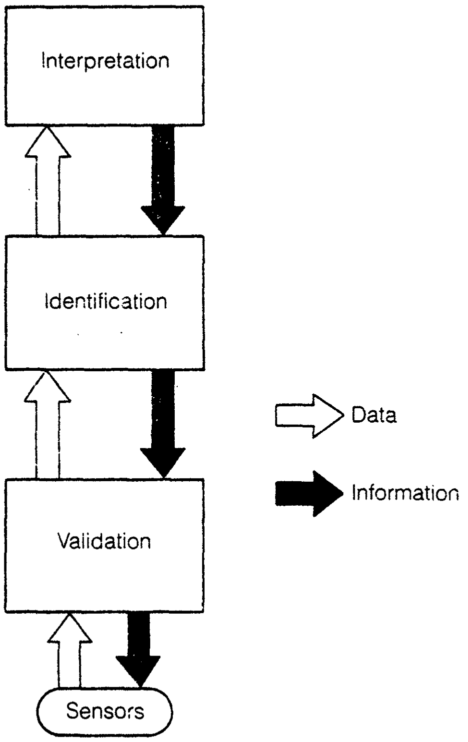


Figure 1

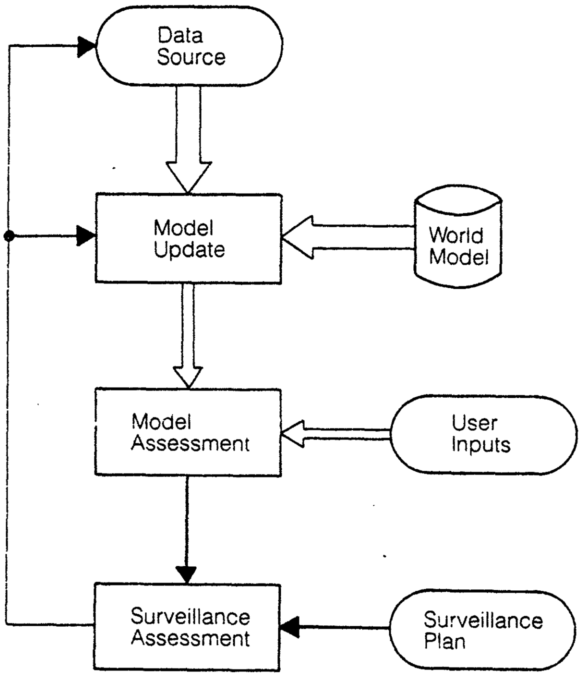


Figure 2 World Model Management

These stages correspond to those necessary to form the Immediate, Current and Medium Term World Models described previously(1).

In addition to the upward flow of data between stages there is a complementary downward flow of information which determines the context within which the data is categorised (e.g. the classification of a contact as a hostile bomber will alert the validation stage to possible track-splitting).

The execution of each of these stages is carried out by a separate World Model Management System as illustrated in Figure 2; each of these systems is divided into sections concerned with data processing for Model update, assessing the status of the Model, and directing the Surveillance process. The separation of the Model Management systems is intended to limit the propagation of errors by preventing false hypotheses from contaminating their source data.

Model update is concerned with using the input data to initiate and update hypothesis in the World Model and to establish confidence levels for these hypotheses on the bases of data quality and correlation with predictions from the World Model.

Model Assessment employs statistical analysis of track quality parameters to maintain a continuous overview of the Model status; the parameters to be evaluated include the range at which categorisation occurs; the ratio of hypotheses to events, and the life times and confidence levels of hypothesis; and the ability to meet user requirements in terms of routine provision of information and of unsatisifed user requests for additional information. These parameters are analysed to detect correlation with range, sector, track category, time, etc and thus to provide warning of Model degradation resulting from equipment malfunction, environmental changes or hostile counter-measures. The results of these analyses are used by Surveillance Assessment to identify significant departure from the Surveillance Plan.

The Development of the Surveillance Plan is external to World Model Management; it is based on the assessed situation, on the group or ship objectives, and on the predicted conditions. Surveillance Assessment provides the means of implementing this plan and of adapting it in response to changing conditions. The implementation is carries out by setting surveillance objectives and defining contexts and modes of operation for subordinate World Model Management systems in response to the discrepancies identified by Model Assessment.

Thus surveillance operations of all levels from the deployment of sensor platforms to the control of individual sensors are continuously adapted to meet the demands of the environment, the requirement of the users and the status of individual equipments(2). The subdivision of this process by the provision of a number of World Model Management systems limits the depth and range of analysis required at each level, thereby reducing the overall processing requirements.

DISSEMINATION OF INFORMATION

In theory the information required for the execution of the various Command System functions is definable in terms of the model of the Command process (1), in practise this is true only when the functions are completely predefined and lack any capability for learning and adaption. The approach is to divide information distribution into 'routine information presentation' and 'user requested information presentation'. Routine information is based on the model of the command process; it is expected to supply all the information requirements at Immediate Level (Command System Category 5(1), the bulk of the requirements at Current level (Categories 2 and 4) and to provide the context for user information requests at Medium Term level (Categories 1 and 3). The remainder of this paper is concerned with the filtering of information for routine presentation and does not consider the handling of user requests.

Routine information presentation is based on the related concepts of the Volume of Interest and the Information Resolution. A Volume of Interest corresponds to a volume within the information space of one of the World Models and the associated Information Resolution equates to the quantization of information required within the Volume of Interest. The definition of Volumes of Interest for the various Command System functions assists the Commander in configuring his Command System in accordance with his objectives, in addition the Volumes of Interest and Information Resolutions provide criteria for use in the Model Assessement stage of World Model Management.

The axes of the Volume of Interest correspond to the parameters used in the categorisation of information within the World Models; a typical set of axes are:

- geographical, this relates to the area over which the function is exercised (c.f. the Missile Engagement Zone);
- time-frame, which corresponds to the view into the past and future required

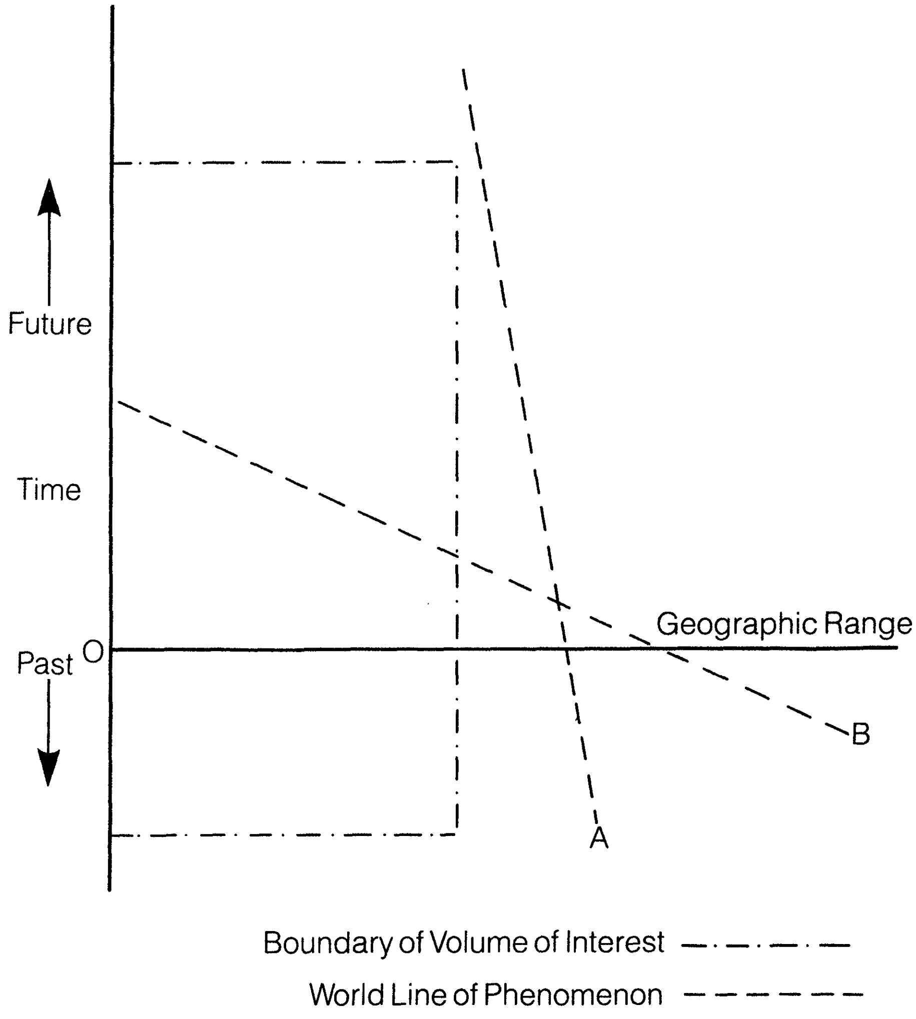


Figure 3 Operation of Volume of Interest

for scheduling, directing and coordinating activities;

- category, which identifies the appropriate class of information (e.g. air, surface or sub surface).

The operation of the Volume of Interest in the selection of relevant informaiton is shown in Figure 3. In this diagram the geographic axes have been compressed to produce a single axis, geographic range, and the categorisation axis has been ignored; the origin represents the centre of interest with respect to range, and the present moment in relationships to time. The various objects and phenomena (e.g. threat tracks) are represented as world-lines, with their present positions indicated by the points of intersection of the world-lines and the Geographic Range axis. The criterion for reporting an object becomes one of testing whether its world-line enters the Volume of Interest (shown chain dotted); thus B is reported although the physically closer object A is not. The asymetry of the Volume along the time axis arises because the direction and monitoring of present activities required a relatively short view into the past compared with the view into future required for scheduling future actions.

In practice errors in measurements of the positions and movements of objects, object manoeuvres, and uncertainties in prediction will lead to the objects being represented to envelopes of world-lines; additionally, the Volumes of Interest are unlikely to have the geographic symmetry tacitly assumed in producing this figure, removing any particular significance from the geographic centre of the Volume.

The previous paper(1) examined some of the implications of an hierarchical command system, one of which was the development of lower level (Current and Immediate) goals and objectives from the expansion of higher level (Medium Term) objectives, this translates into a requirement that the subordinate Volumes of Interest must be included within the superior Volume of Interest. Figure 4 illustrates the nesting of Medium Term, Current and Immediate Volumes of Interest relative to some arbitrary common set of axes; the squares marked Medium Term, etc, correspond to the projections of the volumes of Interest into this coordinate system, and the tessellations within these squares indicate the respective Information Resolutions; the numbers 1 to 4 indicate events occuring within these Volumes of Interest. The variation in size of the Volumes of Interest

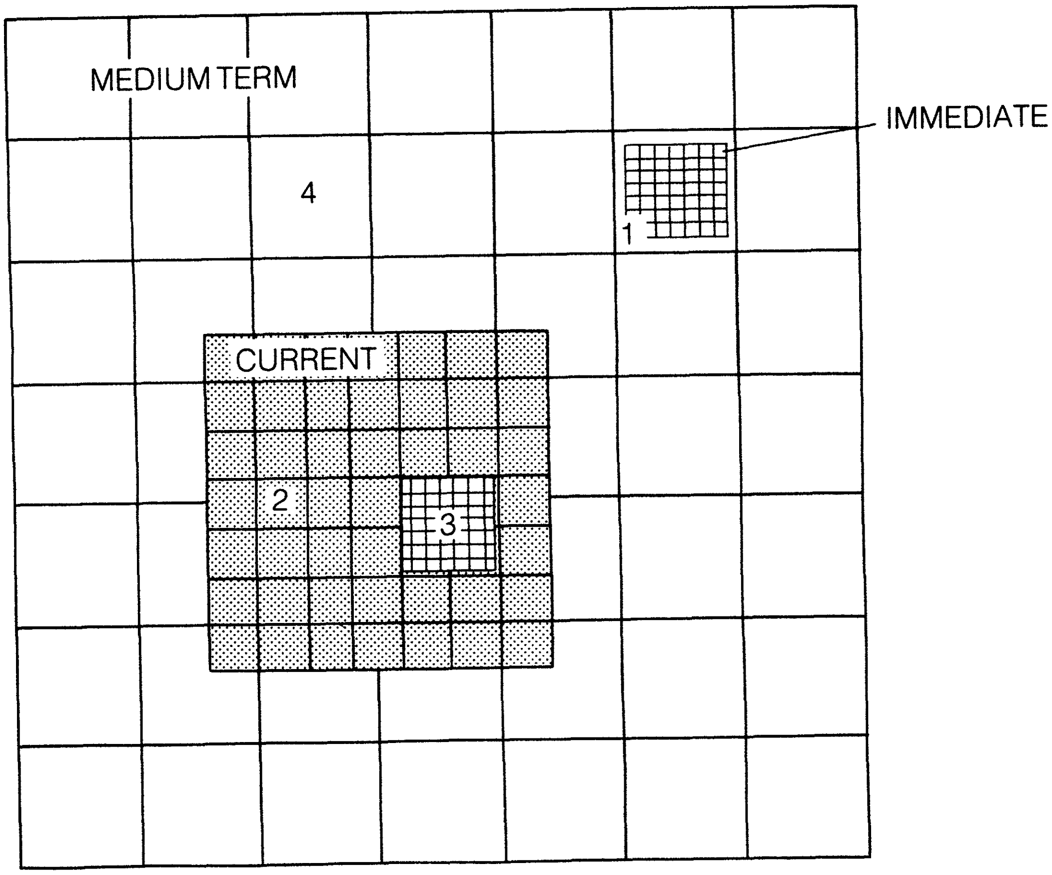


Figure 4 Relationship of Volumes of Interest

and Information Resolution corresponds to the variation in scope and detail between Medium Term, Current and Immediate information requirements.

On the basis that communication requires a known shared contest to make it intelligible, it is apparent that <u>routine</u> communications between the functions utilising these Volumes of Interest is limited to the following cases:

- Medium Term and Current can communicate about events 2 and 3;
- Medium Term and Immediate cannot communicate, as the inclusion of the Immediate Volume of Interest within the Medium Term Information Resolution cell prevents the Medium Term function from determining whether event 1 or 3 are observable by the Immediate functions.

Non-routine communications require the collection or communication of additional information to provide the necessary context, for example the Machine Term function may obtain more detailed information via a user request on the World Model before attempting to communicate with the Immediate function about event 1.

A corrollary to this constraint on routine communication is that functions which have overlapping Volumes of Interest must routinely communicate on actions undertaken within the overlap region to permit coordination; for example Anti-Submarine operations using helicopter or maritime patrol aircraft must be coordinated with anti-air warfare operations in areas where they overlap.

These concepts are applicable to the employment of Tactical Data Links as well as to communication within a single ship; applied to Tactical Data Link operation they make it apparent that the tasks to be supported are; the maintenance of a common context within the group, the exchange of command and control messages based on the common context, and the exchange of command and control messages plus supporting context for events outside the existing shared context. The bulk of the link traffic should be concerned with the maintenance of the common context, and the transfer of out of context command and control information plus supporting information should make up a very small proportion of the traffic; consequently it is of interest to consider whether a single link can be optimised to support the three forms of traffic. A further question worthy of consideration is whether a single link can support the requirement of both Medium Terms and Current command and control.

CONCLUSIONS

The related concepts of the World Models and of Volume of Interest and Information Resolutions provide a basis for the management of information gathering and distribution in a manner which is continuously responsive to the needs of the various users and in a form amenable to a high level of automation. The definition of the information management system in this form also makes it more readily understandable by its users.

ACKNOWLEDGEMENTS

This paper originates from work carried out with the support of Procurement Executive, Ministry of Defence by the C^3 Group of companies; the group consisted of Ferranti Computer Systems Ltd, PA Management Consultants Ltd., Plessey Radar Ltd, Scicon Consultancy International Ltd, Software Systems Ltd., and System Designers Ltd.

REFERENCES

1. Morgan, P.D: 'A Total System approach to Command System design; Proceedings of the 5th MIT/ONR Workshop on C^2; 1982.

2. Foster, S, Schloss, W.A. and Rockmore, A.J; 'Towards an Intelligent Editor of Digital Audio: Signal Processing Methods', Computer Music Journal, 6(1), pp 42-51; 1982.

THE ROLE OF TEAMS IN THE ORGANISATION OF WORK

J.N. Clare,

Scicon Consultancy International,
49 Berners Street,
London W1P 4AQ, UK.

Abstract Where men work together it is possible to consider two basic types of working relationship, those of Team or Group. A Team consists of ~~of~~ coordinating specialists while a Group is made up of collaborating generalists. When considering where men operate within a command system the concepts of information management can be used to select the most appropriate relationship for any collection of men. Finally some of the advantages and disadvantages of each organisational relationship are considered.

INTRODUCTION

In considering the design of a command system it is necessary to determine how the way men work together interacts with the design. This paper discusses some of the characteristics of collections of people working together. The positions where men will function with a command system are then described. The concepts related to information organisation and management are then used to suggest the most relevant organisation of work in any given situation. Finally some of the implications, advantages and disadvantages of working structures are discussed.

MEN WORKING TOGETHER

This general review is based on the work of Hallam and Stanmers of Aston University on 'Multi-Man-Machine Systems'(1).

The first point is to consider the ways that men working together may be organised Briggs & Naylor(2) define a Team as "two or more operators working in task or goal oriented environment".

Klaus and Glaser(3). go further in defining both a Team as fairly loose structure.

Klaus and Glaser define four basic characteristics which differentiate between a Team and a Group as follows:-

(a) Structure; in a Team this is well defined with each individual having a clearly defined relationship with the other members of the organisation. A group has an ill define structure.

(b) Individual assignment, in a Team each member has a clearly defined role with associated defined responsibilities and task requirements. In a Group the members share responsibility for the total task rather than any sub-task

(c) Individual contribution, in a Team member is a specialist in the tasks he does and his inter-relationship with the other members is one of Co-ordination. In a Group each member will be more or less capable of fulfilling all the sub-tasks, i.e a generalist. The inter-relationship between individuals will be that of collaboration.

(d) Overlap between individuals, in a Team there is little overlap between individuals tasks, responsibilities and relationships. In a Group the overlap may be total.

A closer examination of the concepts of colloboration and co-ordination suggests that these two underlie the differences between Groups and Teams. Thus where the relationship individuals is that of co-ordination then it is closer to the definition of a Team, when the relationship is that of collaboration then it is closer to the definition of a Group. This latter assertion assumes that for these to be joint responsibility the structure must of necessity be looser.

It is now necessary to reasses the definitions of a Team and a Group. In so far as we can determine the relative degree of co-ordination to collaboration, then a collaboration, then a collection of men can be called a Team or a Group. However, in affect, we have a continuous distribution from a Team with only co-ordination relationships, through a mixture to a pure Group where all relationships are collaborative.

There are two specific extra requirements for the existance of a Group compared with a Team. These are training and interpersonal communications. Group members must have sufficient experience of working together in order to establish effective models of each others personal internal models of the world. In a Team the representation of the others internal model may be common across all individuals carrying out a specific task. In order to develop and maintain the models of other individuals within a Group there will a consequent increased load on interpersonnal communications. Where there is not direct contact there will need to be provision of adequate communication facilities.

OPERATIONAL POSITIONS

An Operational Position is defined as the node in the organisation where men fulfil roles. The concept of an operational Position is derived from the logical structure and function of the system. The implication is that any position may require many men or only part of a man's effort.

In the paper on the Total Systems Approach(4) it is shown that the structure may be considered under two headings namely Task Definition and Resource Management. In the simplest case we can define five basic types of position,

a) Command Cm

A position which has the authority and responsibility for the direction co-ordination and control of military resources.

Operations Direction OD

A position which has the authority and responsibility for the direction of allocated resources within a defined Sphere of influence.

Resource Management M

A position which is responsible for the timely provisions of a resource, for maintaining the effective operation of that resource and for making recommendations for the effective use of that resource.

Resource Control

A position which is responsible for the immediate manipulation of a resource in order to achieve a defined goal.

Co-ordination Co

A position which is responsible for identifying where conflicts occur as a result of the requirements placed on the system by users. Co-ordination is carried out at the behest of the commander, but since it is assignable to another it is separately defined.

In the paper on Total Systems Approach it is shown that in the complex Naval environment there is need to consider resources at both Group and Ship level. This leads to a nesting of operational positions with respect to the various Command System Categories.

From this nesting it is possible to derive a general set of Operational Positions at Group level, figure 1. This shows the basic division of Task Definition functions and Resource Management functions; at Group level there are no control positions as resources only exist at ship level.

The Operational Positions for a ship are shown in figure 2. In this case the inter conectivity is complicated both by the need to meet both Area and Self Defence requirements and also by the direction of those aspects from ship Command and from Group Operations Direction.

ASSIGNMENTS OF POSITIONS TO MEN

Any operational position may be filled by any number of men from a part of a man to many men. This leads to three basic assignment situations:-

a) The division of roles within a position betwen several individuals.

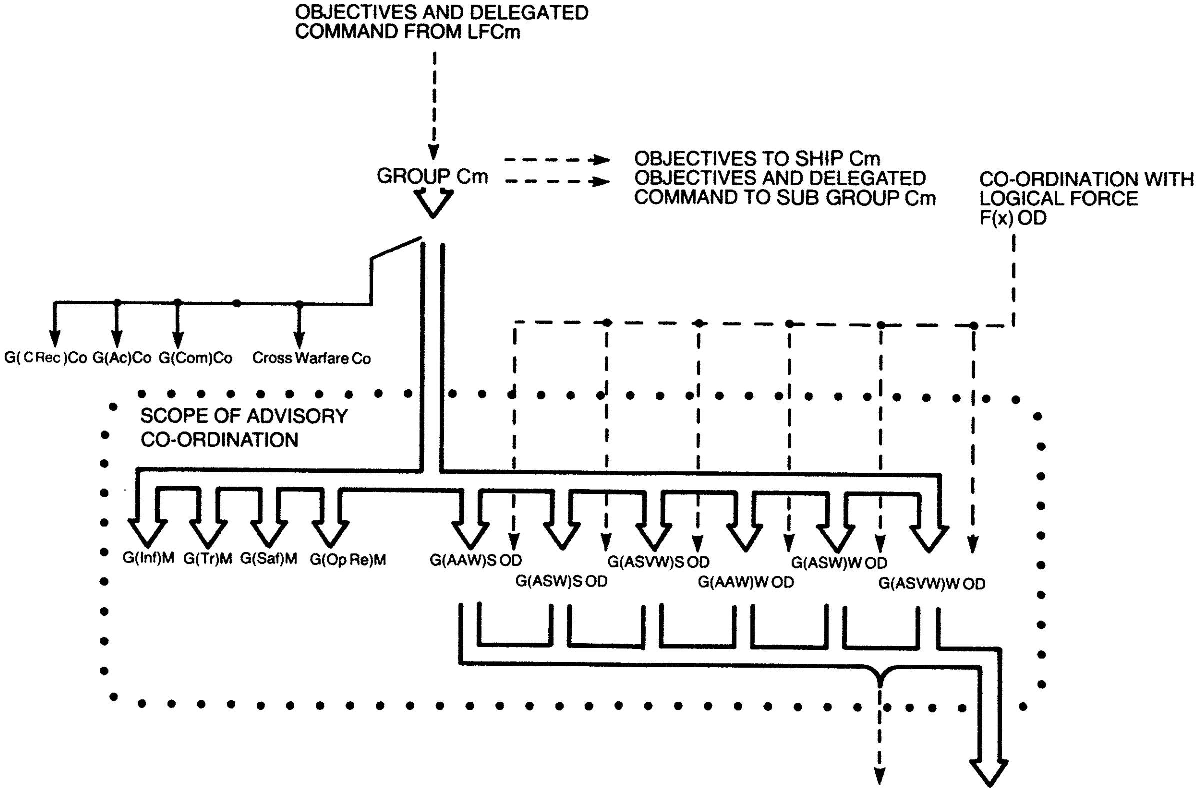
b) The mapping of a single position to a single man.

c) The aggregation of several position for a single man.

A critical aspect in the assignment of roles to individuals is the understanding that is necessary for them to have comprehensions of events within and external to the command system. It is suggested that for a man to be effective in a number of concurrent roles then the Volumes and Information Resolutions associated with those roles must be compatible. The optimium being when the Volume of Interest and Information Resolution are identical.

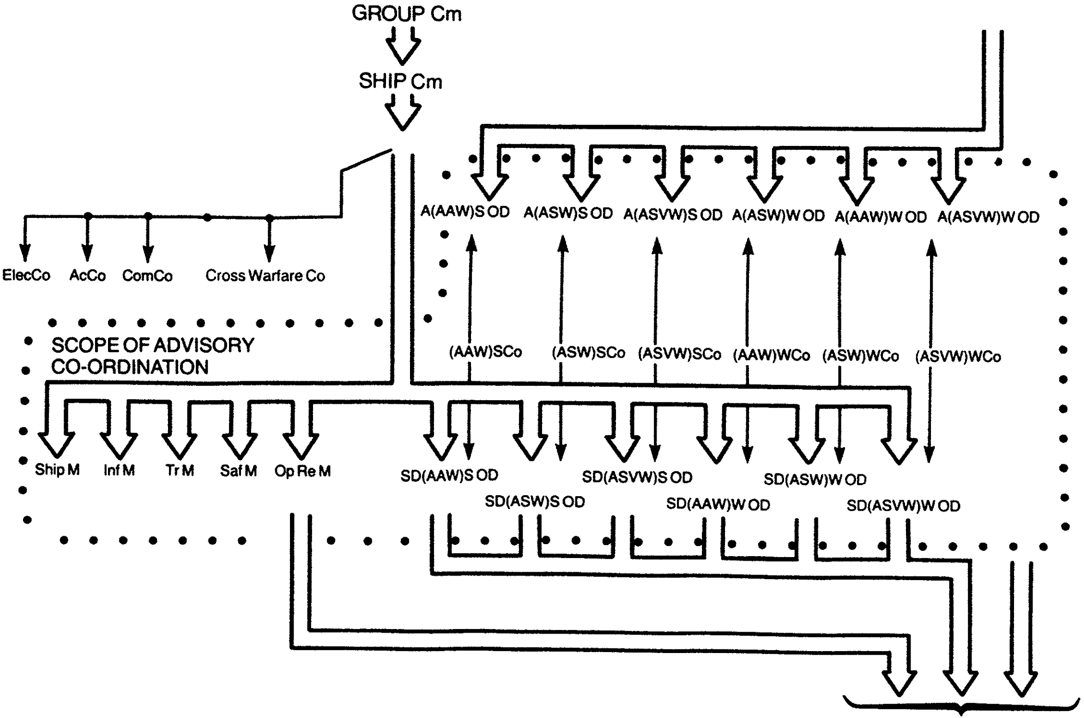
Where a number of men occupy an operational position then it is possible to divide the subtasks between individuals so that Volumes of Interest either overlap or do not overlap. Where there is overlap there will inevitably be an increased requirements for co-operation between individuals.

When a number of men share an operational position with high overlap between Volumes of Interest, then a Group relationship will the most appropriate. When men do not share Volumes of Interest but are required to co-operate then a Team relationship will be the most appropriate.



Operational Positions at Group Level

Figure1



Operational Positions at Ship Level

Figure2

In the other assignment situation of one man fulfilling the roles associated with several position or completely filling one position then he will normally have a Team relationship with the rest of the command system personnel.

In a more complex organisation it may be possible that a man may fulfil a role which is part of an operational position which has an associated group relationship. However, it is possible for that man to have an additional role which means a Team relationship with others and thus an essential conflict.

This conflict needs to be considered carefully as it is possible that the group relationship may provide a more satisfying job and hence bias the man away from his other responsibilities.

IMPLICATIONS OF USING TEAMS OR GROUPS

Given a Team or Group relationships it is necessary to consider the advantages and disadvantages; some of the more important implications are:

a) Resilience to external change - because of the flexibility and adaptability of a group it should be resilient to unexpected events occuring. A Team will require an external supervisor to adapt and manipulate the individual roles in order to meet the new situations.

b) Resilience to internal change - because a group depends on the collaboration betwen individuals it will be susceptible to changes in Group membership. Because of the independence of individuals a Team will be resilient to changes in membership.

c) Task Loading - unless a team's tasks are very carefully controlled, it is inevitable that as load increases individuals will become overloaded rather than the whole team. In order to cope with the variability of load it will be necessary to run the team at a loading level such that expected variations so not overload individuals. A consequence of this is that many team members may be bored and underloaded leading to ineffective individual performance.

   In a Group individuals can operate a higher load levels since transient peaks will be spread over a number of individuals.

d) Performance Monitoring - it is necessary to monitor performance both for training assessment and in order to optimise overall system effectiveness. A Group will generally be considered as a whole since individual performance is masked by others. In a Team both team and individual performance can be assessed.

e) Training - members of a team will be able to undertake a significant amount of individual training since inputs can be simulated. A Group will require significant whole whole group training in order to establish collaboration relationships.

f) Career Progression - because individuals are subsumed within the Group it will have adverse effects on the judgements of individual competence.

g) Interpersonal relationships - as a Group is dependent on individuals working closely together, interpersonal conflict would leave a serious effect relative to that in a Team.

From this list is can be seen that Teams and Groups both have advantages which must be assigned up in the decision to use one or the other.

It is of note that if a collection of men are initially brought together as Team then they may move towards a Group relationship. However, when a Group is convened directly it will have difficultly moving to a Team relationship.

CONCLUSIONS

It has been shown that is it possible to organise men so that they work with either a Team or Group relationship. Those relationships are important in the type of job the men are required to do and are also part of the effectiveness of the overall system. Where the relationships are not clearly defined we can expect inefficiences in system performance.

This inevitably leads to the situation that the organisation of men's working relationships cannot be left to chance following the design and implementation of a system. It is vital that the types of relationship to be implemented are included in the design phase so that job can be designed and the necessary facilities provided to support the men. Those facilities include not only the work station with its interfaces and communications but also the trainers and simulation that will ensure that competent men are assigned to the system.

ACKNOWLEDGEMENTS

This paper orginates from work carried out with the support of Procurement Executive, Ministry of Defefence by the C^3 Group of companies; the group consisted of Ferranti Computer Systems Ltd, PA Management Consultancy Ltd, Plessey Radar Ltd, Scicon Consultancy International Ltd, Software Sciences Ltd, and Systems Designers Ltd.

REFERENCES

(1) Hallan J. and Stanmers R.B. (1979); "The Optimum Distribution of Tasks among Operators in a Multiman-Machine System University of Astom.

(2) Briggs G.E. and Naylor JC (1964); Experiment on team training in a CIC - type task environment"; Naval Training Device Centre, Washington, NAVTRADEVCEN 1327-1.

(3) Klaus D.J. and Glaser R (1968); "Increasing team proficiency through training"; American Institute for Research, Pittsburg AIR-CI-6/68.

(4) Morgan P.D. (1982) "A total Systems Approach to Command System Design", 5th MIT/ONR Workshop on C^3 Systems, Monterey M.I.T.

FUNCTIONAL MAINTENANCE BY SUBORDINATE COMMANDER SIGNALING IN HIERARCHICAL COMMAND STRUCTURES

Stephen Kahne

Case Western Reserve University
Cleveland, Ohio 44106

Abstract. Subordinate commander signaling (SCS) in military C^3 systems is discussed. The command structure and information structure in C^3 systems are generally different and permit commanders to receive information from subordinate commanders as well as from other intelligence sources. A commander centered model of the C^3 system is used to illustrate the SCS concept.

A fundamental basis for analysis and design of C^3 systems is still lacking. This note suggests a commander-oriented model of the command hierarchy which may lead to a model appropriate for analysis. The Navy command system is substantially different from the general forms of hierarchies suggested in most large scale systems [1] efforts due to the strict chain of command nature of the military situation and the central role of the enemy. In this case Navy doctrine quite clearly describes the hierarchical relationship between commanders and suggests that the "nearest neighbor" concept is a central one in Navy command structures since it is expected that most of a commanders interaction with others in the hierarchical structure will be with his "nearest neighbors" in the structure. Thus his major command interactions are with commanders not more than one echelon removed from his. Although skip echelon command is not unheard of we will consider the case that information from lower echelons reaches a commander from the nearest lower echelon. The implication of this is that a local model of a commanders environment may provide substantially more information for him in this highly structured hierarchy than one would expect in a less structured situation. The model of a commander which postulates a detailed model of the "local" environment with a more vague model of the "distant" environment may well be suitable in the C^3 application [2]. It also implies that one should examine the limiting case in which the only information available to the commander is from his "nearest neighbors" - indeed such information sources which provide him information may be defined to be his nearest neighbors.

It is known that in certain cases of decentralized systems, nearest neighbor information may be adequate for a local commander to deduce important global properties. For example, algorithms have been developed to determine optimal pathways through networks when the information held by each network mode is only local [3].

The overall question being addressed is the maintenance of command system function when certain of the commanders or assets are not available. Thus system function is to be maintained when local functions are eliminated. This note deals with models of command structure and their related information structure.

Consider the command structure shown in Figure 1. The command structure and information structures are two entirely different properties of a military C^3 system. Figure 1 shows command structure only. Commanders receive information from several sources depending on the nature of the military situation and, in part, their personal style. A_{53}, the assets of C^5_{53} is explicitly shown for illustrative purposes. Indeed it is understood that each bottom level commander has assets which may be engaged in battle.

To properly describe the command hierarchy in Figure 1 it would be necessary to explicitly enumerate the assets of each commander. For example:

$$A_{41} = \{C^5_{51}, C^5_{52}, C^5_{53}, A_{53}\}$$

$$A_{33} = \{A_{41}, C^6_{42}\}$$

This nested collection of assets and commanders may be used to describe the command hierarchy [6]. This does describe the assets of a particular commander but does not indicate the information structure which provides him with data on which to base his commands. It should be expected that the command structure itself will provide some information to him and it is this part of his information structure that we are primarily interested in in this note. In particular we are interested in signalling within the command structure.

In this model each lower echelon commander has precisely one immediate superior. The notation provides structural information about a commanders place in the hierarchy. C^d_{ab} is the label of the b-th commander in the a-th echelon. The superscript defines the superior officer. By definition, the superior officer is in the (a-1)th echelon. If his label is $C^h_{(a-1)g}$, then $d = g + (a-1)$. Thus the superscript portion of the label is equal to the sum of the subscripts of the label of the immediate superior commander.

The task of a particular commander C^k_{ij} is to analyze the information received from various sources to determine what commands to give to $C^{i+j}_{i+1,p}$ for $p = 1, 2, \ldots$ In this sense the commanders are data processors and command parsers who must allocate to each nearest neighbor subordinate some portion of the task to be accomplished.

Assume that each commander has a known discrete finite repertoire of commands which he can issue to lower echelon commanders. This is done based on his perception of the state of his assets and those threats which to him appear relevant to the situation he is commanding. The question is how a commander learns about the environment in which he is operating.

We suggest that a phenomenon first reported in the economics literature is at work here: signalling. In C^3 the signalling may occur through a superior commanders observations. of his subordinates' commands: their response to his command from above.

Let us denote the command repertoire of C^k_{ij} as

$$X_{ij} = \{U_r(C^k_{ij}),\ r = 1, 2, \ldots, m_i\}$$

For notational convenience we have assumed that each commander in the i-th echelon has has m_i discrete controls in his repertoire. To account for inter-echelon differences several of the controls may be assumed null.

In the nested command structure postulated it is clear that each commander will have a different view of the overall battle situation. Lower echelon commanders may have detailed but narrow views of a battle while higher echelon commanders will have a broader more comprehensive view. There have been concerns expressed about how one defines the "state" of a battle or a military environment. The whole question of appropriate aggregation of variables [5] to form useful models of C^3 systems and battle environments has not been addressed. Various well defined aspects such as conditions of weapons, placement of personnel, supply inventory levels, etc. may be known to lowest echelon commanders. There is not need for higher echelon commanders to have direct knowledge about such details. There is too much data of too little value. The way this data is aggregated is by the choice of action of the commanders who are in closest proximity to these details. In other words the state of the system is aggregated into the control choices of the commanders in the command hierarchy.

It must be easier to accurately monitor friendly forces than the enemy. Therefore it would be useful to determine how effectively commands could be generated based on information gleaned from the actions of subordinate commanders - subordinate commander signalling [4] (SCS). The control actions of the subordinate commanders will, in part, result from inductive reasoning which would otherwise be difficult to model in a manner useful to the superior commanders.

Some information structures may be useful ones for eventual analysis. Such justification for considering artificial information structures is similar to analysis of linear systems as an initial attempt to examine nonlinear system behavior. Some ordering of information structures may lead to useful but approximate results for structures which eventually emerge when real data is available about in place C^3 system behavior

Little is known about subordinate commander signalling.

To illustrate the potential value of knowing subordinates commands as a signal about the state of the environment they are observing, consider that each commander is described by an ordinary set of differential equations and exists in a noise free environment:

$$C^k_{ij}:\quad \dot{\underline{x}}^k_{ij} = \underline{f}_{ij}(\underline{x}_{ij}, \underline{u}_{ij}, \underline{w}_{ij})$$

$$\underline{y}_{ij} = \underline{g}_{ij}(\underline{x}_{ij}, \underline{u}_{ij}, \underline{w}_{ij})$$

where $\underline{w}_{ij} = \underline{0}$, a vector of noise disturbance
$\underline{x}_{ij}$ is the "state" of C^k_{ij}
$\underline{u}_{ij}$ is the control applied to C_{ij}
$\underline{y}_{ij}$ is the C^k_{ij} output which serves as control input to some other commander determined by the command structure.

With no noise and weak assumptions on $\underline{f}_{ij}$, knowledge of $\underline{y}_{ij}$ at some initial time t_0 and $\underline{u}_{ij}$ for $t \geq t_0$ determines $\underline{x}_{ij}$ and $\underline{y}_{ij}$ which then serves to drive some other commander(s). If a superior commander "knew" $\underline{u}_{ij}$ he would know the complete state of his subordinate C^k_{ij} for $t_0 \geq 0$ and its output $\underline{y}_{ij}$.

In the presence of certain noise processes, possibly arising from enemy actions perfect knowledge of $\underline{u}_{ij}$ may not yield perfect knowledge of $\underline{x}_{ij}$ and $\underline{y}_{ij}$. It is not presently known under what conditions this knowledge of a subordinates controls will yield valuable information about the state of a commanders assets and the battle environment.

If analysis shows that this type of information structure is particularly useful for improving the quality of a commanders decisions, C^3 communications support systems should emphasize secure, reliable links for subordinate commander signaling. Clearly ordinary differential equations are not appropriate models for commanders. However if such descriptions can, in some sense, approximate commander behavior this discussion suggests that subordinate commander signals may indeed help a superior commander to more fully comprehend the state of the battle environment. Moreover, the knowledge gained occurs in a decentralized manner which has advantages of survivability in a hostile environment. It also makes use of required command communication networks but, of course, in a direction opposite to the command flow. Moreover it is consistent with requirements for control migration in the C^3 system [7].

CONCLUSION

It is suggested that subordinate commander signaling in hierarchical command structures may be a useful way for superior commanders to improve the quality of their knowledge about the state of a battle environment.

REFERENCES

1. J.D. Palmer, R. Saeks (eds.), The World of Large Scale Systems, IEEE Press, NY 1982.

2. R. R. Tenney, N.R. Sandell,"Structures for Distributed Decision Making", IEEE Transactions, Vol. SMC-11, No. 8, August 1981, pp. 517-27.

3. J.M. Abram, I.B. Rhodes, "Some Shortest Path Algorithms with Decentralized Information and Communication Requirements", IEEE Transactions, Vol. AC-27, No. 3, June 1982, pp. 570-582.

4. Y.C. Ho, M.P. Kastner, "Market Signaling: An Example of a Two-Person Decision Problem with Dynamic Information Structure", IEEE Transactions, Vol. AC-23, No. 2, April 1978, pp. 350-61.

5. N.R. Sandell, et al, "Survey of Decentralized Control Methods for Large Scale Systems", IEEE Transactions, Vol. AC-23, No. 2, April 1978, pp. 108-28.

6. J.S. Lawson, Jr., "Command Control as a Process," IEEE Control Systems Magazine, Vol. 1, No. 1, March 1981, pp. 5-11.

7. S. Kahne, "Control Migration: A Characteristic of C^3 Systems", IEEE Control Systems Magazine, Vol. 2, No. 3, September 1982.

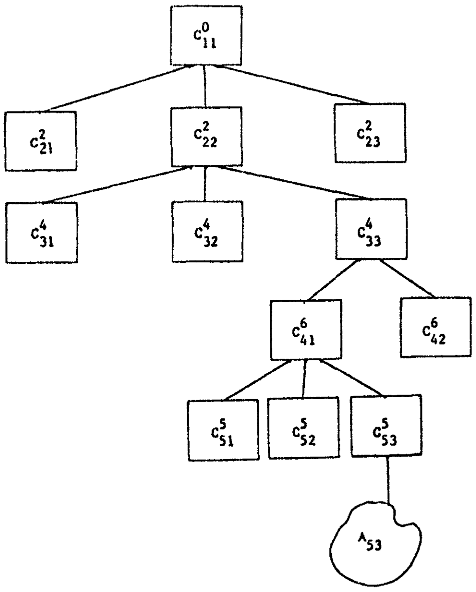


Figure 1. Example Command Structure

DECISION SUPPORT SYSTEMS FOR ORGANIZATIONAL DECISION MAKING

Edison Tse[1]
Department of Engineering-Economic Systems
Stanford University, Stanford, CA 94305

Richard M. Tong[2]
Advanced Information & Decision Systems
Mountain View, CA 94040

Abstract. In this paper we consider the problem of developing a theory for the assessment and design of decision support systems for organizational decision making. We begin by constructing a three-fold view of decision making, distinguishing between the decision process itself, the tasks performed in reaching a decision, and the role of the various individuals involved. We then consider the problem of measuring the benefits and the costs of introducing a decision support system. Finally, we discuss some wider design issues.

INTRODUCTION

The purpose of this paper is to construct a descriptive model of organizational decision making. This will allow us to develop an understanding of what is required for better design of systems that support this activity.

An organization is a complex entity with various types of interconnection between its members. There can be formal and informal communications, and authority structures that are rigid (hierarchical) nor flexible (heterarchical). The behavior of organizations has been studied extensively in the literature (Simon, 1976; Litterer, 1969) and we will not be directly concerned with this in the present paper. However, we will assume, throughout our discussion of organizational decision making, that the organization has a conventional hierarchical management structure.

A decision is an irrevocable allocation of resources, and any such action that affects the well-being of the members of the organization will be referred to as an organizational decision. Organizational decision making is then the process by which such decisions are made. Similarly, those members of the organization who are responsible, either individually or jointly, for the allocation of resources are called organizational decision makers. Further, this decision making may take place separately, or in conjunction with other members of the organization.

Another feature of organizational decision making is the distributed nature of expertise and knowledge. That is, no single member of the organization possesses all information that is relevant to all decision problems. However, for a particular problem there may be a sub-group within the organization that does indeed have this information. In such a case, decision makers can get access to the information, albeit with some cost. An important issue in developing our model of organizational decision making will be the identification of such sub-groups and the level of detail of information that resides within them.

A decision support system (DSS) is defined as any system whose purpose is to facilitate the decision process. According to this definition, a management information system (MIS) is an example of a DSS. While a variety of DSS`s have appeared in the literature (Alter, 1980; Keen & Scott Morton, 1978), there is no theory that allows us to decide on the appropriateness of a given DSS for a class of decision problems or a group of decision makers. The foundations of such a theory is the goal of the research reported in this paper.

In developing our ideas we have constructed three interacting views of the organizational decision making problem. These reflect the stages of the decision process itself, the tasks that need to be performed in executing this process, and the roles that the members of the organization can play in arriving at the decision. The key question in designing an effective DSS

---

[1] Research partially supported by ONR Contract No. N00014-75-C-0738.

[2] Research partially supported by RADC Contract No. F30602-81-C-0303.

thus becomes: What tool is to be used by whom at what stage of the decision process? If this question can be answered, then we can begin to develop the correct tools for the appropriate people, and then offer them the tools at the right stage of the decision process.

A PROCESS MODEL OF DECISION MAKING

Decision making is a multi-stage iterative process. Within it are sequences of "convergent" and "divergent" sub-processes that take the decision maker from the recognition that he has a decision problem to the implementation of his preferred choice. For example, in the conventional decision analysis paradigm (Raiffa, 1974) constructing a decision tree is a process of diverging, whereas the selection of just one of the alternatives through expected utility maximization is a process of converging.

There are two types of decision: the tactical, where we are concerned with slight modifications to existing plans, and the strategic, where we are concerned with replacing existing plans. Tactical decisions are often characterized by stimulus-response behavior and typically involve convergent processes such as estimation and feedback control. Strategic decisions, on the other hand, are much more complex and our primary concern in this paper.

Figure 1 shows a model of the strategic decision making process. This breakdown into seven distinct phases is not unique, but it does allow us to distinguish the different divergent and convergent processes. Each divergence provides a scope for the following convergence; each convergence provides a focus for the following divergence. Clearly, the specific details of this cannot be predetermined since they depend on both the problem and the individuals involved. We will, however, consider each stage briefly.

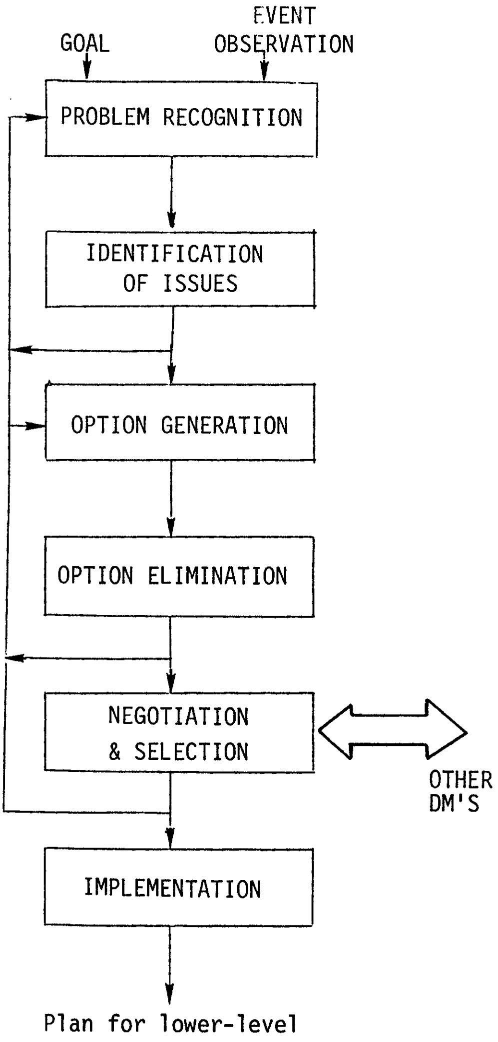


Fig. 1 A Process Model of Decision Making

Problem Recognition

This is a process of diverging. It is triggered by either a re-direction of goal from an authority at a higher level in the hierarchy, or by a drastic change in the decision environment. The scope of this process is determined by the degree to which the new goal differs from the old goal, or by the degree to which the new situation differs from the old. If the process is triggered by goal re-direction, then the divergence involves the listing of different goal representations. Whereas if the process is triggered by a situational change, then the divergence involves the generation of new hypotheses about the state of the world.

Identification of Issues

This is a process of converging. In it the main issues of the strategic decision problem are determined. The process is started with a search for information relevant to the problem. Typically this involves a set of questions that need to be answered: What are the important attributes?, How should these be combined to serve as a proxy for the perceived goal?, Should there be hard and soft constraints on these attributes and if so how should they be set to represent the decision maker's security and aspiration levels?

Option Generation

This is a process of diverging. It involves a statement of the possible alternatives and typically requires specialized domain knowledge. In routine problems this can be straightforward, requiring only the retreival of pre-stored options. In novel situations, however, this can be a time consuming and difficult process. In many problems, option generation is the key phase that determines the effectiveness of the overall decision process (Tong et al, 1982).

Option Elimination

This is a process of converging. In it the options generated in the previous stage are carefully evaluated and the least desirable ones pruned away. Situation specific knowledge is required to do this, and the output of the process will be a set of preferred options for a specific decision maker.

Negotiation and Selection

In this process all those decision makers who are involved in the particular organizational decision attempt to form a concensus. Actually this is a diverging process followed by a converging process. It is characterized by the addition of new concerns, and re-definition of goals and options as each decision maker attempts to justify his individual goal representation and preferences. Finally, though, one option is selected as the correct way of allocating the common resource.

Implementation

This is the divergent process by which the final decision is translated into a nominal plan and sets of detailed instructions for lower-level members of the organization.

A TASK MODEL OF DECISION MAKING

In this section we will describe the set of tasks that are performed in the different stages of the decision process. We call this a task model which itself defines a collection of appropriate decision making tools. From the process model (Fig. 1), we see that the tools in this toolbox can be divided into four categories. A mapping between the process and task model is shown in Fig. 2, where we indicate how the various stages require different tasks to be performed. A cross indicates a strong relationship; thus, for example, options generation requires knowledge acquisition and modeling.

Tools for Knowledge Acquisition

This task includes information retrieval from data bases as well as consultation with individuals who have specialized knowledge. Since knowledge that is relevant to a particular problem is distributed within the organization, there is a need for interactive tools that guide the user to appropriate sources of information as distinct from tools that help retrieve it. Knowledge acquisition is a task that is performed in three stages of the decision process: identification of issues, options generation and option elimination. However, the level of knowledge required is different for each stage. Thus during options generation broad overviews are needed, whereas during options elimination detailed cause and effect models are needed. This implies that several knowledge sources may be necessary. Notice too that divergent processes are typically concerned with locating information and convergent processes with extracting it.

Tools for Modeling

Modeling is the task of abstracting the knowledge acquired and constructing relationships between attributes and goals, and between actions and attributes. We call these goal modeling and systems modeling respectively. Modeling is a task that supports divergent processes; examples of tools being influence diagrams, analytic hierarchies (Saaty, 1980), and differential equations.

Tools for Analysis

This category includes all those tools that are normally used for analysis, such as optimization methods, heuristic search techniques, and estimation procedures. The result of using one of these tools will be a solution to some analysis problem.

Tools for Communication

These tools are essential if the decision makers are to arrive at a concensus. They allow interchange of concerns and perspectives. Such tools are also needed for communicating high-level decisions to lower-level decision makers.

A ROLE MODEL OF DECISION MAKING

In organizational decision making there are organization members other than the decision maker who may influence the actual choice made. We have developed a role model that helps us understand the interactions between these players, and we briefly describe the various groups as follows:

Stakeholders (SH). This refers to the group within the organization that stands to gain or lose as a consequence of the outcome of the decision that is made. Strictly, every member of the organization is a stakeholder, but practically we use this label to describe only those whose concerns are

STAGE TASK	PROBLEM RECOGNITION	ISSUE IDENTIFICATION	OPTION GENERATION	OPTION ELIMINATION	NEGOTIATION & SELECTION	IMPLEMENTATION
KNOWLEDGE ACQUISITION	X	X	X	X		
MODELLING	X		X		X	
ANALYSIS		X		X	X	
COMMUNICATION					X	X

Fig. 2 A Mapping from Tasks to Stages

directly sensitive to the outcome. A decision which is responsive to the stakeholders' needs is called a responsive decision.

Decision Makers (DM). This refers to the group within the organization that has the authority to allocate resources, and is therefore responsible for such actions. The job of the decision maker is to make responsive decisions.

Domain Experts (DE). This refers to the group within the organization that possesses knowledge in specific disciplines. We assume that domain experts acquire knowledge independently of the specific decision problem. Each member of this group will have specialized knowledge, but may not be able to communicate it to other members of the group.

Decision Support Staff (DS). The members of this group are also known as planners, or decision analysts, or systems experts. Their primary function is to support the decision maker in making the appropriate responsive decision. They help formulate the relevant issues, find the appropriate sources of problem relevant knowledge, integrate this knowledge to help generate options, analyze the options and form viable sets of options for consideration by the decision maker.

Actors (AC). These are the members of the organization who implement the plans chosen by the decision makers. Note that in a hierarchical organization all but the highest decision maker can take on the role of actor.

It is not necessary that these groups be disjoint. In fact in the usual situation an individual may play several roles simultaneously, or may alternate between roles. Such multiple role playing can introduce certain unique phenomena into organizational decision making. One instance is conflict of interest. For example, when individuals are members of DE as well as SH, they may provide biased knowledge that they hope may influence the decision in their favor. Another example is when individuals are members of DM as well as SH. In this case there may be a tendency for individuals to represent overall organizational goals in line with their personal desires. A second phenomenon is role confusion in which an individual may play an inappropriate role at a particular stage of the decision process. For example, a decision maker who is also a domain expert may spend too much effort in the latter role, when he/she ought to be making decisions based on the analysis an recommendations of the support staff.

A mapping from role to decision stage is shown in Fig. 3. As before, a cross in a cell indicates a strong relationship between stage and role.

TOWARDS A THEORY OF DECISION SUPPORT

The three models we have described give us a framework on which to build a theory of decision support. This must be in two parts: a theory of measurement and a theory of design. We need to be able to assess the effectiveness of a DSS before we can build an improved version. In this section we will consider some of the measurement questions.

STAGE / ROLE	PROBLEM RECOGNITION	ISSUE IDENTIFICATION	OPTION GENERATION	OPTION ELIMINATION	NEGOTIATION & SELECTION	IMPLEMENTATION
SH	X					
DM	X	X			X	X
DS		X	X	X		
DE		X	X	X		
AC						X

Fig. 3 A Mapping from Roles to Stages

First we would like to distinguish a tool from a DSS. Our concept of a tool is rather broad and we include anything that the decision maker may use. Thus a coin is a tool if tossed to resolve a simple binary choice. So is a complex computer simulation if its output helps determine the value of some important parameter in a decision problem. A DSS, on the other hand, is a collection of tools and a set of procedures for providing the right tools to the right people at the right stage of the decision process. A DSS thus consists of two parts: the toolbox and the user-interface. The user-interface is that part of the DSS with which the user interacts. It recognizes, or is told, which decision stage is being traversed and then helps the user select the right tool(s) for that stage. For example, if options generation is being performed, then appropriate tools are from the modeling category rather than the analysis category. The user-interface should also be aware of the role that the user is playing.

A measurement theory should concern itself with both the "benefits" and the "costs" of providing a DSS for organizational decision making. The measurement of benefit is a difficult topic but an important concept is compatibility. We distinguish two kinds: task-compatibility and role-compatibility. A DSS is task-incompatible with a decision problem if the tools do not match the tasks to be performed. The degree of task-compatibility may be related to the percentage of correct tools for the different stages of the decision process. A DSS is role-incompatible if it cannot provide tools appropriate to the roles played by the members of the organization involved in the decision. Once we have a measure for compatibility, then benefit will be conditional upon it. Benefit itself has two components: benefit to the individual and benefit to the organization. We are concerned with attributes such as speed of decision making, "correctness" of decisions, and confidence in the decision process and its outcomes. We should also expect that factors such as the frequency of the decision problem and the potential impact of the decision on the organization would be involved in arriving at a total benefit measure for the DSS.

The costs associates with a DSS include psychological factors as well as the obvious dollar amounts. It is important, for instance, that the introduction of a DSS does not cause a deterioration in performance as a result of it being unacceptable to those who are expected to work with it. Some of the costs are transient and associated with changes in working style that the DSS necessitates. Others are permanent and associated with maintaining and upgrading the DSS.

SOME DESIGN ISSUES

A design theory of DSS`s must be able to answer the following type of question: How elaborate should the toolbox be?, What specific tools should be included?, How sophisticated should the user-interface be?, Can we define the notion of an "optimal" DSS? Answers to these questions will, to a large extent, be organization specific. For an operational type of problem, where the issues are clear and the options limited, the emphasis should probably be on tools for analysis rather than modeling. For high-level planning problems, however, where issues are not clear and the option set is fuzzy, then modeling tools are more important. Indeed, as we move from the upper to the lower-levels of the organizational hierarchy then we would expect to change from tools that help divergent thinking to tools that help convergent thinking; that is from tasks that involve heuristics and creativity, to those that involve algorithms and analysis.

Similarly, the sophistication of the user-interface must depend on the user´s familiarity with the tools in the toolbox. The casual user will require more help from the system than someone who uses it regularly; the novice in a particular area will need more help than the expert in the same area.

In general, we see that the exact content and implementation of the DSS can be determined either by the problem or by the role. That is, a DSS may support a certain class of problems or it may support a specific role. The extent to which it could, or should, do both is an interesting open question.

Other issues that the design theory should address are the balance between "active" and "passive" decision support, the value of "adaptivity" versus "rigidity", and the relative importance of emphasizing "generality" over "specificity". A DSS can be passive in that it simply provides access to a variety of tools, or it can be active by making the user aware of potential flaws in use of the tools. A DSS can be rigid, providing a fixed set of tools and user interactions, or it can be adaptive to the changing needs of its users. A DSS can be specific to a particular problem/role, or it can be capable of supporting a class of problems/roles. All things being equal, we would expect an active, adaptive and general system to be the most powerful, but also the most costly. Whether this would also be the most effective is exactly the question our design theory should answer.

CONCLUSIONS

In this paper we have developed a three-fold view of organizational decision making which recognizes a distinction between models of the decision process, the decision tasks and individual roles. Our contention is that a Decision Support System should enhance the decision process by providing tools that match both tasks and roles. We have used our insight to lay the foundations of a two part theory of decision support; a theory of measurement and a theory of design. While our work is still in its infancy, we have identified a series of questions that need to be considered and believe that our approach will yield appropriate answers.

REFERENCES

Alter, S.L. (1980). Decision Support Systems. Addison-Wesley, Reading.

Keen, P.G. & Scott Morton, M.S. (1978). Decision Support Systems. Addison-Wesley, Reading.

Litterer, J.A. (1969). Organizations (2nd Edition). John Wiley, New York.

Raiffa, H. (1974). Decision Analysis: Choice Under Uncertainty. Addison-Wesley, Reading.

Saaty, T.L. (1980). The Analytic Hierarchy Process. McGraw-Hill, New York.

Simon, H.A. (1976) Administrative Behavior. The Free Press, New York.

Tong R.M., et al (1982) Options Generation Techniques for Command and Control. Tech. Report TR3012-1, AI&DS, Mountain View.

3020

AN APPROACH TOWARDS EVOLUTIONARY DEVELOPMENT OF COMMAND AND CONTROL SYSTEMS

Author: Edward J. Shanahan, Jr.
Georgia Institute of Technology
Engineering Experiment Station
Command and Control Branch
Atlanta, Georgia 30332
Telephone: (404) 894-3523

With contributions by: H. Bennett Teates

ABSTRACT

The Administration's new DoD acquisition policies of "controlled decentralization" and "pre-planned product improvement" form the basis for new approaches to the acquisition process and place the responsibility for these approaches in the hands of the Services. Whereas the need for experimentation, demonstration or design verification of new systems is widely accepted and practiced, there is a need for a method to exploit and more quickly transfer our information-systems technology to deployed operational combat systems. This paper traces this need from the DoD acquisition process and proposes the use of Advanced Experiment Demonstrations (AEDs) in a Use-Learn-Develop cycle within the Conceptual Phase of development. The paper then addresses an approach to the development of (AEDs) and cites the experience gained by the author in the development of AEDs.

INTRODUCTION

"Over the past decade, computers have come to play an important and significant part in improving the capability of our military forces to gather, process, and disseminate information. Successful applications have occurred most frequently where the computer was used as a control device tightly coupled to some physical process or weapon system. On the other hand, the use of computers to support tactical operations, strategic planning and the projection and evaluation of alternate courses of action have been far less successful and valuable than the designer had intended and the user had expected."(1)

One of the reasons for this failure is the inadequate forum for information-system technology transfer and constraints inherent to a time-consuming DoD acquisition process. Table 1 compares the development and acquisition of information systems with weapon systems.(2)

DEPARTMENT OF DEFENSE ACQUISITION

In the system development and acquisition process there are two principal agents: the combat developer and the material developer. The combat developer, acting on behalf of the eventual user, establishes system functional design requirements and develops a concept for the use of the proposed system. As the material developer translates the design requirements into engineering specifications and continues through the various stages of development, test and evaluation, and eventual production; the combat developer refines the doctrine and procedures for system employment, establishes manning levels and the training required for personnel to use the system, and by circumspection, if not directly, determines the number and skill levels of personnel needed.

A 1978 Defense Science Board Report(3) on this acquisition process found that since 1960, the "front end" period, from initial program conception to Advanced Development, had increased from less than two years to an average of five. Other conclusions in the report indicated that it was taking longer to obtain necessary decisions because of lack of information and increased levels of review.

In recognition of these deficiencies, the Reagan Administration has taken steps to make the process more efficient. Two themes of the revised acquisition process are "controlled decentralization" and "pre-planned product improvement" (P3I). Controlled decentralization is the delegation to the Services the recognition of a need and initial advancement of a potential solution, as well as all production, testing and deployment of the final product once Engineering Development is passed. Pre-planned product improvement (P3I) is the acquisition of a basic capability with incremental upgrades planned over the life-cycle of the system.

This latter change in policy is particularly important to the development and acquisition of information systems pertinent to resource management and command and control. First, it is in keeping with another Defense Science Board report, this one on command and control. The Board reported that... "the most important characteristic of command and control systems (SIC) is their need for adaptability to user needs and for their evolutionary change over time.(4) Second, the most expedient remedy to the disparity between the size and quality of our military forces as opposed to forces of the Warsaw Pact is the employment of our technology base on the battlefield in the areas of electronics, optics and digital computers. This base, now evolving new generations of capability every five years or less, cannot be exploited as long as the acquisition and deployment periods are longer than the research and development (R&D) period.

In summary, the current acquisition system, even with the changes of the new administration;

- Separates the user, the researcher and the developer-resulting in longer communication channels, misunderstandings, and inadequate or overdesigned systems.
- Penalizes the project manager (PM) for attempting to capture new, "risky" technology - thereby making the PM a very conservative and unimaginative developer.
- Provides no mechanism for early experimentation-where failure is also recognized as progress and where "technological risks" can be exposed.

A NEW APPROACH

It is the thesis of this paper that, for information system acquisition, the synthesis of "operational experience" with "available technology", the exisiting approach is inadequate. A new acquisition method is needed that provides a better user-developer relationship. Moreover, the new method should provide a means by which technology, residing in highly decentralized centers in DoD, contractor IR&D programs, and academic and industrial R&D, can be made useful. Since it can be useful only to the extent that the acquisition process is flexible enough to take quick advantage of opportunities as they arise, the method proposed is illustrated in Fig. 1.

USE-LEARN-DEVELOP

Figure 2 illustrates an approach to information system acquisition which proposes a modification of the current "front-end". It is designed to force a closer interaction between the operational and technological aspects of information systems research, evaluation, and acquisition. It is termed the "use-learn-develop" approach.

The Implementation Phase is in no way different from the current engineering development, production, and deployment phases of the existing acquisition process. However, the Conceptual Development Phase is different. The essential difference is the early, direct interaction of the researcher, developer and the user in an Advanced Experimental Demonstration (AED). The structure of this phase would generally consist of two steps.

Step 1, "concept formulation", consists of (1) analysis of the identified operational needs and deficiencies of existing capabilities; (2) an essessment of fiscal, timing, interoperability, standardization, and other constraints; and (3) a program plan. Advanced Experimental Demonstrations (AEDs)(5) and engineering and development facilities may be initiated during Step 1 if required to aid in the formulation of the initial concept.

Step 2, "use-learn-develop", consists of two major elements. The first element takes full advantage of one or both of the Engineering and Development Facility and the Advanced Experimental Demonstrations. The second element provides for the preparation of detailed development specifications.

Where AEDs comprise two elements:

1. An AED refines the requirement, assesses the technical approach, and validates the concept. The using command participates directly in this step. Maximum use is made of existing military and commercial hardware and software which is functionally acceptable to the using command and which may or may not be suitable for subsequent field evaluation trials.
2. AEDs transfer the system developed in the engineering and development facility to the using command for further evolution and evaluation in an operational environment. Evolution of the system is directed at tailoring the system to meet the identified need under the stresses of field operations.

The second element may result in a detailed definition of the information system requirements and specificatioins (including software requirements) for subsequent use by a program manager in the acquisition and deployment of the system. Conversely, it may result in failure or non-acceptance by the user. In either case, it is specific information which, provided to a program manager, makes his job easier, less "risky", and leads to the faster implementation of systems.

Phase II Implementation Phase, encompasses, to the extent required, modification of an existing system, full-scale engineering development, or production and development of the system for operational use. This structure may take one of four possible forms depending upon the state of hardware and software resulting from the Conceptural Phase. The four possible forms are:

1. Modifications - It is to be anticipated that some equipment will be militarized, but most will not and, therefore, the system used in the AED will have to be changed to account for more realistic military environments.

2. Deployment - In this form, the system used in the AED exists in either military or suitable commerical form, and the software has been developed. All that remains to be done is to deploy the hardware or the software to the using command(s).

3. Production and Deployment - In this form, suitable system hardware designs exist in either miliitary or commercial form, and the sfotware has been developed. What remains to be done is the purchase or production of sufficient quantities and deployment to the using command(s).

4. Engineering Development, Production, and Deployment - In this form, the conceptual phase was conducted with modified or brassboard equipment (either military or commercial) which is not operationally suitable for the intended application. Thus, full-scale engineering development is required. This form may also include revisions to the prototype software used in the Conceptual Phase.

THE PROCESS

AEDs are key to the success of the Use-Learn-Develop approach. The purpose of an AED is to demonstrate and evaluate a capability which has inference to needed military applications. It is through this controlled process that anticipated requirements can be investigated in a research environment. This action leads to the added re-assurance of program managers as they consider using advanced technology and often to the development of an interim capability suitable for fielding.

The manner by which these AEDs are formulated are as follows:

- Determination of research issues.
- Identification and formation of AEDs.
- Establish procedures for determining benefits of AEDs.
- Establish costs of AEDs.
- Interlink the AEDs into a growing capability.

A format was developed to assure understanding between all parties involved with the process. This format, given in Table 2, is highlighted by two main elements: purpose and lessons learned. Purpose describes the principle which will be demonstrated, while lessons learned indicate details from an operational and technical viewpoint. For example, the purpose may indicate that computer security in a distributed mode is to be examined; lessons learned may be the operational concept of user interaction with the system and technical considerations of software modules restricting and granting access to users with a variety of access levels and need-to-know.

Another element which appears to be mundane, yet experience has proven to be important, is the specification of the user. This paragraph requires specifically, by position and organizational level, who will be the users of the system. Normally, in military command and control systems, this will be an enlisted man with ranks in the E-4 to E-6 range or a junior officer (O-1 and O-2). These are the people who will have physical access to and will interact with the equipment. Often in development cycles, this requirement is not clearly defined, resulting in field grade officers at combat development commands establishing requirements for work performance for which they do not have an in-depth understanding. Specifying the user in an AED increases the importance of the user community's role in the "user-learn-develop" cycle. As they have the personnel assigned to their units who, by day-to-day work, understand the job which is being assisted by automation in the detail required. The role of enlisted personnel; namely using the equipment for the purpose of producing corrective guidance during the development cycle, should not be under-estimated.

APPLICATIONS

There are three applications of the "use-learn-development" cycle currently being advocated or employed in the development of Army systems. The initial establishment of the concept was done for an Army research agency. Details of the determination of research issues, formulation of AEDs, generation of cost data, and selection of appropriate demonstrations in a five-year plan can be found in the final report for that project(5).

Implementation of the concept is being accomplished by two Army agencies: a materiel developer (Communications-Electronics Command) and a user organization (US Army Forces Command.) Both are using the use-learn-develop approach to establish interim C2 capabilities for combat forces. In the case of the materiel developer, a data base management system is being established on military equipment in support of the Maneuver Control System and will evolve over time by the addition of operational (and technical) capabilities. The final report on the project provides more information

(6). US Army Forces Command has established a project called "MICROFIX," which uses off-the-shelf microcomputers in support of intelligence units. The initial effort provides the intelligence analyst with the capability of manipulating enemy activity and unit capabilities and of displaying this information textually and graphically. Personnel from intelligence units have been provided "hands-on" opportunities to use the system. Learning has been twofold: a refinement of the requirement statement based on feedback from the user, and the user has developed a better understanding of the capabilities of the technology available to support him.

POTENTIAL PROBLEM AREAS

Potential problem areas in using the AED and "use-learn-develop" approach are:

- subjectivity of user acceptance;
- loss of project thrust with personnel change;
- Hand over of project/equipment;
- formal testing methods; and
- maintenance as system grows.

The key to the "use-learn-develop" cycle is interaction between the end-user of the system, the combat developer, and the materiel developer. Often, this interaction takes the form of "this is good," "I don't think that this report in its present form is useful," and similar comments. Project directors must be careful in dealing with these subjective judgements in terms of the capability of the individuals who make the statements and the degree of their acceptance. While this may be a problem, the risk must be taken, as the ultimate test of systems is user acceptance.

Successful projects are directed by individuals who are dynamic, enthusiastic and are intellectually, technically and and emotionally capable of the required leadership. Transfers of the key personnel in the project will have greater impact than a project employing the traditional development approach.

The question of when does the project become a fielded system, when does a user community take charge from a developer, etc., is an important consideration to any evolutionary development effort. While advocating a dynamic, evolutionary approach, one must be aware that the environment in which this project exists is a large bureaucracy unaccustomed to such activity. The only answer is close coordination by all parties resulting from an understanding of the initial and subsequent AEDs.

Coupled with this bureaucracy problem is the requirement for formal testing of systems. While there have been major changes in philosophy at the higher echelons of the acquisition cycle, (i.e., within the Office of the Secretary of Defense) implementation of the evolutionary approach at the actual test level is not complete.

Finally, as a system evolves, maintenance in terms of hardware, software, and procedures grows rapidly. The need for configuration management in a "use-learn-develop" cycle is keen as change to meet identified user requirements is the thrust of the approach.

SUMMARY

In sum, the military information system community has seen a decrease in fielded technology and an increase in the amount of time required to provide support to the combat elements. The evolutionary, AED/use-learn-develop approach is a method of developing systems which will provide a better defined product with "state-of-the-art" technology. The gains of such an approach are: shortened concept-to-field-time, delivery involvement of the user in his system, evolutionary growth of capabilities tested in the field, and an interim capability constantly being improved.

REFERENCES

(1) Teates, H. Bennett, "The Role of Decision Support Systems in Command and Control" SIGNAL, Journal of the Armed Forces Communications and Electronics Association, Sept 1982.

(2) Greinke, Everett D. , Tactical C^3I Architecture," Presentation given at AFCEA Conference by Office of the Secretary of Defense, Office of Combat Support, 1979.

(3) "Report of the Acquisition Cycle Task Force," Defense Science Board, March 15, 1978.

(4) "Report of the Defense Science Board on C^2 System Management," OSDR&E, July 1978.

(5) Shanahan, Edward J. Jr., Final Report, GIT/EES, Project A-2526, Advanced Information System Research Project, June 1980, Contract DAAK70-79-D-0087.

(6) Shanahan, Edward J. Jr., Final Report, GIT/EES Project A-3075, Technical Support to U.S. Army Forces Command, September 1982. Contract F33657-80-G-0077 Item 1E01.

FUNCTIONS/CHARACTERISTICS	WEAPON SYSTEMS	INFORMATION SYSTEMS
REQUIREMENTS	THREAT AND NEEDS COMPARATIVELY WELL DEFINED	DIFFICULT TO DEFINE
SYSTEM FUNCTIONS	UNIQUE AND RELATIVELY SIMPLE	COMPLEX AND POORLY DEFINED
INTERACTION WITH SYSTEMS OF OTHER SERVICES/NATIONS	COMPARATIVELY LITTLE	VERY HIGH
TEST AND EVALUATION	RELATIVELY STRAIGHTFORWARD	COMPLEX WITH MULTIPLE INTERACTIONS. MAY BE ITERATIVE
QUANTIFICATION OF EFFECTIVENESS	METHODS ESTABLISHED	METHODS NEED DEVELOPMENT
SYSTEM EVOLUTION	FEW CHANGES. LITTLE IMPACT ON OTHER SYSTEMS	POTENTIALLY CONTINUOUS ADAPTATION INTERACTING SYSTEMS MUST BE CHANGED SIMULTANEOUSLY
IMPACT ON MILITARY COMMAND AND CONTROL PROCESS	LITTLE DIRECT EFFECT	VERY STRONG, MAY REVOLUTIONIZE MILITARY C^2 PROTOCOLS
SYSTEM PROGRAM MANAGEMENT OFFICE	FUNCTIONS WELL IN ISOLATION ONCE REQUIREMENTS ARE DEFINED	INTERACTIONS WITH OTHER SYSTEMS CAUSE DIFFICULTIES

TABLE 1 Comparison of Information Systems and Weapon Systems Development and Acquisition

AED FORMAT

TITLE:	Short phrases that provide the most concise description of the content of each AED.
ABSTRACT:	An explanation of the AED generally including background information, rationale for conduct of the test, and expected output.
Purpose:	A description of the principle that will be demonstrated.
Lessons Learned:	A description of the diverse areas into which the test will provide insight. The questions asked fall under the general heading, "What will we learn?" and include such thoughts as: • How the battlefield information system functions now; how it ought to function in the study time period; and how it will function in the study time period. • How to acomplish a particular subfunction better. • User performance characteristics interfacing with machine.
Environment:	The test environment in which the capability to be demonstrated will be used.
User:	The specific military operator or user. Generally the user is specified by position and organizational level or military ocupation specialty.
Equipment:	Generally described by function or specifically defined current or projected military hardware. A determination was made as to the basic equipment needed to conduct the test including equipment needed to simulate external functions. The number and type were described along with comments as to the intended use.
Assumptions:	Specific limits or conditions forecast for the AED that further defined level of effort or had a bearing on cost determination.
Special Requirements	Forecast needs such as government furnished equipment or other specific requirements that define the test parameters.
Research Issue:	The issue or issues that the specific SED would address or provide additional insight.

TABLE 2 AED Format Descriptions

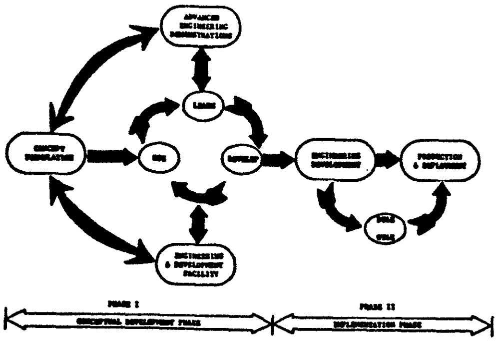


FIGURE 1. ADVANCED INFORMATION SYSTEM DEVELOPMENT.

CONVERGENCE AND ASYMPTOTIC AGREEMENT IN DISTRIBUTED DECISION PROBLEMS*

by

John N. Tsitsiklis*
Michael Athans*
Room 35-406
Laboratory for Information and Decision Systems
Massachusetts Institute of Technology
Cambridge, Massachusetts, 02139

ABSTRACT

We consider a distributed team decision problem in which different agents obtain from the environment different stochastic measurements, possibly at different random times, related to the same uncertain random vector. Each agent has the same objective function and prior probability distribution. We assume that each agent can compute an optimal tentative decision based upon his own observation and that these tentative decisions are communicated and received, possibly at random times, by a subset of other agents. Conditions for asymptotic convergence of each agent's decision sequence and asymptotic agreement of all agents' decisions are derived.

I. INTRODUCTION.

Consider the following situation: A set $\{1,\ldots,N\}$ of N agents possessing a common model of the world (same prior probabilities) and having the same cost function (common objective) want to make an optimal decision. Each agent bases his decision on a set of observations he has obtained and we allow these observations to be different for each agent. Given this setting, the decisions of the agents will be generally different. Aumann [4] has shown, however, that agreement is guaranteed in the following particular case: If the decision to be made is the evaluation of the posterior probability of some event and if all agents' posteriors are common knowledge, then all agents agree. (In Aumann's terminology, common knowledge of an event means that all agents know it, all agents know that all agents know it, and so on, ad infinitum.)

The situation where each agent's posterior is common knowledge is very unlikely, in general. On the other hand, if agreement is to be guaranteed, posteriors have to be common knowledge. The problem then becomes how to reach a state of agreement where decisions are common knowledge, starting from an initial state of disagreement.

Geanakoplos and Polemarchakis [7] and Borkar and Varaiya [6] gave the following natural solution to the above problem: Namely, agents start communicating to each other their tentative posteriors (or, in the formulation of [6] the conditional expectation of a fixed random variable) and then update their own posterior, taking into account the new information they have received. In the limit, each person's posterior converges (by the martingale convergence theorem) and assuming that "enough" communications have taken place, they all have to converge to a common limit.

The above results hold even when each agent obtains additional raw observations during the adjustment process and when the history of communications is itself random. Similar results were also proved for a detection problem [6].

A related – and much more general situation – is the subject of this paper; we assume that the agents are not just interested in obtaining an optimal estimate or a likelihood ratio, but their objective is to try to minimize some common cost function, given the available information. (Clearly, if each agent has a different cost function no agreement is possible even if each agent had identical information.) In this setting, we assume that agents communicate to each other tentative decisions (which initially will be different). That is, at any time, an agent computes an optimal decision given the information he possesses and communicates it to other agents. Whenever an agent receives such a message from another agent, his information essentially increases and he will, in general, update his own tentative decision, and so on. In the sequel we prove that the qualitative results obtained in [6], [7] for the estimation problem (convergence and asymptotic agreement) are also valid for the decision making problem for several, quite general, choices of the structure of the cost function. However, tentative decisions do not form a martingale sequence and a substantially different mathematical approach is required for the proofs. We point out that estimation problems are a special case of the decision problems studied in this paper, being equivalent to the minimization of the mean square error.

A drawback of the above setting is that each agent is assumed to have an infinite memory. We have implicitly assumed that the knowledge of an agent can only increase with time and, therefore, he has to remember the entire sequence of messages he has received in the past. There is also the implicit assumption that if an agent receives additional raw

---

* Research was supported by the Office of Naval Research under contract ONR/N00014-77-C-0532 (NR 041-519)

data from the environment, while the communication process is going on, these data are remembered forever. These assumptions are undesirable, especially if the agents are supposed to model humans, because limited memory is a fundamental component of the bounded rationality behavior of human decision makers [13]. We will therefore relax the infinite memory assumption and allow the agents to forget any portion of their past knowledge. We only constrain them to remember their most recent decision and the most recent message (tentative decision) coming from another agent. (For a particular class of communication protocols, we even allow them to forget their most recent decision.) We then obtain convergence results similar to those obtained for the unbounded memory model, although in a slightly weaker sense.

A particular problem of interest is one in which all random variables are jointly Gaussian and the cost is a quadratic function of an unknown state of the world and the decision. It was demonstrated in [6] that the common limit to which decisions converge (for the estimation problem) is actually the centralized estimate, i.e. the estimate that would be obtained if all agents were to communicate their detailed observations. We prove (section 4) that the same is true in the presence of memory limitations, provided that each agent never forgets his own raw observations. (That is, he may only forget past tentative decisions sent to him by other agents.) We indicate that for linear quadratic Gaussian (LQG) problems our scheme is essentially a decomposition algorithm for solving static linear estimation problems. As we point out in section 4, this scheme has certain appealing features: there is significant parallelism in the computations which matches nicely with the assumed distribution of the data; also, in the course of the algorithm, acceptable estimates are obtained much earlier than the time that would be needed to compute the optimal estimate by centralizing the information. These tentative estimates can be very useful whenever there are strict time limits within which certain decisions have to be made.

We also consider (section 5) a slightly different scheme in which each agent transmits his tentative decision to a coordinator. The latter evaluates a weighted average of the tentative decisions he has received and sends it back to all agents. We show that our results remain valid for this scheme as well and suggest an economic interpretation in which the coordinator can be viewed as some sort of market mechanism. We also show that making optimal tentative decisions corresponds to Nash strategies for a certain sequential game.

A weak point of the model is that not only each agent has the same prior information and knows the statistics of the other agents' observations but also has the same model of the probabilistic mechanism that generates inter-agent communications. In particular, if this is a deterministic mechanism, an agent must know the precise history of communications between any pair of other agents, a strong requirement. If it is a stochastic mechanism, then there are two possibilities: either the history of communications becomes commonly known on-line (at the expense of additional communications) or each agent will have to make probabilistic inferences about the communications between all other agents. These weaknesses disappear, however, if every tentative decision is broadcast simultaneously to all other agents, at each stage. In that case the history of communications is simple, commonly known and easy to remember. (This will be the case, for example, if a set of experts with the same objective teleconfer and take turns into suggesting what they believe to be the optimal decision.)

Finally, we point out the ways in which our scheme is different from other schemes for distributed decision making or computation: In team decision theory [9] each agent tries to behave optimally, while trying to anticipate the behavior of the other agents; the issue of who implements what component of the decision vector is very important; we are interested instead in consensus and in a common decision, quite independently of implementation issues. In many schemes for distributed computation [2,5] each agent specializes in updating only those components of the decision vector that have been assigned to him, whereas in our scheme each agent updates the entire decision vector.

Motivation.

There are many real world situations in which several agents (or processors) with different on-line information have to cooperate, combine their information and arrive at a common decision. Examples can be drawn from power, air traffic control and command and control systems. What distinguishes such situations is that:

(i) There are often rigid time limits within which preliminary or final decisions must be made.

(ii) There are often communications limitations, restricting the number and the nature of the messages that can be exchanged. This is particularly true in tactical command and control systems [3].

(iii) Conflict resolution procedures involving higher levels of the decision making hierarchy, are undesirable because they are likely to result in delays and tend to overload these higher levels.

The need arises for a scheme that leads to a final common decision (consensus) while taking into consideration the above limitations. While our scheme guarantees exact agreement after, possibly, an infinite number of exchanges of messages, in most situations approximate agreement is acceptable. Our scheme then, is appropriate whenever empirical or theoretical considerations indicate that the average number of communications required for approximate agreement satisfies the time constraints imposed by the actual situation.

A scheme that would involve the centralization of all data (by communicating them to a predetermined agent) is usually undesirable for the following fundamental reasons: It is often the case that too many data are available, which would overload communications channels; moreover, good decisions can often be made on the basis of aggregations of the initial data; furthermore, if an agent is a model of a human the plethora of data would saturate his short term memory. This implies that it is preferable to communicate aggregate data. Determining an optimal way to "aggregate" is not a well-posed problem. If constraints are placed on the number of bits to be transmitted, the problem becomes computationally intractable, even in very simple situations, where there is only a finite number of possible events and decisions [10]. On the other hand, if a message is allowed to be any real number, all data can be coded in a single message. Also, for any fixed aggregation protocol, an agent could slightly change his message and code all information in the least significant bits of his message. (This is reminiscent of decentralized control problems in which an agent may observe the decisions of other agents - the so-called control sharing pattern [1,12].) Any such trick is very sensitive to noise in the channel and is effectively just a more complicated way of centralizing information. Since centralization was deemed undesirable in the first place, any indirect way of centralizing should be also undesirable and explicitly prohibited.

The above discussion implies that a particular aggregation of the data should be chosen by means of some ad-hoc rule that guarantees that certain desirable characteristics are present. Unless some particular structure on the problem is assumed, the optimal tentative decision given an agent's information seems to be a very natural message that an agent could transmit and this is the reason that we have adopted such a framework in this paper.

II. MODEL FORMULATION.

In this section we present a mathematical formulation of the model informally described in the introduction. We start with the general assumptions and later proceed to the development of alternative specialized models to be considered (e.g. memory limitations, particular forms of the cost function etc.). As far as the description of the sequence of communications and updates goes, we basically adopt the model of Borkar and Varaiya [6] except that time is considered to be discrete. As in [6], events are timed with respect to a common, absolute clock. As far as notation is concerned, we will use subscripts to denote time and superscripts to denote agents.

We assume that we are given a set $\{1,\ldots,N\}$ of N agents, an underlying probability space $(\Omega,\mathcal{F},\mathcal{P})$ and a real valued cost function $c{:}\Omega\times U\mapsto R$, where U is the set of admissible values of the decision variable. It will be useful in the sequel to distinguish between elements of U and U-valued random variables. The letter v will be used to denote elements of U whereas u, w will be used to denote U-valued random variables (measurable functions from Ω to U).

Assumption 1: Either

(1.1): U is a finite set, or

(1.2): $U=R^n$, for some n.

Assumption 2: The cost function c is nonnegative and jointly measurable in (ω,v). Moreover, $E[c(v)]<\infty$, $\forall v\in U$. When assumption (1.2) holds, we assume that there exists a positive and measurable function $A{:}\Omega\mapsto R$ such that

$$A(\omega)\|v_1-v_2\|^2\leq\frac{1}{2}[c(\omega,v_1)+c(\omega,v_2)]-c\left(\omega,\frac{v_1+v_2}{2}\right),$$

$$\forall\omega\in\Omega,\quad\forall v_1,v_2\in U.\tag{1}$$

(Remark: If we fix v_1, v_2, $v_1\neq v_2$ and take expectations of both sides of (1), it follows that A is integrable.)

Inequality 1 implies that c is a strictly convex function of v and strict convexity holds in a uniform way, for any fixed $\omega\in\Omega$. It also follows that $c(\omega,v)$ is continuous for any $\omega\in\Omega$. This assumption is satisfied, in particular, if c is twice continuously differentiable in v and its Hessian is positive definite, uniformly in v, for any fixed $\omega\in\Omega$.

We may use the function A, defined in assumption 2, to define a new measure μ on $(\Omega,\mathcal{F})$ by

$$\mu(B)=\int_B A(\omega)\,d\mathcal{P}(\omega),\qquad B\in\mathcal{F}.\tag{2}$$

This measure will be used in section 3.

We now consider the generic situation facing agent i at some time n. Let $\mathcal{F}^i_n\subset\mathcal{F}$ be a σ-field of events describing the information possessed by agent i at time n. Because of assumption 2, the conditional expectation $E[c(v)|\mathcal{F}^i_n]$ exists (is finite), is $\mathcal{F}^i_n$-measurable and is uniquely determined up to a set of measure zero, for any fixed $v\in U$. Agent i then computes a tentative decision u^i_n that minimizes $E[c(v)|\mathcal{F}^i_n]$. The following Lemma (proved in [14]) states that u^i_n is well-defined and $\mathcal{F}^i_n$-measurable.

Lemma 1: Under assumptions 1.2, 2, there exists a $\mathcal{F}^i_n$-measurable random variable u^i_n, which is unique up to a set of measure zero, such that

$$E[c(u^i_n)|\mathcal{F}^i_n]\leq E[c(w)|\mathcal{F}^i_n],\qquad\text{almost surely,}\tag{3}$$

for any U-valued, $\mathcal{F}^i_n$-measurable random variable w. The same results are true, (except for uniqueness) under assumptions 1.1, 2.

We continue with a description of the process of communications between agents. When, at time n, agent i computes his tentative optimal decision u^i_n, he may communicate it to any other agent. (If u^i_n is not unique, a particular minimizing u^i_n is selected according to some commonly known rule.) Whether, when and to which agents u^i_n is to be sent is a random event whose statistics are described by $(\Omega,\mathcal{F},\mathcal{P})$. In particular, it may depend on the data possessed by agent i at time n. So, we implicitly allow the agents to influence the process of communications, although we do not require this influence to be optimal in any sense. This allows the possibility of signalling additional information, beyond that contained in u^i_n, by appropriately choosing when and to which agents to communicate. We allow the communication delays to be random but finite. We also assume that when an agent receives a message he knows the identity of the agent who sent it.

We now impose conditions on the number of messages to be communicated in the long run; these conditions are necessary for agreement to be guaranteed. Namely, we require that there is an indirect communication link from any agent to any other agent which is used an infinite number of times. This can be made precise as follows:

Let $A(i)$ be the set of all agents that send an infinite number of messages to agent i, with probability 1. Then, we make the following assumption:
Assumption 3: There is a sequence $m_1, \ldots, m_{k+1} = m_1$ of not necessarily distinct agents such that $m_i \in A(m_{i+1})$, $i = 1, 2, \ldots, k$. Each agent appears at least once in this sequence.

The main consequence of assumption 3, which will be repeatedly used, is the following: If $\{h^i{:}1, \ldots, N\}$ is a set of numbers such that $h^i \leq h^j$, $\forall j \in A(i)$, $\forall i$, then $h^i = h^j$, $\forall i, j$.

We continue with a more detailed specification of the operation of the agents. We introduce assumptions on the knowledge $\mathcal{F}^i_n$ which are directly related to the properties of the memory of agent i. An agent may receive (at any time) observations on the state of the world or receive tentative decisions (messages) of other agents. The knowledge of an agent at some time will be a subset (depending on the properties of his memory) of the total information he has received up to that time. We consider four alternative models of memory, formalized with the four assumptions that follow.

Let w^i_n be any message received by agent i at time n. Our most general assumption requires that w^i_n and u^i_{n-1} are remembered at time n:
Assumption 4: (Imperfect Memory) For all n, the σ-field $\mathcal{F}^i_n$ is such that u^i_{n-1} and w^i_n are $\mathcal{F}^i_n$-measurable.

Assumption 4 can be further weakened if some restrictions are imposed on the communications protocol:
Assumption 5: (Imperfect Memory) For each n there exists a set $I(n)$ of agents such that:
a) $u^i_{n-1} \in \mathcal{F}^j_n$, $\quad \forall i \in I(n-1)$, $\quad \forall j \in I(n)$
b) $\mathcal{F}^i_n = \mathcal{F}^i_{n-1}$, $\quad \forall i$ not in $I(n)$.

Intuitively, $I(n)$ is the set of agents that update their decision at time n. Assumption 5 is satisfied by the following two common communication protocols, provided that agent i may obtain additional observations only at times such that $i \in I(n)$:

Ring Protocol: $I(n) = \{k\}$, where k is the unique integer such that $1 \leq k \leq N$ and $k + mN = n$, for some integer m. Here, exactly one agent updates at any time instance and communicates his tentative decision to the next agent, and so on.
Star Protocol: $I(n) = \{1, \ldots, N-1\}$, if n is odd; $I(n) = \{N\}$, if n is even. Here all agents but the last one update simultaneously, communicate to the last agent who updates and communicates to all other agents and so on.
Assumption 6: (Own Data Remembered) Let G^i_n be the subfield of $\mathcal{F}$ describing all information that has been observed by agent i up to time n, except for the messages of other agents. We assume that $G^i_n \subset \mathcal{F}^i_n$.

With assumption 6, we allow the agents to forget the messages they received in the past, but they are restricted to remember all their past observations. In this case the total information available to all agents is preserved.
Assumption 7: (Perfect Memory) We let assumptions 4 and 6 hold and assume that $\mathcal{F}^i_n \subset \mathcal{F}^i_{n+1}$, $\forall i, n$.

Whenever assumption 7 holds, we will denote by $\mathcal{F}^i_\infty$ the smallest σ-field containing $\mathcal{F}^i_n$, for all n.

We conclude this section by defining a few special cases of particular interest:
(i) Estimation Problem: We are given a R^n-valued random vector x on $(\Omega, \mathcal{F}, \mathcal{P})$. The objective is to minimize the mean square error. Hence, the cost function is $c(v) = (x-v)^T(x-v)$, where T denotes transpose. It is easy to see that this is a particular case of a strictly convex function covered by assumption 2, with $A(\omega)$ being a constant.
(ii) Static Linear Quadratic Gaussian Decision Problem (LQG): Let x be an unknown random vector. Let the sequence of transmission and reception times be deterministic. We assume that the random variables observed by the agents are zero mean and, together with x, jointly normally distributed. We allow the total number of observations to be infinite. Let $U = R^n$. The objective is to fix v so as to minimize the expectation of the quadratic cost function $c(v) = v^TRv + x^TQv$, with $R > 0$. It follows that the optimal tentative decision of agent i at time n is $u^i_n = GE[x|\mathcal{F}^i_n] = E[Gx|\mathcal{F}^i_n]$, where G is a precomputable matrix. If we redefine the unknown vector x to be equal to Gx instead of x, we conclude that we may restrict to estimation problems, without loss of generality.
(iii) Finite Probability Spaces: Here we let Ω be a finite set. Then, there exist finitely many σ-fields of subsets of Ω. Strict convexity implies that for each σ-field $\mathcal{F}_o \subset \mathcal{F}$ and any $\omega \in \Omega$ of positive probability there exists a unique optimal tentative decision. This implies in turn that tentative decisions take values in some finite subset of U, with probability 1. We will therefore assume, without loss of generality, that U is a finite set.

III. CONVERGENCE AND AGREEMENT RESULTS.

In this section we state and discuss our main results. All proofs can be found in [14]. Assumptions 2 and 3 will be assumed throughout the rest of the paper and will not be explicitly mentioned in the statement of each theorem. We start with the least restrictive assumptions on memory:

Theorem 1: We assume that transmissions and receptions are deterministic, that communication delays are bounded and that the time between two consecutive transmissions from agent j to agent i (with $j \in A(i)$) is bounded. Then, under assumptions 1.2 (convex costs) and either assumption 4 or 5 (imperfect memory):
a) $\lim_{n\to\infty}(u^i_{n+1} - u^i_n) = 0$, in probability and in $L_2(\Omega, \mathcal{F}, \mu)$.

b) $\lim_{n-\infty}(u_n^i - u_n^j) = 0, \forall i, j$, in probability and in $L_2(\Omega, \mathfrak{F}, \mu)$.

Consider the following situation: At time zero, before any observations are obtained, the sequence of transmissions and receptions is selected in random, according to a statistical law which is independent from all observations to be obtained in the future and from $c(v)$, for any $v \in U$. In other words, communications do not carry any information relevant to the decision problem. other than the content of the message being communicated. Suppose that the sequence of communications that has been selected becomes known to all agents. From that point on, the situation is identical with that of deterministic communications. In fact, a moment's thought will show that it is sufficient for the history of communications to become commonly known as it occurs: agent i only needs to know, at time n, what communications have occured up to that time, so that he can interpret correctly the meaning of the messages he is receiving.

We can formalize these ideas as follows: We are given a product probability space $(\Omega \times \Omega^*, \mathfrak{F} \times \mathfrak{F}^*, \mathfrak{P} \times \mathfrak{P}^*)$ where $(\Omega, \mathfrak{F}, \mathfrak{P})$ describes the decision problem and where $(\Omega^*, \mathfrak{F}^*, \mathfrak{P}^*)$ describes the communications process. We assume that for each $\omega^* \in \Omega^*$, the resulting process of communications satisfies the assumptions of theorem 1. Then, note that each $\omega^* \in \Omega^*$, we obtain a distributed decision problem on $(\Omega, \mathfrak{F}, \mathfrak{P})$ with deterministic communications. In that case:

Theorem 2: Under assumptions 1.2, either 4 or 5, and independent, commonly known communications (as described above), $\lim_{n\to\infty}(u_{n+1}^i - u_n^i) = \lim_{n\to\infty}(u_n^i - u_n^j) = 0$, in probability with respect to $\mathfrak{P} \times \mathfrak{P}^*$.

Strictly speaking, Theorems 1 and 2 do not guarantee convergence of the decisions of each agent. Suppose, however, that the agents operate under the following rule: Fix some small $\gamma > 0$. Let the sequence of communications and updates of tentative decisions take place until $|u_n^i - u_n^j| < \gamma$, $\forall i, j$ (small disagreement) and $|u_{n+1}^i - u_n^i| < \gamma$, $\forall i$ (small foreseeable changes in tentative decisions). Then, we obtain:

Corollary 1: With the above rule and the assumptions of Theorems 1 or 2, the process terminates in finite time, with probability 1, for any $\gamma > 0$.

When Ω and U are finite, convergence and agreement are obtained after finitely many stages:

Theorem 3: If Ω and U are finite sets, if each agent communicates all the values of v that minimize $E[c(v)|\mathfrak{F}_n^i]$ and if assumption 4 holds, then there exists some positive integer M such that

$$u_M^i = u_M^j, \quad \forall i, j \quad \text{and} \quad u_{M+n}^i = u_M^i, \quad \forall i, \forall n, \forall \omega.$$

Strictly speaking, tentative decisions in the above theorem are not elements of U but subsets of U. This is to compensate for the possibility of non-uniqueness of optimal tentative decisions. The equalities appearing in Theorem 3 have to be interpreted, therefore, as equalities of sets.

We now assume that the agents have perfect memory. We obtain results similar to theorems 1 and 2 under much more relaxed assumptions on the communications process. Namely, we only need to assume the following:

Assumption 8: Let M_k^{ij} be the k-th message sent by agent j to agent i. We assume that when agent i receives M_k^{ij}, he knows that this is indeed the k-th message sent to him by agent j.
Remark: This assumption is trivially satisfied if messages arrive at exactly the same order as they are sent, with probability 1.

Theorem 4: Under assumptions 1.2 (convex costs), 7 (perfect memory) and 8, there exists a U-valued random variable u^* such that $\lim_{n\to\infty} u_n^i = u^*$, $\forall i$, in probability and in $L_2(\Omega, \mathfrak{F}, \mu)$.

For estimation problems ($u_n^i = E[x|\mathfrak{F}_n^i]$), theorem 4 can be slightly strengthened: [5, Theorem 2]

Theorem 5: For estimation problems, under the assumptions of theorem 4, convergence to u^* takes place with probability 1.

We now consider the case where U is finite but (unlike theorem 3) Ω is allowed to be infinite. Several complications may arise, all of them due to the fact that optimal decisions, given some information, are not guaranteed to be unique. We discuss these issues briefly, in order to motivate the next theorem.

Suppose that $U = \{v_1, v_2\}$. It is conceivable that $E[c(v_1)|\mathfrak{F}_n^i] - E[c(v_2)|\mathfrak{F}_n^i]$ is never zero and changes sign an infinite number of times, on a set of positive probability. In that case, the decisions of agent i do not convege. Even worse, it is conceivable that $E[c(v_1)|\mathfrak{F}_n^i] > E[c(v_2)|\mathfrak{F}_n^i]$ and $E[c(v_1)|\mathfrak{F}_n^j] < E[c(v_2)|\mathfrak{F}_n^j]$, for all n and for all ω in a set of positive probability, in which case agents i and j disagree forever. It is not hard to show that in both of the above cases $E[c(v_1)|\mathfrak{F}_\infty^i] = E[c(v_2)|\mathfrak{F}_\infty^i]$, on a set of positive probability and this non-uniqueness is the source of the pathology. The following theorem states that convergence and agreement are still obtained, provided that we explicitly exclude the possibility of non-uniqueness.

Theorem 6: Under assumptions 1.1 (finite U) and 7 (perfect memory) and if the random variable u^i that minimizes $E[c(w)]$ over all $\mathfrak{F}_\infty^i$-measurable random variables is unique up to a set of measure zero, for all i, then $\lim_{n\to\infty} u_n^i = u^i$, almost surely, and $u^i = u^j$, $\forall i, j$.

Although the preceding theorems guarantee that (under certain conditions) all agents will agree, nothing has been said concerning the particular decision to which all agents' decisions converge. In particular, it is not necessarily true, as one would be tempted to conjecture, that the limit decision is the optimal centralized solution (that is, the solution to be obtained if all agents were to communicate all their information). On the other hand, the centralized solution is reached for LQG problems, under the perfect memory assumption [6] and is also reached generically for an estimation problem on a finite probability space [7]. This issue will be touched again in the next section.

Robustness with respect to Communication Noise.

Schemes that centralize information by coding (e.g. by using the least significant bits of the allowed messages [1,12] tend to require high bandwidth and are sensitive to noise in the communication channel. In our scheme, although real numbers are being transmitted (infinite information content), the least significant bits are not as essential. As a result, the qualitative convergence properties of our scheme are retained even if communications of the tentative decisions are assumed to be noisy. We provide a proof of this fact for estimation problems, under the perfect memory assumption.

Suppose, as before, that at random times agent j communuicates his optimal tentative decision u_n^j. However, the message received by the other agents is $\hat{u}_n^j = u_n^j + q_n^j$, where q_n^j is a random vector representing the noise in the channel. For simplicity, we assume that the noise vectors are independent, identically distributed.

Theorem 7: Assume noisy communications (as described above). For estimation problems, under assumption 7 (perfect memory), there exists a U–valued random variable u^* such that $\lim_{n\to\infty} u_n^i = u^*$, $\forall i$, with probability 1.

IV. THE LINEAR QUADRATIC GAUSSIAN (LQG) MODEL.

In this section we specialize and strengthen some of our results by restricting to the Linear Quadratic Gaussian model described in section 2. (Recall that any such problem is equivalent to an estimation problem; therefore, $u_n^i = \hat{x}_i^n = E[x|\mathcal{F}_n^i]$, for some random vector x.) Theorems 1, 3 and 4 are applicable. Moreover, the results of [6] guarantee that, under assumption 7 (perfect memory), u_n^i converges to the optimal centralized estimate, given the information possessed by all agents. The following theorem states that the same is true under the weaker assumption 6.

Theorem 8: For the LQG problem, under the assumptions of Theorem 1 and assumption 6 (imperfect memory; own data remembered), $\lim_{n\to\infty} \hat{x}_n^i = \hat{x}$, in the mean square, where $\hat{x} = E[x|\mathcal{F}_\infty]$ and $\mathcal{F}_\infty$ is the smallest σ-field containing $\mathcal{F}_n^i$, for all i, n.

Note that theorem 8 is much stronger than theorem 1 which was proved for the general case of imperfect memory. We have here convergence to a limit solution which is also guaranteed to be the optimal centralized solution.

Our next result concerns the finite dimensional LQG problem in which the total number of observations is finite. Namely, the smallest σ-field containing $\mathcal{F}_n^i$ for all i, n is generated by a finite number of (jointly Gaussian) random variables. In that case, the centralized solution is going to be reached by all agents in a finite number of stages, provided that all agents have perfect memory.

Theorem 9: For the LQG problem with finitely many observations and under assumption 7 (perfect memory), the centralized solution is reached by all agents in a finite number of stages.

Theorems 8 and 9 imply that the scheme considered in this paper may be viewed as an algorithm for solving static linear estimation problems, an issue that we discuss below.

The intuitive argument behind Theorem 9 is the following: once an agent has received enough messages, he is able to infer exactly the values of the observations of the other agents (or of some appropriate linear combinations of these observations) and compute the centralized solution himself. So, communicating optimal tentative decisions is in this case just another way for communicating all information to all other agents. This scheme does not seem to have any particular advantages (in terms of communication and computation requirements) over the scheme where each agent communicates all his data directly.

However, the scheme of Theorem 8 (imperfect memory) seems to have some appealing features, as we indicate below. Suppose that we have a single processor who obtains a NM dimensional vector of observations. He then divides his observations into N M–dimensional vectors that will play the role of the agents of our scheme. Finally, the processor, instead of inverting the $NM \times NM$ covariance matrix to obtain the optimal estimate (which would require $O(N^3M^3)$ operations), he uses the scheme of Theorem 8. At each round there will be one inversion per block of data, that is $O(NM^3)$ operations per round. If, for example, an acceptable estimate is obtained after $O(N)$ rounds, the final objective will have been accomplished with a total of $O(N^2M^3)$ operations, which is one order of magnitude less than the usual algorithm. It is not hard to show that if the noises in observations belonging to different blocks are uncorrelated, agreement is obtained after two rounds only. Accordingly, if the noises in observations in different blocks are weakly correlated, we expect our scheme to be faster than the standard algorithm. We present below some numerical results that support the above statements. So, our scheme leads to a potentially advantageous decomposition algorithm for static linear estimation problems. (This algorithm has some conceptual similarities with those suggested in [8].)

We now discuss some issues related to the distributed implementation of the decomposition algorithm, where each block of data actually corresponds to a physically distinct agent (processor). For any i, n, $\hat{x}^i_n = a^i_n y$, where a^i_n is a row vector and y is the vector of all available observations. When agent j receives $\hat{x}^i_n$, he must also learn a^i_n, in order to be able to extract information from $\hat{x}^i_n$. There are two choices: Either a) agent j computes a^i_n, which may be done off-line, or, b) agent i transmits a^i_n to agent j. Which of the two should be done clearly depends on whether communications or computations are more costly. Whether one of the above two variations can be useful depends on the particularities of the actual situation and its inherent communication and computation limitations. More numerical experience is needed before a definite answer can be given.

Numerical Results.

Let x be an unknown scalar, zero mean, random variable to be estimated ($E[x^2] = 5$). Let $y_i = x + w_i$, ($i = 1, \ldots, 18$) be the observations. The noises w_i are assumed to be independent of x. (The covariance Σ_w of the noises was randomly generated.) We split the 18-dimensional observation vector into blocks of data (corresponding to distinct agents) and used the decomposition algorithm of Theorem 8. We employed the ring protocol and assumed that at each stage an agent only knows his own observations and the most recent message he received (Assumptions 5, 6).

Let M_i be the number of observations assigned to agent i. We considered two alternative decompositions: (i) $N = 2$, $M_1 = 10$, $M_2 = 8$; (ii) $N = 6$, $M_1 = \ldots = M_6 = 3$. We first executed the algorithm using the covariance Σ_w and, then, once more using the covariance $\Sigma_w + I$.

The results are presented in Figures 1, 2. The horizontal axis denotes stages (each stage corresponds to an update by some agent) and the vertical axis indicates the associated mean square error. The dotted horizontal line indicates the centralized mean square error. The curves $D1$ and $D2$ correspond to the first and second decomposition, respectively. As expected, convergence was much faster when the identity was added to the initial covariance; moreover the first decomposition converged much faster than the second.

To illustrate the merits of the decomposition algorithm we performed a rough count of operations. We only took matrix inversions into account, assuming that the inversion of a $M \times M$ matrix requires M^3 operations, which is accurate enough for our purposes. With this counting scheme, the centralized algorithm required 5832 operations. The points A, B in the graphs were reached after 4100, 1152 operations, respectively. This leads to the following conclusion: While the first decomposition needs very few stages to converge, it does not have any particular computational advantages. The second decomposition, however, leads to an estimate close to the optimal with much fewer operations than the centralized algorithm.

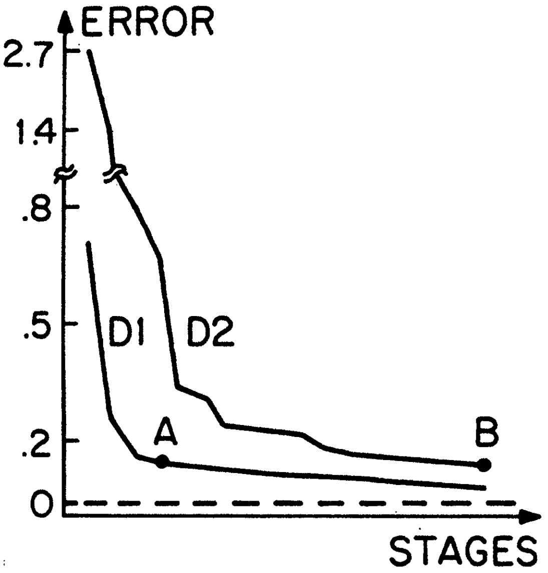


Figure 1: Mean Square errors of distributed algorithms with covariance Σ_w; dashed line represents performance of centralized algorithm.

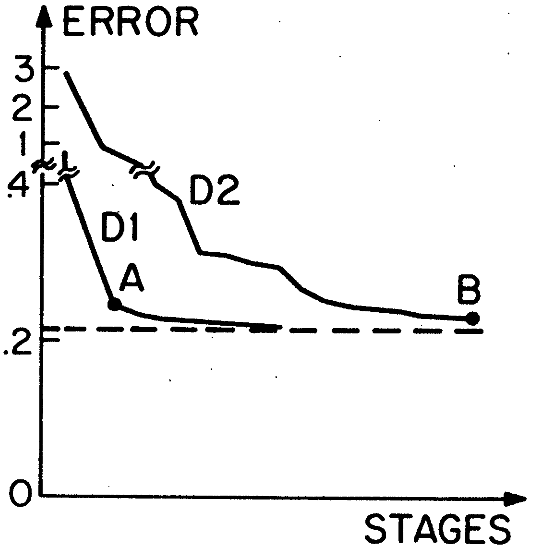


Figure 2: Mean Square errors of distributed algorithms with covariance $\Sigma_w + I$; dashed line represents performance of centralized algorithm.

V. A MODEL INVOLVING A COORDINATOR.

In the previous sections we had assumed that for any pair of agents i, j, agent i is allowed to communicate to j. In this section we assume that a particular agent (denoted by the superscript o) has special status and acts like a coordinator. The scheme we envisage is the following: At each instance of time n, agent i evaluates u^i_n which he communicates to the coordinator. The coordinator then combines u^1_n to u^N_n to produce a tentative decision u^o_n. We assume that the coordinator has no data of his own.) He then transmits u^o_n to all other agents which accordingly update their decisions. Were the coordinator to combine u^1_n to u^N_n "optimally", the above scheme would reduce to the one of the previous sections and our past results would apply. We assume, however, that the coordinator simply sets

$$u_n^o = \sum_{i=1}^{N} a^i u_n^i$$

where the coefficients a^i are deterministic, positive and $\sum_{i=1}^{N} a^i = 1$. The implicit behavioral assumptions are: (i) The coordinator has no memory and (ii) he need not have a good knowledge of the problem. He only knows how much he can rely on each of the other agents; this is reflected by his choice of the coefficients a^i which may be thought of as a "reliability index" for agent i in the eyes of the coordinator. We then obtain:

Theorem 10: The conclusions of theorems 1, 4, 8, remain true (under their respective assumptions) with the scheme introduced in this section.

The above scheme can be viewed as a framework for cooperation, where the coordinator simply aids the agents; or, for LQG problems, as another decomposition algorithm. It can be also interpreted, however, from an entirely different point of view: Suppose that the agents are selfish and independent individuals, faced with identical situations, possessing different information and having to make repetitive decisions. They can certainly benefit by observing past decisions of the other agents but assume that this is not possible. They are able, however, to observe a weighted average u_n^o of all decisions made in the last stage, which they take into account for their future actions. The motivation for such a model comes primarily from economics: Each agent is a buyer (or seller) in the same market and at each stage he obtains some aggregate information (e.g. the average price) on the transactions that were made in the previous stage. In this sense the "coordinator" simply represents a market mechanism. Our results state that, eventually, an informational equilibrium will be reached. Such an equilibrium has been studied by Radner [11] in a different setting. However, there was no demonstration of an adjustment process that could lead to such an equilibrium. Our scheme provides a model of rational behavior which, if followed by each agent, leads to equilibrium.

Moreover, within such a context (of selfish individuals confronted with identical situations) and for LQG problems with perfect memory, optimal tentative decisions constitute a set of strategies in Nash equilibrium for a certain game. (This is why optimal tentative decisions can be called "a model of rational behavior".) Let us define the game of interest more precisely.

Let y_n^i be a vector of jointly Gaussian random variables that generate G_n^i, the σ-algebra of events known to agent i at time n if he had received no messages. At each stage, agent i selects a decision u_n^i and incurs (but does not observe) a cost $a^n c(u_n^i)$, where $0 < a < 1$ and c is a quadratic cost function. Then, $u_n^o = \Sigma a^i u_n^i$ is formed and communicated to all agents. The total cost to agent i is $J^i = \Sigma_{n=1}^{\infty} a^n E[c(u_n^i)]$. A strategy for agent i is a sequence $\{\gamma_n^i,\ i = 1, 2, \ldots\}$ of measurable functions such that $u_n^i = \gamma_n^i(y_n^i, u_1^o, \ldots, u_{n-1}^o)$. Let $\Gamma = \{\Gamma^i : i = 1, \ldots, N\}$ denote the particular set of strategies where each agent at each stage plays his optimal tentative decision. (Note that these are linear strategies.) Then,

Theorem 11: Γ is a set of strategies in Nash equilibrium.

VI. CONCLUSIONS.

A set of agents with the same objective who start communicating to each other their tentative optimal decisions are guaranteed to agree in the limit. Under certain assumptions, this is true even if the decisions are received in the presence of noise and even if their memory is limited and they are allowed to forget some of their past knowledge. Moreover, they are guaranteed to converge to the optimal centralized decision for linear estimation problems, provided that they do not forget their own observations. This leads to a decomposition algorithm for static linear estimation problems. Similar results are obtained if the agents do not communicate directly but receive messages from a coordinator who evaluates a weighted average of all tentative decisions. In the latter framework, for LQG problems with perfect memory, optimal tentative decisions are Nash strategies for a certain game and admit an economic interpretation.

These results are valid when all agents share the same model of the world. The characterization of the behavior of agents with different models (perceptions) is an open problem.

ACKNOWLEDGEMENT

We would like to thank Professor R. Dudley of the Mathematics Department at MIT for suggesting the proof of Lemma 1.

VII. REFERENCES.

[1] M. Aoki, "On Decentralized Linear Stochastic Control Problems with Quadratic Cost", *IEEE Transactions on Automatic Control*, Vol. AC-18, pp.243-250, June 1973.

[2] K.J. Arrow and L. Hurwicz, "Decentralization and Computation in Resource Allocation", in *Essays in Economics and Econometrics*, R.W. Pfouts, ed., University of North Carolina Press, Chapel Hill, NC, 1960, pp. 34-104.

[3] M. Athans, "The Expert Team of Experts Approach to C^3 Organizations", *IEEE Control Systems Magazine*, Vol. 2, No. 3, Sept. 1982.

[4] R.J. Aumann, "Agreeing to Disagree", *The Annals of Statistics*, Vol. 4, No. 6, 1976, pp. 1236-1239.

[5] D.P. Bertsekas, "Distributed Dynamic Programming", *IEEE Transactions on Automatic Control*, Vol. AC-27, No. 3, pp. 610-616, June 1982.

[6] V. Borkar and P. Varaiya, "Asymptotic Agreement in Distributed Estimation", *IEEE Transactions on*

Automatic Control, Vol. AC–27, No. 3, pp. 650–655, June 1982.

[7] J.D. Geanakoplos and H.M. Polemarchakis, "We can't Disagree Forever", Institute for Mathematical Studies in the Social Sciences, Technical Report No. 277, Stanford University, Stanford, California, 1978.

[8] A.J. Laub and F.N. Bailey, "An Iterative Coordination Approach to Decentralized Decision Problems", *IEEE Transactions on Automatic Control,* Vol. AC–23, No. 6, pp. 1031–1036, December 1978.

[9] J. Marschak and R. Radner, *Economic Theory of Teams,* Yale University Press, New Haven, 1972.

[10] C.H. Papadimitriou and J.N. Tsitsiklis, "On the Complexity of Designing Distributed Protocols", submitted to *Information and Control,* August 1982.

[11] R. Radner, "Rational Expectations Equilibrium: Generic Existence and the Information Revealed by Prices", *Econometrica,* Vol. 47, No. 3, pp. 655–678, May 1979.

[12] N.R. Sandell, Jr. and M. Athans, "Solution of some Nonclassical LQG Stochastic Decision Problems", *IEEE Transactions on Automatic Control,* Vol. AC–19, No. 2, pp. 108–115, April 1974.

[13] H. A. Simon, *The Sciences of the Artificial,* Second Edition, The MIT Press, Cambridge, Mass., 1981.

[14] J. N. Tsitsiklis and M. Athans, "Convergence and Asymptotic Agreement in Distributed Decision Problems", *Proceedings of the 21st Conference on Decision and Control,* Orlando, Florida, December 1982.

A QUEUING METHODOLOGY FOR THE ANALYSIS AND MODELING OF AIR DEFENSE COMMAND AND CONTROL SYSTEMS

Dr. Martin G. Bello
Dr. John R. Delaney
Dr. Leslie C. Kramer
Dr. Michael Athans

ALPHATECH, Inc., 3 New England Executive Park, Burlington, Massachusetts 01803

Abstract. In this paper we present a queuing-theory based approach to the modeling and analysis of AD-C^2 problems. We describe a notional AD system whose functional components are modeled by particular canonical queuing forms. We outline the computational issues underlying the use of the queuing model for the quantification of various AD system MOP's and MOE's, and sketch avenues for continuing development of the queuing methodology.

SECTION 1. INTRODUCTION

In this paper we outline the development of a queuing-theory based methodology for the modeling and analysis of Air Defense (AD) C^2 systems. The growth of this methodology was spurred by recent interest in quantifying response times associated with particular AD-C^2/C^3 system cases, where response time is defined as the time between a targets entry into the geographic domain of the AD system and the first engagement. Our modeling approach is motivated by the following basic identifications between AD and queueing systems:

1. Air defense systems must process targets or groups of targets (identified as customers) by destroying them with defensive weapons (identified as servers).

2. Air defense requires the coordination of assets via a C^2/C^3 system. Messages, i.e., target track reports and target allocations (identified as customers), are generated and used by people and machines (identified as servers), and moved over communications links (identified as servers), and

3. Since the presumed goal of penetrators is to reach a given destination, there is a race between C^2/C^3 system and the target, implying limited available time for the completion of the various component C^2 processes (identified with the servicing of customers).

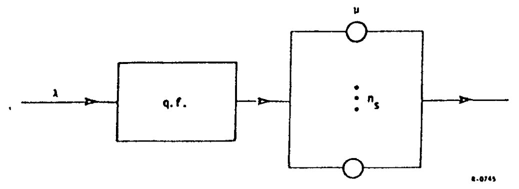
We will call the penetrating threats which arrive external to the AD system the exogenous customers, and the messages generated internal to the C^2/C^3 system will be termed endogenous customers. In rough terms, the queuing-theory based approach to modeling and analysis of AD-C^2, whose development we sketch in what follows, allows us to characterize the total processing delays associated with various component C^2 processes, i.e., the time expended waiting for scarce human and machine resources to become available as well as that for the execution of the process, as a function of the load on the AD system, i.e., the arrival rate of exogenous penetrators, and as constrained by the limited available processing times inherent in the target versus C^2/C^3 system race. Hence, we are ultimately able to derive the statistics of the time between target entry into the system and the completion of any specified component C^2 process, given that the specified process occurs, i.e., the statistics of various response times.

Our presentation is organized as follows. In Section 2 we define some notation and pictorial representations for the three classes of queuing models which we employ. Next, in Section 3 we describe the structure of a notional AD system, which we represent by the queuing model described in Section 4. The queuing model of Section 4 is built up from a set of C^2/C^3 model elements, which represent particular functional parts of the notional AD system, and are specified by canonical queuing forms defined in subsection 4.1. The interconnection geometry for these model components is delineated in subsection 4.2. Then in Section 5 we address the computational issues underlying the use of the described model by sketching in subsection 5.1 the algorithmic structure of the analysis of our queuing model, discussing in subsection 5.3 the controlling factors behind the calculation of the queuing model parameters. Finally, in Section 6 we identify future directions for continuing development of the queuing methodology.

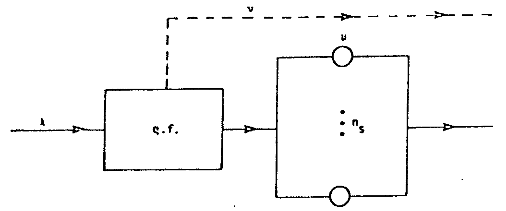
SECTION 2. QUEUING THEORY AND SOME QUEUING MODELS

In general, queuing theory considers the analysis of systems where "customers" arrive at some random time instants, wait in a queuing facility until a "server" is available, and then have their service requirement fulfilled, requiring some random processing time, before exiting from the system. Across the multiplicity of different applications of the theory, the notions of what constitute "customers" and "servers" are variable. In any of these applications queuing theory is employed to characterize both the statistics of the delays experienced by customers, and the statistics of resource utilization.

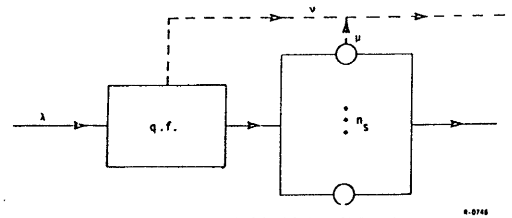
We next present some pictorial conventions for representing several types of queuing models which we will employ. In Figure 2-1 we depict three classes of queuing models. The parameters λ, μ, n_s are identified as the customer arrival rate, customer service rate, and the number of servers, respectively. In the case of each model we imagine that customers arrive at some random time instants with independent, exponentially distributed interarrival times (of mean value $1/\lambda$), and wait in some infinite storage queuing facility (q.f.) until one out of n_s servers is available to initiate processing (the n_s-circles correspond to service facilities). This customer "processing" is modeled as requiring an exponentially distributed time interval of mean value $1/\mu$. In the case of the standard multiserver queuing model a customer will wait indefinitely, or reside in the system indefinitely for his "processing" to be completed.



Standard Multiserver Queuing Model



Multiserver Queuing Model with Type I Reneging



Multiserver Queuing Model with Type II Reneging

Fig. 2-1. Classes of Multiserver Queuing Models

For our purposes, in order to capture the time-constraints implied by the target versus C^2/C^3 system race, we need to model situations where either the waiting or total system time associated with a customer is limited by some time, after which the customer leaves the system without completing "processing." Hence, we introduce the idea of multiserver queuing models with Type I or Type II reneging, pictured in Figure 2-1. Type I and Type II reneging correspond respectively to the cases when either the waiting, or total system time, is limited by some exponentially distributed reneging time, with mean value $1/\lambda$.

SECTION 3. THE STRUCTURE OF A NOTIONAL AD SYSTEM

We next describe the organizational structure of a notional AD system after which the queuing model which we present will be patterned. Let us imagine that the system is confined to some geographic domain, which is broken into a finite number of regions. With each region are associated specific surveillance and threat prosecution assets. For example, surveillance assets may consist of land based radars, radars associated with airborne interceptors, or some airborne early warning facility. Threat prosecution assets may consist of SAM launchers or flights of interceptors on combat air patrol. We assume that regional surveillance assets detect targets and generate track reports which are passed to some <u>Local Information Fusion Center</u>. Human operators at these local level information fusion centers select and forward track reports to a <u>Global Information Fusion Center</u>, which in turn delivers track reports to a <u>Global Air-Space Control Center</u>. We note that in some cases it may be possible for regional surveillance assets to pass reports directly to the <u>Global Information Fusion Center</u>. Human decision-makers at the <u>Global Air-Space Control Center</u> allocate the target for "processing" by one of several subsidiary <u>Local Air-Space Control Centers</u>, which in turn allocates the target for threat prosecution by the assets of one or several subsidiary geographic regions. These regional threat prosecution assets may be controlled locally by ACI/GCI operators, or SAM battery commanders, respectively. We diagram the flow of information and control for our notional AD system in Figure 3-1. The direction of the arrows in the diagram are not meant to preclude the existence of interactions between system elements at the same organizational level. For example, in our case for the sake of simplicity we assumed that a target which has been allocated for "processing" by the threat prosecution assets of a given region, and escapes from that region undestroyed, is automatically handed over for "processing" by the threat prosecution assets of any adjacent region. Similarly, we assume that a target which is reported in one region, but escapes from that region undestroyed, is automatically handed over to the surveillance operators associated with surveillance assets of any adjacent region.

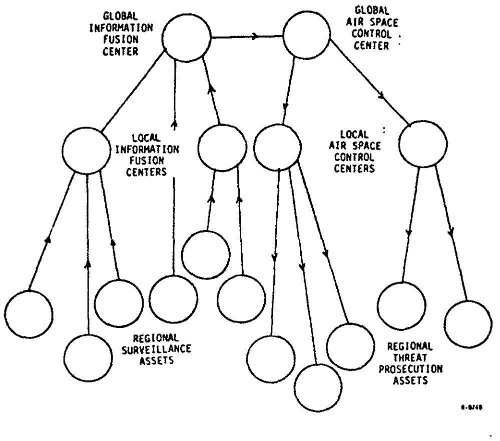


Fig. 3-1. Diagram for Notional AD System Organization

SECTION 4. THE QUEUING MODEL FOR REPRESENTING THE NOTIONAL AD SYSTEM

4.1 C^2/C^3 Model Components

We next define a set of C^2/C^3 model components, relating to the functions of surveillance, reporting, air-space control, and threat prosecution, which directly mirror the organizational elements defined for the notional AD system of Section 3. We will use a particular shorthand notation to represent these various model components, which we define in what follows. On the surveillance and reporting side of the model, we will let S_i^o denote a model for the i-th region's aggregate target detection and track report generation capability. We also let S_j^1 and S^2 denote models for the i-th local and the global information fusion centers, respectively On the air-space management and threat prosecution side of our model we let C_i^o denote a model for the i-th region's aggregate threat prosecution capability. In addition, we let C_j^1 and C^2 denote models for the i-th local and global air-space control centers, respectively.

We now proceed to identify the queuing forms which we have employed to represent the above-defined model components. For each of the models we have assumed exponentially distributed interarrival times (Poisson arrivals) and service times. Our representations for the S_j^1, S^2, C_k^1, and C^2 model components are each of the form of the standard multiserver queuing model depicted in Figure 2-1. In the case of S_j^1 and S^2, λ is identified with an arrival rate of target track reports (endogenous customers) and $1/\mu$ corresponds to the average time required for selection and forwarding of a single track report. In the case of C^2, λ is identified with an arrival rate of first track reports (endogenous customers) and $1/\mu$ corresponds to the average decisionmaking time required at the Global Air-Space Control Center for allocating the target to an appropriate local air-space control facility. Hence, the output process from C^2 is interpreted as a flow of target allocations (endogenous customers). Similarly, in the case of C_j^1, λ is identified with an arrival rate of target allocations (endogenous customers) and $1/\mu$ is interpreted as the average decisionmaking time required at a local air-space control facility to allocate a target for threat prosecution by the resources of an appropriate region. In the case of each of the models S_j^1 and S^2, C_k^1 and C^2, n_s corresponds to the number of simultaneous track report selection and forwarding or target allocation operations, respectively, that may be performed at the information fusion and air-space control centers, respectively.

We next describe our queuing representation for S_j^o, pictured in Figure 4-1 which corresponds to the concatenation of an infinite server facility with a q-server facility. The infinite server facility represents the delay between a target's entry into the i-th region and its detection by either ground- or air-based surveillance units, and hence $1/\mu_d$ may be interpreted as the average detection delay. The q-server facility represents the operation of track report generation, and hence $1/\mu_r$ corresponds to the average time required to complete a single report. The parameter λ represents the arrival rate of as yet unreported targets (exogenous customers) into the i-th region, λ_x corresponds to the arrival rate of targets that were reported first in some earlier region, but are either unallocated or have leaked after being unsuccessfully "processed." We assume that information about the targets in the λ_x stream has been handed over from other surveillance and reporting units, so that the model does not associate the same detection delay with these targets as for those in the λ stream. Finally, we note that we have used the idea of reneging in our representation of S_j^o. Reneging from the infinite-server facility corresponds to targets which pass through the region undetected, and therefore $1/\nu$ represents the average transit time through the i-th region. Similarly, $1/\nu_r$ represents the residual time available to initiate a track report for targets that have been detected but were previously unreported. We have used reneging from the queue, i.e., Type I reneging, in the representation of S_j^o to model the saturation effect on the assumed q tracking operators produced by a large number of nearly simultaneous detections. Hence, the pathways for reneging from S_j^o represent flows of penetrators (exogenous customers), while the output from the q-server facility corresponds to flows of track reports (endogenous customers).

The representation of our final modeling component, C_i^o, varies depending on whether the i-th regional defenses correspond to either SAMs or AIs which "process" single targets, or to flights of AIs which "process" groups of targets, where the group size is less than or equal to the number of AIs per flight. In Figure 4-2 we dipict our representation of the earlier case. The parameter λ represents the total arrival rate of penetrators allocated for "processing" by the i-th region's defenses. The quantity $1/\mu_{en}$ corresponds to the average time required for a single AI or SAM versus target engagement, p_s corresponds to the probability of success of a single engagement, and n_s corresponds to the number of simultaneous engagements that can occur. The motivation behind the choice of $(\mu_{en}p_s)$ as a service parameter is that if we assume the independence of the outcome of successive engagements, it requires an average of $(1/p_s)$ engagements for target destruction, and hence an average time interval $1/p_s\mu_{en})$. The type II reneging, i.e., reneging from both the queue and server, models both the fact that targets may escape without weapons commitment having occurred, or due to unsuccessful attack by the defense, i.e., due to incomplete or to unsuccessful "processing." The quantity ν^{-1} represents the average time-interval over which successful threat prosecution by the resources of the i-th region must occur. We finally observe that the exogenous customers (penetrators) which complete "service" disappear, i.e., it is only the leakers that appear as C_i^0 system outputs.

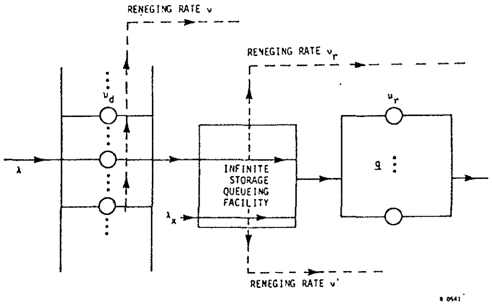


Fig. 4-1. Queuing Model Form for Representing S_j^0.

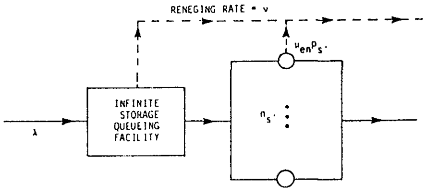


Fig. 4-2. Queuing Model Form for Representing C_i^0 for AIs or SAMs "Processing" Single Targets.

In the case where the i-th region's defenses corresponds to flights of AIs, we employ the representation of C_i^0 in Figure 4-3, where n_s is interpreted here as the number of flights that can simultaneously conduct engagements, $n_{AI/FL}$ denotes the number of AIs per flight (or more loosely the maximum number of simultaneous engagements that may be associated with a single flight), and n_{tj} denotes the number of targets currently engaged by the j-th flight. The parameters λ, μ_{en}, and p_s have the same interpretation as in Figure 4-2. For the j-th flight we imagine that roughly $((n_{AI/FL})/n_{tj})$ AI's are assigned to each target, and hence the probability of successful processing for a single <u>flight versus target</u> interaction varies approximately as

$$[1-(1-p_s)^{(n_{AI/FL}/n_{tj})}].$$

Therefore, following the same reasoning as in the model of Figure 4-2, the service rate parameter associated with the j-th flight is chosen as

$$[\mu_{en}[1-(1-p_s)^{(n_{AI/FL})/n_{tj})}]).$$

Since the service parameters associated with the different flights depend on the n_{tj}'s, we say that the service time distribution is state-dependent, i.e., it depends on the current "state" of the queuing model.

4.2 Interconnection Geometry for C^2/C^3 model Components

Having detailed the structure of the queuing systems which we employ to represent S_i^0, S_j^1, S^2, C_i^0, C_j^1, and C^2, we next describe the manner in which these models are interconnected by pathways for the flow of both exogenous and endogenous customers. We first note that in terms of our queuing model a raid is described by a set of penetration routes, i.e., a set of sequences of regional traversals, together with the exogenous penetrator arrival rates ν_ℓ associated with each route, and in addition, the reneging parameters ν_i associated with each region where the quantity ν_i^{-1} corresponds to the average transit time for a penetrator across the i-th region. In Figure 4-4 we diagram the surveillance and reporting geometry intrinsic to our model for a hypothetical fifteen region AD system. Superimposed on our diagram for the flow pattern of endogenous target track reports, we sketch in dotted lines a single hypothetical exogenous penetrator route. (Many such routes are included in our overall model.) We note that each region which a target traverses offers the opportunity for the generation of a new target track report, which is forwarded ultimately to S^2 possibly passing through an intermediate S_j^1.

While Figure 4-4 displays the topology of the surveillance and reporting side of a model for a hypothetical fifteen region AD system, in Figure 4-5 we depict the interconnections between model elements representing the airspace management and threat prosecution functions. We note that the processing load at C^2 is modeled as being derived from first reports of targets. The processing load at at a given C_j^1 is modeled as being derived from the flow of targets that have been reported and had a corresponding track report processed by C^2 before they exit from geographic domain of the regions for which C_j^1 provides airspace control. Finally, the processing load at a given C_i^0 is modeled as being derived from the flow targets that are reported and processed by both C^2 and an appropriate C_j^1, before they exit from the i-th region, combined with the flow of targets that leak into the i-th region after being unsuccessfully processed or unsuccessfully allocated for processing in some other region.

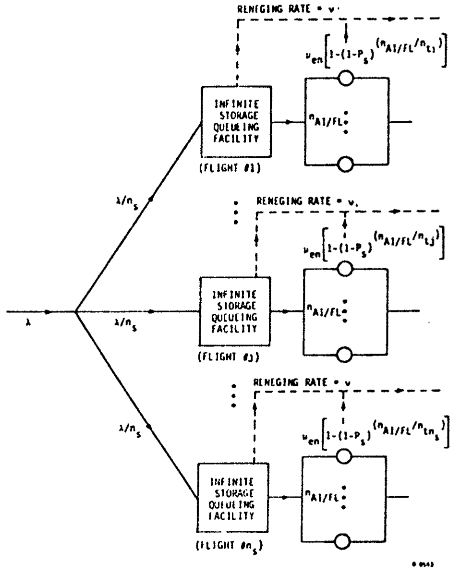


Fig. 4-3. Queuing Model Form for Representing C_i^0 for Flights of AIs "Processing" Target Groups.

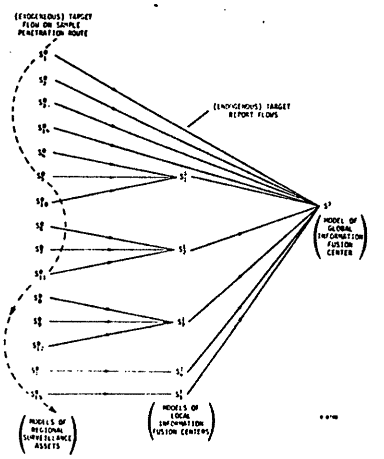


Fig. 4-4. Surveillance and Reporting Geometry for Representation of a Hypothetical Fifteen-Region AD System.

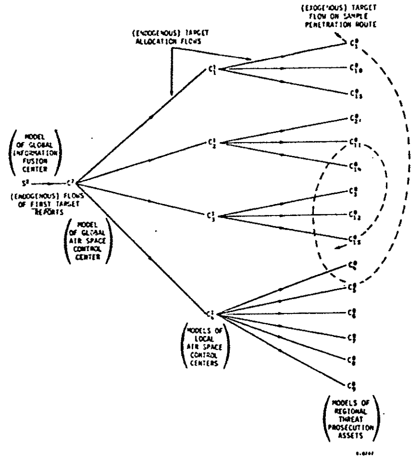


Fig. 4-5. Airspace Control and Threat Prosecution Geometry for a Hypothetical Fifteen-Region AD System.

SECTION 5. COMPUTATIONAL ISSUES

5.1 Algorithmic Structure of Model Analysis

Having described the manner in which the various model components interact to form a queuing model for representing our notional AD system, we next give an overview of the algorithmic structure of our analytic implementation of the model. Similar to the analysis of the response of an electric circuit, in the analysis of queuing models we must distinguish between the transient and steady-state behavior of the system. In our analysis of the described model for the notional AD system, we focus on the steady-state behavior of the system statistics. This approach is valid since we expect that the length of time required for a penetrator to traverse the geographic domain of the system and the time intervals associated with various C^2/C^3 processes (target detection, track report generation, target allocation, and threat prosecution) are small relative to the total duration of a raid. Hence the transient period of the system is short relative to the duration of a raid and the steady-state behavior is a valid description of the system behavior.

Just as the steady-state transform analysis of the response of electrical networks involves the solution of algebraic equations for the magnitudes and phases of voltages or currents, the steady-state analysis of a network of queues involves the solution of algebraic equations for the customer flow variables, i.e., the input arrival rates to different queues. The structure of our problem is such that we can specify the system statistics (i.e., various queuing probabilities and waiting and system time distributions) for any of the queuing forms associated with our model components S_i^0, S_j^1, S^2, C^2, C_j^1, and C_i^0 by the knowledge of three sets of flow variables, which we define in what follows.

Let us define the vector $\vec{\lambda}_{tu}$ as

$$\vec{\lambda}_{tu} \triangleq \begin{pmatrix} \lambda_{tu1} \\ \cdot \\ \cdot \\ \cdot \\ \lambda_{tu15} \end{pmatrix} \tag{5-1}$$

where λ_{tui} denotes the arrival rate of as yet <u>unreported</u> targets into S_i^0. Similarly, we define the vector $\vec{\lambda}_x$ as

$$\vec{\lambda}_{x} \triangleq \begin{pmatrix} \lambda_{x1} \\ \cdot \\ \cdot \\ \cdot \\ \lambda_{x15} \end{pmatrix} \tag{5-2}$$

where λ_{xj} denotes the arrival rate of either undestroyed or unallocated targets into S_i^0 that have been <u>first</u> reported earlier. finally, we define $\vec{\lambda}_t$ as

$$\vec{\lambda}_{t} \triangleq \begin{pmatrix} \lambda_{t1} \\ \cdot \\ \cdot \\ \cdot \\ \lambda_{t15} \end{pmatrix} \tag{5-3}$$

where λ_{ti} denotes the total arrival rate of targets that are allocated for processing by C_i^0, i.e., the sum of the target flows first allocated for processing at C_i^0 as well as those that leak into C_i^0 after being unsuccessfully processed earlier. By using the structure imposed by both the C^2/C^3 versus target race and the geometry of the raid as well as the geometry of the interconnections of both surveillance and control modeling elements, and by doing a queuing theory analysis of the input-output customer flow characteristics associated with each of our model components, we can obtain a system of nonlinear, simultaneous algebraic equations satisfied by the flow vectors $\vec{\lambda}_{tu}$, $\vec{\lambda}_x$, and $\vec{\lambda}_t$, of the following form:

$$\vec{\lambda}_{tu} = F_s(\vec{\lambda}_{tu},\vec{\lambda}_x) \ , \tag{5-4}$$

$$\vec{\lambda}_{t} = F_c(\vec{\lambda}_{t},\vec{\lambda}_{tu},\vec{\lambda}_x) \ , \tag{5-5}$$

and

$$\vec{\lambda}_{x} = F_{sc}(\vec{\lambda}_{t},\vec{\lambda}_{tu},\vec{\lambda}_x) \ , \tag{5-6}$$

We note that $F_S(.,.)$, $F_C(.,.,.)$, and $F_{SC}(.,.,.)$ are meant to denote vector-valued functions of vector arguments. Given the solution of the system of equations described by Eqs. 5-4 through 5-6, and employing queuing theory, the topology we specified for the interconnections of our model components, and the constraints implied by the C^2/C^3 system versus target race, we can derive the system statistics associated with all our defined model components, which may then be identified with the statistical behavior of the actual AD-C^2/C^3 system.

It is convenient to implement the solution of Eqs. 5-4 through 5-6, and hence the queuing analysis of the overall model, in an iterative fashion. We begin with a starting value for $\vec{\lambda}_x$ of all zero components, solve Eq. 5-4 for $\vec{\lambda}_{tu}$, use the result to solve for $\vec{\lambda}_t$ in Eq. 5-5, and ultimately recompute $\vec{\lambda}_x$ using Eq. 5-6, repating the cycle until some desired convergence is achieved. In the process of this iteration we use the intermediate $\vec{\lambda}_{tu}$'s, $\vec{\lambda}_t$'s, and $\vec{\lambda}_x$'s to compute the system statistics associated with each of our model components, i.e., the S_i^0's, S_j^1's, S^2, C^2, C_j^1's, and C_i^0's.

5.2 Calculation of System MOE's and MOP's

We begin our discussion of the calculation of system performance and effectiveness measures by noting that many of the "outputs" of the analysis of our queuing model are directly interpretable as either MOP's or MOE's. These "outputs" are the system statistics and customer arrival rates associated with the surveillance model components S_i^0, S_j^1, and S^2, and the air-space control and threat prosecution model components C^2, C_j^1, and C_i^0. By system statistics we mean any of the various probabilistic characterizations of system behavior, i.e., the average number of customers in the queue or in service, the queuing probability, the average total system time, and the probability of service completion for a queuing model with reneging. When these system statistics are interpreted in the air defense setting as MOP's and MOE's, they may be grouped into three classes: reaction time statistics, workload or resource utilization statistics, and leakage statistics. We proceed in what follows to present examples of each type of statistic that are "outputs" of our queuing analysis.

We first note that the average system times associated with any of our model components (i.e., S_i^0, S_j^1, S^2, C^2, C_j^1, and C_i^0) are identified with average reaction times. For example, the average system time at C_i^0, conditioned on service completion, corresponds to the average delay between the allocation of a target for "processing" by the defensive assets of a given region, and the target's destruction, given that destruction occurs. The average system time at C^2 may be identified with the average delay between the arrival of a new target track report at the Global Air-Space Control Center and the delivery of a target allocation to an appropriate Local Air-Space Control Center.

Next we observe that customer population statistics, e.g., the average number of customers in the queue or server, and the queuing probability, correspond to resource utilization MOP's. For example, the average number of customers in the service facility at S_i^0 and C_i^0 correspond to the average number of targets currently under track and the average number of targets currently being engaged, respectively. The queuing probability at S_i^0 and C_i^0 tells us the likelihood that all the available reporting and threat prosecution assets of the i-th region are currently committed.

Our final class of system performance and effectiveness measures, leakage statistics, may be related to the probabilities of service completion at S_i^0 or C_i^0 respectively. For example, one minus the probability of service completion at S_i^0 may be identified with the probability that an as yet unreported target traverses the i-th region without being reported, and the product of one minus the probability of service completion at S_i^0 and the input arrival rate of unreported targets into S_i^0 corresponds to the leakage rate of unreported targets from the i-th region. Similarly, one minus the probability of service completion at C_i^0 corresponds to the probability that an assigned target leaks from the i-th region undestroyed, and a product of one minus the probability of service completion at S_i^0 and the input arrival rate of target allocations corresponds to the leakage rate of assigned targets from the i-th region.

The measures of system performance and effectiveness which we have discussed up to now have been identified directly as system statistics associated with the various queuing model components. We can also consider more global measures of system behavior which are derived by combining the more "local" queuing model component statistics. Primary examples of such system-wide measures of performance are various response time statistics, i.e., the time between a target's entry into the zone and the completion of any one out of the sequence of command and control processes necessary for threat prosectuion. One such response time statistic is the average time between target entry into the geographic domain of our notional AD system and the first engagement, along penetration route-ℓ, which we will refer to by the notation $\bar{\tau}_{e,en}^{(\ell)}$. While we will not present the details here, we note that one may compute $\bar{\tau}_{e,en}^{(\ell)}$ as a function of the following queuing model analysis outputs:

1. engagement probabilities and engagement delay statistics at C_i^0's;

2. report generation probabilities and report generation delay statistics at S_i^0's; and

3. processing delay statistics at S_i^1's, S^2, C^2, and C_i^1's.

Finally, we note that in addition to overall system response time statistics we may calculate various overall system leakage statistics. From report generation probabilities at the S_i^0's we can compute leakage rates of unreported targets from different regional or collections of regional domains, while from service completion probabilities at C_i^0's we can compute leakage rates of reported and allocated targets from various regional, or groups of regional domains.

5.3 Derivation of Queuing Model Parameters

In order to accomplish the previously outlined analysis of our aggregate queuing model of the notional AD system, we must not only specify the queuing system parameters associated with each of the model components, i.e., S_i^0, S_i^1, S^2, C^2, C_i^1, and C_i^0, but we must define the queuing parameters which determine the structure of the penetrator raid, i.e., the target arrival rates associated with each penetration route, the description of each route in terms of successive regional traversals, and the regional reneging parameters - the inverses of the average regional transit times. Therefore, we need algorithms that convert the physical description of the AD-C^2/C^3 system and the penetrator raid into the queuing parameters associated with the various model components and the raid, respectively. We will call these algorithms analyst interface models.

While we will not give an exhaustive discussion of existing analyst interface models here, we note that in general the calculation of the desired queuing parameters is controlled by the following types of issues:

1. knowledge of human information processing capabilities,

2. technical characteristics of penetrator and AD-C^3 system hardware,

3. the geographic distribution of C^2/C^3 system elements, and

4. doctrinal constraints.

To illustrate the interplay between the above issues in the formation of model parameters we first examine the calculation of the detection rate parameter, μ_{di}, associated with S_i^0, for a special case. Let us assume that the only surveillance assets assigned to the i-th region are flights of interceptors on a "search and destroy" mission. From the theory of random search, assuming that we have n_{FL} independent random search operations over a region of area A, with searchers that have a instantaneous detection range R_d, velocity V_I, and angular field of view θ, the probability that a given target is detected at some time $t^1 \leq t$, $P_d(t)$, is given by

$$P_d(t) = 1 - \exp\left\{\frac{-2\, n_{FL}\, R_d\, \sin(\theta/2) V_I}{A}\, t\right\} , \tag{5-7}$$

and hence the detection rate parameter μ_{di} is given as

$$\mu_{di} = \frac{2 n_{FL} R_d \sin(\theta/2)\, V_I}{A} . \tag{5-8}$$

From relation (5-8) we can argue that the calculation of μ_{di}, for the above special case, is controlled by the above issues (2) and (3).

As a second example of a simple analyst interface model we consider the choice n_{s_i}, the effective number of threat prosecution servers, or service subsystems, associated with C^0_i. In the case of a region with a defense consisting of flights of AI's, n_{s_i} may be determined by doctrinal considerations. If we let n_{c_i} denote the total number of controlled flights permitted by ground or airborne facilities (GCI or ACI operators), then intercepts may either be tightly monitored up to the point of weapons release, or loosely supervised such that only the location of an assigned threat is indicated. In the case of "loose" control the effective number of target servicing sybsystems is identical to the number of flights, n_{FL_i},

$$\text{i.e., } n_{s_i} = n_{FL_i} \tag{5-9}$$

while in the case of "tight" control"

$$n_{s_i} = \min\{n_{c_i}, n_{FL_i}\} \tag{5-10}$$

Similarly, in the case of a region with a SAM defense, for which targets must be illuminated by a fire control radar in order to be "processed," the effective number of simultaneous engagements is determined as

$$n_{s_i} = \min\{n_{L_i}, n_{FC_i}\} , \tag{5-11}$$

where n_{L_i} denotes the number of SAM launchers, and n_{FC_i} refers to the number of fire control radars.

SECTION 6. FUTURE DIRECTIONS

Interest in the quantification of AD system response time has impelled the development of a queuing theory based modeling and analysis approach to AD-C^2/C^3 problems. We conclude this paper by proceeding to identify avenues for the continuing development of the queuing methodology.

6.1 Analyst Interface Model Development

One major area for continuing work that supports our queuing modeling approach is in the development of more sophisticated analyst-interface models, that convert the physical characteristics of the AD system into parameters for the various queuing model components. On the threat prosecurtion side of our model, we might like to investigate more detailed modeling of interceptor versus penetrator or SAM versus penetrator interactions, to obtain better statistical information about the time required for a single engagement, or for re-engagement. On the surveillance side of our model, we would like to have models capable of translating information such as a target's trajectory relative to some surveillance site, technical characteristics of radars, and the statistics of the penetrator's radar cross-section into a probability distribution for the time between target entry into a region, and its reliable detection. In addition, we would like to have models which describe the changes in queuing parameters related, to either the surveillance or threat prosecution aspects of our model, when various forms of EW or C^3 countermeasures are employed.

6.2 Model Refinements and Extensions

We next focus on some options for further refining our existing model and its analysis. We first note that nowhere in our current model have we explicitly introduced the communications aspect of the air defense system. It is possible to do so implicitly by absorbing communication delays into the processing times at the various model components. However, we could extend the generality of our present model by embedding the various model components $S^0_\cdot$, $S^1_\cdot$, S^2, C^2, $C^1_\cdot$, and $C^0_\cdot$ in a communications network structure. The endogenous customers, i.e., track reports and target allocations, would flow through the communications network to reach their appropriate destinations. Nodes in the communications network would again be identified as queues, with processing rates related to the message sizes and transmission rates.

In our discussion concerning new analyst-interface models we mentioned more detailed modeling of the detection process. Another step in the direction of more faithful representation of the surveillance aspect of the AD system would be to disaggregate our S^0_i models into models of individual surveillance sites. Penetration routes would then be defined as traversals from one site to another

The representations of human decisionmaking implicit in our queuing models for $S^0_\cdot$, $S^1_\cdot$, S^2, C^2, $C^1_\cdot$, and $C^0_\cdot$, i.e., the first-come, first-serve service discipline, are particularly simple. It may be that work in the area of human-operator modeling will suggest some refinements of this aspect of our model.

In reviewing the air space management and threat prosecution doctrine inherent in our current model we note that it is totally centralized, i.e., the endogenous customers must flow through the full sequence $S^0_\cdot \to S^1_\cdot \to S^2 \to C^2 \to C^1_\cdot \to C^0_\cdot$

in order to enable the physical "processing" of targets. Especially in the event of various saturation modes, i.e., the S^2 or C^2 saturation, it is desirable to be able to reconfigure the system to some more decentralized mode of air space control, and to investigate the behavior of the system in its new mode of operation. In addition, it may be desirable to investigate the effect of different target handover, and allocation strategies than those implicit in our current implementation of the model.

Also, in our current version of the model we have only one type of target and one type of threat prosecution asset associated with each region. It may be of interest in the future to consider different classes of targets, with different associated priorities, and different threat prosecution and surveillance characteristics. Similarly, we may wish to consider C_i^0 models with more than one type of server, allowing us to distinguish between interceptors on CAP or strip alert.

We note that the steady-state queuing analysis we described in 5.1 is designed to describe the steady-state behavior of the AD system. Hence, in the future it may be desirable to investigate approximate techniques for transient queuing systems analysis in order to describe the transient behavior of the AD system.

Finally, we observe that nowhere in our current model do we account for the effects of attrition of the defensive forces. In the military operations research literature, Lanchester's equations and their generalizations have been used to model the attrition process for two-sided interactions between opposing forces. Hence, we may be able to couple the use of some extended set of Lanchester's equations, with our current model, to do a quasi-static sequence steady-state analyses for various system resource levels predicted by some Lanchester-like attrition calculation.

BIBLIOGRAPHY

Kleinrock, L., Queuing Systems, Volume I and Volume II, John Wiley & Sons (New York), 1975.

Syski, R., Introduction to Congestion Theory in Telephone Systems, Oliver and Boyd (London), 1962.

Ancker, C. J. and Gafarian, A. V., "Queuing with Impatient Customers Who Leave at Random," Journal of Industrial Engineering, Volume 13, No. 2, 1962, pp. 84-90.

Barrer, D. Y., "Queuing with Impatient Customers and Ordered Service," Operations Research, Volume 5, No. 5, 1957, pp. 650-656.

THREE-LEVEL HIERARCHICAL DECISION PROBLEMS*

Peter B. Luh

Dept. of Elec. Eng.
& Comp. Sci.
Univ. of Connecticut
Storrs, CT 06268

Tsu-Shuan Chang

Dept. of Elec. Eng.
SUNY at Stony Brook
Stony Brook, NY 11794

Taikang Ning

Dept. of Elec. Eng.
& Comp. Sci.
Univ. of Connecticut
Storrs, CT 06268

Abstract Three-person, three-level hierarchical decision problems are considered. Decision maker 0(DM0) is at the top level of the hierarchy. DM1 and DM2 are respectively at the middle level and the bottom of it. The problems are solved by the inducible region approach. We first derive inducibility conditions systematically, where we identify the dual purposes of DM0's strategy. Intuitively, if DM1's cost function depends on DM0's control variables, DM0 can assign adequate values to them (through his strategy) as a threat and thus imposes direct control on DM1's decision. On the other hand, if DM2's cost function depends on DM0's control variables, then DM0 can modify DM2's cost function and this in turn changes DM1's ability to induce DM2's behavior. Thus DM0 has indirect influence on DM1's decision. In case that DM0's control variable does not appear explicitly in DM2's cost function, dual control degenerates to direct control. The inducible region in this case can then be expressed in very simple terms. For unconstrained, strictly convex problems, we show that the inducible region equals to the whole space. Except for degenerate cases, DM0 can in fact use a piecewise linear strategy to induce the desired team solution. We also demonstrate how the above results can be extended to problems with incomplete information through a recursive conversion process.

1. INTRODUCTION

In a multi-level hierarchical decision problem, a higher level decision maker is typically assumed to have the authority to declare his strategy and enforce it on decision makers in lower levels. A lot of progress has been made recently in the study of two-person, two-level problems (under the title of Stackelberg games, see <BAS79>, <PAP79 > <TOL81a,b>, <HO81,82>, <CHA82a,b>, <LUH82a>). Roughly speaking, these approaches can be divided into two categories, i.e., the team solution approach and the inducible region approach. Basically, the team solution approach is to find first the team solution of the origional problem or an equivalent problem, and then construct a leader's strategy to achieve the team solution. On the other hand, the inducible region approach (introduced formally in <CHA82b>) is (1) to find the collection of all possible outcomes, known as the inducible region, (2) to obtain the optimal inducible outcome for the leader from the inducible region, and then (3) to construct a leader's strategy to induce it.

On three-level problems, results have been reproted in <BAS81> and <TOL81b>. In <BAS81>, dynamic problems are considered and a set of sufficient conditions are obtained for the achievement of team solution. While results in <TOL81b> are closely related to ours, we shall discuss them in detail later in section 5.

In this paper, we investigate three-level problems using the inducible region approach. In section 2, we introduce the solution concept and formulate our model. The inducible region is then presented mathematically in section 3. In section 4, we discuss the inducibility conditions. We identify in this process that the leader's strategy actually serves dual purposes. i.e., both direct control and indirect influence on his immediate follower. In section 5, we study the subclass of problems considered in <TOL81b>. We show that the inducible region can be expressed in very simple terms in this case, and compare our results with Tolwinski's results. In section 6, we show that the inducible region equals to the entire decision space for problems with strictly convex cost functions and unconstrained control variables. Furthermore, except for degenerate cases, there exists a piecewise linear hierarchical equilibrium strategy for the leader. In section 7, we demonstrate how the above

---

* The research work reported in this paper was supported by the National Science Foundation under Grant ECS 8105984, Grant ECS 8210673, and by the U.S. Office of Naval Research under the Joint Services Electronics Program by Contract N00014-75-C-0648.

results can be extended to problems with incomplete information through a recursive conversion process. Concluding remarks are given in section 8.

2. SOLUTION CONCEPT AND PROBLEM FORMULATION

Consider a three-person decision problem with three levels of hierarchy. The leader (DM0) is in a position to announce his strategy ahead of time and enforce it on the other two decision makers. The first follower (DM1), knowing the leader's strategy, then announces his strategy and enforce it on the second follower (DM2). The second follower, knowing both strategies of DM0 and DM1, thereafter selects his strategy to minimize his cost function. Let the strategy of DMi be denoted as γ_i, with $\gamma_i \varepsilon \Gamma_i$; the decision of DMi as $u_i \varepsilon U_i$; and the cost function of DMi as $J_i(\gamma_0, \gamma_1, \gamma_2)$. The above solution concept can be formulated rigorously as follows. For any given pair (γ_0, γ_1), the second follower is going to react by selecting a $\gamma_2 \varepsilon R_2(\gamma_0,\gamma_1)$, where

$$R_2(\gamma_0,\gamma_1) \overset{\Delta}{=} \{\gamma_2^*: J_2(\gamma_0,\gamma_1,\gamma_2^*) \leq J_2(\gamma_0,\gamma_1,\gamma_2) \forall \gamma_2 \varepsilon \Gamma_2\}, \tag{2.1}$$

is DM2's rational reaction set. For a given γ_0, the first follower is going to select a $\gamma_1 \varepsilon R_1(\gamma_0)$, where

$$R_1(\gamma_0) \overset{\Delta}{=} \{\gamma_1^*: \sup_{\gamma_2 \varepsilon R_2(\gamma_0,\gamma_1)} J_1(\gamma_0,\gamma_1^*,\gamma_2) \leq \sup_{\gamma_2 \varepsilon R_2(\gamma_0,\gamma_1)} J_1(\gamma_0,\gamma_1,\gamma_2) \quad \forall \gamma_1 \varepsilon \Gamma_1\}, \tag{2.2}$$

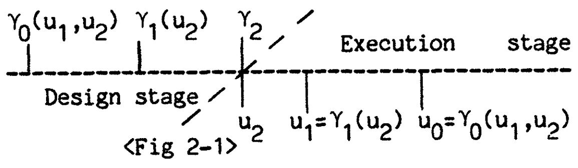
is DM1's rational reaction set. In other words, for a given γ_0, DM1 and DM2 form a two-person, two-level Stackelberg subgame, with DM1 as the leader. For the three-level problem, the task of DM0 is to find his hierarchical equilibrium strategy γ_0^* such that with DM1's and DM2's reactions, J_0 is minimized. That is,

$$\sup_{\gamma_1 \varepsilon R_1(\gamma_0^*)} \sup_{\gamma_2 \varepsilon R_2(\gamma_0^*,\gamma_1)} J_0(\gamma_0^*,\gamma_1,\gamma_2)$$

$$\leq \sup_{\gamma_1 \varepsilon R_1(\gamma_0)} \sup_{\gamma_2 \varepsilon R_2(\gamma_0,\gamma_1)} J_0(\gamma_0,\gamma_1,\gamma_2) \quad \forall \gamma_0 \varepsilon \Gamma_0. \tag{2.3}$$

Any $\gamma_1^* \varepsilon R_1(\gamma_0^*)$ is a corresponding equilibrium strategy for DM1 and any $\gamma_2^* \varepsilon R_2(\gamma_0^*, \gamma_1^*)$ is a corresponding equilibrium strategy for DM2. The supremes in (2.2) and (2.3) take into account the possible nonunique reactions of DM1 and DM2.

In this paper, we assume that each decision maker acts only once, with the sequence of actions as shown in figure 2-1.



<Fig 2-1>

In the design stage, the leader announces γ_0 first. DM1 announces γ_1 second. In the execution stage, DM2, knowing γ_0 and γ_1, selects a $\gamma_2 = u_2$. DM1, having complete information on u_2, then decides u_1 according to his announced γ_1, i.e., $u_1 = \gamma_1(u_2)$. Finally the leader, having complete information on both u_1 and u_2, calculates u_0 according to $u_0 = \gamma_0(u_1,u_2)$. We assume that γ_i can be any function from DMi's information set to U_i. The problem thus formulated is known to have a reversed information structure <HO81> in the sense that though the leader announces γ_0 first, he actually acts last.

3. THE INDUCIBLE REGION CONCEPT

As introduced in <CHA82a,b>, the inducible region, IR, is the set of all possible outcomes from the leader's viewpoint. In the three-level case, it is defined formally as follows. For a given γ_0 any $(\gamma_0, \gamma_1, \gamma_2)$ with $\gamma_1 \varepsilon R_1(\gamma_0)$ and $\gamma_2 \varepsilon R_2(\gamma_0, \gamma_1)$ is called an equilibrium triple according to our definition. And a point (u_0,u_1,u_2) in the decision space such that $u_2=\gamma_2$, $u_1= \gamma_1(u_2)$, $u_0=\gamma_0(u_1,u_2)$ is the real outcome of the problem for this $(\gamma_0,\gamma_1, \gamma_2)$. However, in the design stage, the leader's viewpoint regarding the outcome is somewhat different. According to (2.3), the leader assumes that the followers choose a particular pair (γ_1', γ_2') according to the following equation:

$$(\gamma_1',\gamma_2') = \arg \sup_{\gamma_1 \varepsilon R_1(\gamma_0)} \sup_{\gamma_2 \varepsilon R_2(\gamma_0,\gamma_1)} J_0(\gamma_0,\gamma_1,\gamma_2). \tag{3.1}$$

The point (u_0',u_1',u_2'), such that $u_2'=\gamma_2'$, $u_1'=\gamma_1'(u_2')$, and $u_0'= \gamma_0(u_1',u_2')$ is thus called the outcome from the leader's viewpoint for this γ_0. As in <CHA82b>, we can formulate equivalent classes and, with little abuse of notation, say that from the leader's viewpoint, the unique outcome (u_0',u_1',u_2') is induced by this γ_0. The Inducible Region, IR, is then defined as

$$IR \overset{\Delta}{=} \{(u_0,u_1,u_2): \text{There exists a } \gamma_0 \varepsilon \Gamma_0 \text{ such that } (u_0,u_1,u_2) \text{ is the unique outcome from the leader's viewpoint }\}. \tag{3.2}$$

From the above definition, any point not belonging to IR can not be the outcome (from the leader's viewpoint). Thus if the leader wants to find the best outcome, he can only consider IR. We then have the counterpart of the functional optimization problem of (2.3)

$$J_0^* = \inf_{(u_0,u_1,u_2) \varepsilon IR} J_0(u_0,u_1,u_2) . \tag{3.3}$$

Therefore, the problem boils down how to delineate the inducible region, IR.

4. INDUCIBILITY CONDITIONS

For a given γ_0, DM1 and DM2 form a two-level problem, where the results in <CHA82b> are directly applicable. With this in mind, the leader can be conceived of as facing an equivalent two-level problem, where for a given γ_0, the reaction of the followers (in terms of (γ_1,γ_2) or (u_1,u_2)) is determined by the outcome of the nested problem of DM1 and DM2.

Let us first consider the nested two-level problem. For a given γ_0 the inducible region for DM1, denoted as $IR_1(\gamma_0)$, is given by

$$IR_1(\gamma_0) = \{(u_1,u_2): J_2(\gamma_0(u_1,u_2),\ u_1,u_2) \lesssim m_2(\gamma_0)\}, \tag{4.1}$$

where $m_2(\gamma_0)$ is defined by

$$m_2(\gamma_0) = \inf_{u_2} \sup_{u_1} J_2(\gamma_0(u_1,u_2),u_1,u_2). \tag{4.2}$$

We shall interpret (4.1) intuitively and explain what the special notation $\lesssim$ means. Note that no matter what DM1 is going to do, DM2 can always guarantee himself that J_2 will not be greater than $m_2(\gamma_0)$ by minimizing J_2. Thus any point with $J_2 > m_2(\gamma_0)$ is not inducible because DM2 will never choose such a point. Any point with $J_2 < m_2(\gamma_0)$ is inducible by DM1, since DM1 can penalize DM2 at least $m_2(\gamma_0)$ by maximizing J_2 if DM2 deviates. For those points on the $J_2 = m_2(\gamma_0)$ boundary, the situatuon is more complicated. To concentrate on main issues, we use $\lesssim$ in (4.1) to represent the complications. From now on, the same notation will be used to represent such complications on boundaries.

Let us now turn to the outer layer problem of DM0 and examine whether a point $u'=(u_0',u_1',u_2')$ is inducible or not. A candidate $_0'$ that induces (u_0',u_1',u_2') has to take the following form:

$$\gamma_0'(u_1,u_2) = \begin{cases} u_0', & \text{if } (u_1,u_2) = (u_1',u_2'), \\ P_0(u_1,u_2) & \text{otherwise,} \end{cases} \tag{4.3}$$

where P_0, yet to be specified, is a function from $U_1 \times U_2$ to U_0. In order to induce u' by γ_0', the following two conditions have to be satisfied.

(1) For the given γ_0, DM1 must be able to induce (u_1',u_2'), that is, $(u_1',u_2') \in IR_1(\gamma_0)$; otherwise, the outcome of the nested two-level problem can not be (u_1',u_2'), and u' is not inducible.

(2) DM1 must prefer (u_1,u_2) to other points points in $IR_1(\gamma_0')$; otherwise, DM1 will induce those points he prefers rather than (u_1',u_2'), and thus u' is not inducible.

It is easily seen that the converse is also true. That is, if (1) and (2) hold, then u' is inducible by γ_0'. Thus we have

$u' \in IR$ if and only if there exists a $P_0(.,.)$ such that

(C1): $(u_1',u_2') \in IR_1(\gamma_0')$, and

(C2): $J_1(u') < J_1(P_0(u_1,u_2),u_1,u_2)$
$\forall\ (u_1,u_2) \in IR_1(\gamma_0')$ and $(u_1,u_2) \neq (u_1',u_2')$.

(4.4)

In words, (C1) is the capability condition which says that the leader should let DM1 be capable of inducing (u_1',u_2'). The desirability condition (C2) then says that the leader should make DM1 prefer (u_1',u_2') to other points within DM1's capability.

Let's now derive some explicit results from (C1) ans (C2). First, (C1) and (4.1) implies that

$$J_2(u') \lesssim \inf_{u_2} \sup_{u_1} J_2(\gamma_0'(u_1,u_2),u_1,u_2) \tag{4.5a}$$

$$\leq \inf_{u_2} \sup_{u_1} \sup_{u_0} J_2(u_0,u_1,u_2) \triangleq M_2. \tag{4.5b}$$

The constant M_2 can be thought of as DM2's guaranteed cost in the sense that even if both DM0 and DM1 cooperatively maximize J_2, DM2 can still get M_2. Therefore it is impossible for the leader to induce any point with $J_2 > M_2$. This in turn implies that

$$IR \subset \{u : J_2(u) \lesssim M_2 \}. \tag{4.6}$$

The condition (C2) implies that

$$J_1(u') \lesssim \inf_{\substack{(u_1,u_2) \in IR_1(\gamma_0') \\ (u_1,u_2) \neq (u_1',u_2')}} J_1(P_0(u_1,u_2),u_1,u_2). \tag{4.7}$$

Thus, in order to induce u', the leader has to select $P_0(.,.)$ such that (4.7) holds. For a given u', the most the leader can do for this is to select $P_0(.,.)$ that maximizes the right hand side of (4.7), i.e.,

$$\max_{P_0(.,.)} \{ \inf_{\substack{(u_1,u_2) \in IR_1(\gamma_0') \\ (u_1,u_2) \neq (u_1',u_2')}} J_1(P_0(u_1,u_2),u_1,u_2)\}. \tag{4.8}$$

Note that $IR_1(\gamma_0')$ is given by (4.1), and it is $P_0(.,.)$ dependent. Thus, for the term inside the braces of (4.8), P_0 actually serves two different purposes. First, it affects J_1 directlly, since $P_0(u_1,u_2)$ provides an argument to J_1. Secondly, it affects the region of taking the infimum, since it can shape $IR_1(\gamma_0)$. Intuitively, if the leader at the top of the hierarchy prefers a certain outcome (u_0', u_1', u_2'), he can achieve it by two approaches. He can either use *direct control* (if DM1 tries to achieve some other (u_1, u_2), J_1 will be higher), or have *indirect influence on DM1 through DM2* (if DM1 tries to achieve some other (u_1, u_2), the leader will work through DM2 so that DM2 will not comply with DM1). *The dual roles of the leader's strategy in general are not separable, and make the functional optimization problem (4.8) difficult.* In the following, we shall consider two classes of problems and delineate their inducible regions explicitly.

5. INDUCIBLE REGION FOR A CLASS OF PROBLEMS

Consider the problem in which the leader can not affect DM2 directly, i.e., $J_2 = J_2(u_1, u_2)$. From (4.1), we have

$$IR_1(\gamma_0) = IR_1 = \{(u_1, u_2) : J_2(u_1, u_2) \lesssim \inf_{u_2} \sup_{u_1} J_2(u_1, u_2)\}, \tag{5.1}$$

Thus IR_1 in this case is independent of γ_0. For any given γ_0, DM1 will induce a (u_1, u_2) within IR_1 that minimizes J_1. The problem faced by the leader is then equivalent to the two-level problem of DM0 and DM1 where DM1's decision (u_1, u_2) is restricted to IR_1. As a result, (4.8) degenerates to

$$\max_{P_0} \{ \inf_{\substack{IR_1 \\ (u_1, u_2) = (u_1', u_2')}} J_1(P_0(u_1, u_2), u_1, u_2)\}$$

$$= \inf_{\substack{(u1, u_2) \varepsilon IR_1 \\ (u_1, u_2) \neq (u_1', u_2')}} \max_{u_0} J_1(u_0, u_1, u_2) . \tag{5.2}$$

and the dual purposes of P_0 reduce to *direct control* only. Equation (5.2) is a parameter optimization problem, and in principal can be solved. From (4.7) and (5.2), we then conclude that

$$J_1(u') \lesssim \inf_{(u_1, u_2) \varepsilon IR_1} \sup_{u_0} J_1(u_0, u_1, u_2) \overset{\Delta}{=} M_1, \tag{5.3}$$

where M_1 can be thought of as DM1's guaranteed cost. The maximum penalty strategy is given by

$$\gamma_{0M1}(u_1, u_2) = \begin{cases} u_0^*(u_1, u_2) & \text{if } (u_1, u_2) \varepsilon IR_1, \\ \text{arbitrary} & \text{otherwise.} \end{cases} \tag{5.4}$$

where $u_0^*(u_1, u_2)$ is the solution[2] of taking supreme in (5.3). We then have the following results.

Theorem 1 If J_2 is not a function of u_0, then

$$IR = \{(u_0, u_1, u_2) : J_1(u_0, u_1, u_2) \lesssim M_1 \text{ and } J_2(u_1, u_2) \lesssim M_2\} . \tag{5.5}$$

Proof:

(1) It is apparent that any point outside the region defined in (5.5) in not inducible.

(2) To show that any point $u_0' = (u_0', u_1', u_2')$ in the region is inducible, let us construct a γ_0 as follows.

$$\gamma_0(u_1, u_2) = \begin{cases} u_0' & \text{if } (u_1, u_2) = (u_1', u_2'), \\ \gamma_{0M1} & \text{otherwise.} \end{cases} \tag{5.6}$$

where γ_{0M1} is defined by (5.4). With this γ_0, DM1 will induce (u_1', u_2'); for otherwise, $J_1 > M_1$. Also, DM1 is able to do it since $J_2(\tilde{u}_1, u_2) \lesssim M_2$ and $(u_1', u_2') \varepsilon IR_1$. Q.E.D.

Thus there are three steps to solve this class of problems.

(S1): Delineate IR in (5.5).

(S2): Find the optimal inducible outcome from the problem below.

$$u^* = (u_0^*, u_1^*, u_2^*) = \arg \min_{(u_0, u_1, u_2) \varepsilon IR} J_0(u_0, u_1, u_2) \tag{5.7}$$

(S3): Construct a γ_0^* (e.g., the one given in (5.6) to induce u^*.

Since Tolwinski treated the same problem in <TOL81b>, we shall compare the above results with his results. Denote D_T as the region obtained in <TOL81b>. It is given by (eqs.(37) to (41) in <TOL81b>).

$$D_T = \{(u_0, u_1, u_2) : J_1(u_0, u_1, u_2) \lesssim m_{1T}(u_2) \text{ and } J_2(u_1, u_2) \lesssim m_{2T}\} , \tag{5.8}$$

where

$$m_{1T}(u_2) = \inf_{u_1} \sup_{u_0} J_1(u_0, u_1, u_2), \tag{5.9}$$

$$m_{2T} = \inf_{u_2} \sup_{u_1} J_2(u_1, u_2). \tag{5.10}$$

It was claimed that

$$J_0^* = \inf_{(u_0, u_1, u_2) \varepsilon D_T} J_0(u_0, u_1, u_2). \tag{5.11}$$

It is not difficult to see that in general

---

[1] This interpretation was suggested by Mr. Ying-Ping Zheng, visting scholar at Harvard University.

[2] Since we do not consider the complications on the boundaries, we shall assume the solution of (5.3) exists.

$D_T \not\leq IR$. Furthermore, it is neither $D_T \not\subset IR$ nor $IR \not\subset D_T$. We shall see from the following example that the claim (5.11) in <TOL81b> is incorrect. Another result in the same paper (Theorem 3 of <TOL81b> can be interpreted as a solution to a different model, where DM2 is the second leader and DM1 is at the lowest level of the hierarchy. That is, after γ_0 is announced, DM2 announces $\gamma_2 = u_2$. In the execution stage, the decision sequence is still the same, i.e., u_2, u_1 and u_0. By using the solution concept of (2.2) and (2.3) with the exchange of roles of DM1 and DM2, and following the systematic derivation outlined in section 4, we can obtain Theorem 3 of <TOL81b>.

Example 1

$$J_0 = (u_0-0.88)^2 + (u_1-0.4)^2 + (u_2-0.9)^2$$

$$J_1 = u_0^2 + (u_1-1)^2 + (u_2-1)^2$$

$$J_2 = u_1^2 + u_2^2 \qquad 0 \leq u_i \leq 1 \quad \text{for } i=0,1,2$$

From (4.2),

$$M_2 = \min_{u_2} \max_{u_1} J_2 = 1,$$

then

$$IR_1 = \{(u_1,u_2) : J_2 \lesssim M_2\}$$

$$= \{(u_1,u_2) : (u_1^2 + u_2^2) \lesssim 1\} .$$

From (5.3),

$$M_1 = \min_{(u_1,u_2)\varepsilon IR_1} \max_{u_0} J_1 = 1.72.$$

Thus

$$IR = \{(u_0,u_1,u_2): J_1 \lesssim M_1 \text{ and } J_2 \lesssim M_2\},$$

$$= \{(u_0,u_1,u_2): u_0^2 + (u_1-1)^2 + (u_2-1)^2 \lesssim 1.72 \text{ and } (u_1^2 + u_2^2) \lesssim 1\}.$$

We can visualize that IR_1 is a portion of a cylinder with radius 1 centered at $(u_1,u_2) = (0,0)$. IR is the intersection of IR_1 with a sphere (with radius 1.083 centered at $(u_0,u_1,u_2) = (0,1,1)$). It is clear that the optimum inducible point is $u^* = (0.88, 0.4, 0.9)$ which is the team solution. One γ_0^* is given by

$$\gamma_0^*(u_1,u_2) = \begin{cases} 0.88 & \text{if } (u_1,u_2) = (0.4,0.9), \\ 1 & \text{otherwise.} \end{cases}$$

For this γ_0^*, an optimal strategy for DM1 is

$$\gamma_1^*(u_2) = \begin{cases} 0.4 & \text{if } u_2 = 0.9, \\ 1 & \text{otherwise.} \end{cases}$$

If we follow the results in <TOL81b>, we have

$$m_{1T}(u_2) = 1 + (u_2-1)^2, \qquad m_{2T} = 1, \text{ and}$$

$$D_T = \{(u_0,u_1,u_2) : (u_0^2+(u_1-1)^2) \lesssim 1, \text{ and } (u_1^2 + u_2^2) \lesssim 1\}.$$

It is easy to see that $D_T = IR$; furthermore, $u^* \notin D_T$.

6. UNCONSTRAINED, STRICTLY CONVEX PROBLEMS

We shall show that any given point, $u' = (u_0',u_1',u_2')$, is always inducible for the following problem:

Cost functions: $J_i(u_0,u_1,u_2)$, $i=0,1,2$, where J_1 and J_2 are strictly convex in their arguments.
DM2's decision : $u_2 \,\varepsilon\, U_2 = R$,
DM1's decision : $u_1 = \gamma_1(u_2) \,\varepsilon\, U_1 = R$,
DM0's decision : $u_0 = \gamma_0(u_1,u_2) \,\varepsilon\, U_0 = R$.
(6.1)

Since $J_2(u_0,u_1,u_2)$ is strictly convex in its arguments, the region

$$D_2 = \{ u : J_2(u) \leq J_2(u')\} \tag{6.2}$$

is closed, bounded and strictly convex. Similar statement holds for the region

$$D_1 = \{u : J_1(u) \leq J_1(u') \}. \tag{6.3}$$

Since u_0 is not restricted, there always exists a function, say $\hat{P}_0(u_1,u_2)$, that lies outside D_1 and D_2. Let

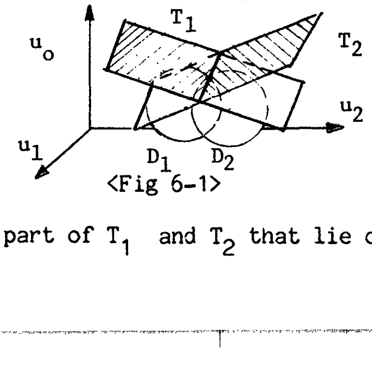
$$\gamma_0^*(u_1,u_2) = \begin{cases} u_0' & \text{if } (u_1,u_2) = (u_1',u_2'), \\ \hat{P}_0(u_1,u_2) & \text{otherwise.} \end{cases} \tag{6.4}$$

Apparently γ_0^* induces u'. Consequently, the leader can always achieve his absolute minimum (team cost), which is defined as

$$\min_{(u_0,u_1,u_2)} J_0(u_0,u_1,u_2). \tag{6.5}$$

Theorem 2 For the uncontrained, strictly convex problem formulated in (6.1), $IR = R^3$, thus the team solution is always achievable.

In fact, except for degenerate cases, γ_0^* can be chosen to be piecewise linear. Consider an arbitrary point $u'=(u_0',u_1',u_2')$ as shown in figure 6-1. In the figure, we also show D_1 and D_2 as defined by (6.3) and (6.2). Let the tangent planes to D_1 and D_2 that pass through u' be denoted as T_1 and T_2 respectively. If the degenerate case does not occur, i.e., none of T_1 and T_2 is perpendicular to the $U_1 \times U_2$ plane,



<Fig 6-1>

then the part of T_1 and T_2 that lie outside

D_1 and D_2 (as shown by the shaded area) can be chosen as a leader's equilibrium strategy. Thus we have

Theorem 3 For unconstrained, strictly convex problems formulated in (6.1), except for degenerate cases, the team solution can be achieved by a piecewise linear strategy of the leader.

7. PROBLEMS WITH INCOMPLETE INFORMATION

Consider the following problem with incomplete information.

Cost functions: $J_0(u_0,u_1,u_2)$, $J_1(u_0,u_1,u_2)$, $J_2(u_1,u_2)$,
DM2's control: $u_2 \in U_2$,
DM1's control: $u_1 = \gamma_1(y_1)$ with $y_1 = h_1(u_2)$ and $u_1 \in U_1$,
DM0's control: $u_0 = \gamma_0(y_0)$ with $y_0 = h_0(u_1,y_1)$ and $u_0 \in U_0$.

The problem can be solved through a recursive conversion process, which is a nontrival extension of the result of <BAS82> for two-level problems. Due to space limitation, we shall just summarize the steps in solving this problem. For details, please refer <LUH82b>.

1. Convert the nested two-level problem and delineate IR_1,

2. Convert the outer-layer problem and delineate IR.

3. Find the optimum inducible outcome for DM0 and construct the strategy.

8. CONCLUDING REMARKS

The concept of inducible region was formulated formally in <CHA82a,b> and used to solve single-stage, two-level problems in a systematic way. The advantage of this approach is the reduction of a functional optimization problem to a parameter optimization (ref.(2.3) and (3.3)). However, because of the mathematical similarity between the results of <CHA82b> and those of <TOL81> (see <CHA82b> for discussions), the usefulness of this approach was not very well recognized.

In this paper, we use the inducible region concept to solve three-level problems. Defects in previous results are observed. Many properties, as well as the difficulty of the problem, reveal themselves throughout the derivation. In particular, the dual purposes of the leader's strategy, direct control and indirect infulence, are identified. In case that the leader's control does not enter into DM2's cost function, dual control degenerates to direct control, and the inducible region can be expressed in very simple terms. For unconstrained, strictly convex problems, the inducible region equals to the whole space. Except for degenerate cases, the leader can in fact use a piecewise linear strategy to induce the desired solution. We finally extend the above results to problems with incomplete information.

REFERENCES

<BAS79> Tamer Basar, Hassan Selbuz, "closed Loop Stackelberg Strategies with Application in the Optimal Control of Multilevel Systems", IEEE Transactions on Automatic Control, Vol. AC-24, No.2, April 1979, pp. 166-179.

<BAS81> Tamer Basar, "Equilibrium Strategies in Dynamic Games with Multi-Level of Hierarchy", Automatica, Vol. 17, No.5, 1981, pp. 749-754.

<BAS82> Tamer Basar, "A General Theory for Stackelberg Games with Partial State Information", Large Scale Systems, Vol. 3, 1982, pp. 47-56.

<CHA82a> Tsu-Shuan Chang, Peter B. Luh, "The Concept of Inducible Region in Stackelberg Games", the Proceedings of the 1982 American Control Conference, Arlington, Virginia, June 1982, pp.139-140.

<CHA82b> Tsu-Shuan Chang, Peter B. Luh, "A Complete Solution for Two-Person, Single-Stage Deterministic Stackelberg Games", to appear in the Proceedings of the 21st IEEE Conference on Decision and Control, Orlando, Florida, Dec. 1982.

<HO81> Yi-Chi Ho, Peter B. Luh, Ramal Muralidharan, "Information Structure, Stackelberg Games and Incentive Controllability", IEEE Transactions on Automatic Control, Vol. AC-26, No. 2, April 1981, pp. 454-460.

<HO82> Yi-Chi Ho, Peter B. Luh and G. J. Olsder, "A Control Theoretic View on Incentives", Automatica, Vol. 18, No.2, March 1982, pp. 167-179.

<LUH82a> Peter B. Luh, Tsu-Shuan Chang, Shi-Chung Chang, "On Dynamic, Deterministic Stackelberg Games", the Proceedings of the 1982 American Control Conference, Arlington, Virginia, June 1982, pp. 409-411.

<LUH82b> Peter B. Luh, Tsu-Shuan Chang, Taikang Ning, "Three-Level Hierarchical Decision Problems", submitted.

<PAP79> George P. Papavassilopolous, Jose B. Cruz, Jr., "Nonclassical Control Problems and Stackelberg Games", IEEE Transactions on Automatic Control, Vol. AC-24, No. 2, April 1979, pp. 155-166.

<TOL81a> B. Tolwinski, "Closde-Loop Stackelberg Solution to Multi-Stage Linear-Quadratic Game", Journal of Optimization Theory and Application, Vol. 34, No. 4, August 1981, pp. 485-501.

<TOL81b> B. Tolwinski, "Equilibrium Solution for a Class of Hierarchical Games", in Applications of Systems Theory to Economics, Mangement and Technology, J. Gutenbaum and M. Niezgodka, Eds., Warsaw, PWN, 1981, .pp 581-600.

EFFECTIVENESS ANALYSIS OF C^3 SYSTEMS

Vincent Bouthonnier

Alexander H. Levis

Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, Mass., USA

Abstract. A methodology for analyzing and assessing the effectiveness of command, control and communications (C^3) systems is developed. The analysis is carried out by characterizing separately both the system and the mission in terms of attributes. These attributes are determined as functions of primitives that describe the system, the mission, and the context within which both operate. Then the system capabilities and the mission requirements are compared in a common attribute space. This comparison leads to the evaluation of partial measures of effectiveness which are then combined to yield a global measure. The methodology is illustrated through the assessment of the effectiveness of a communications network operating in a hostile environment.

INTRODUCTION

System effectiveness is an elusive concept that encompasses technical, economic, and behavioral considerations. When the system to be evaluated is one which provides a service, such as a command, control and communications (C^3) system, then the needs of the organization that uses it must be taken into account. Furthermore, the worth of the service it provides may change in value as missions change, as technologies change and as the opponent's capabilities change. Thus, any methodology that is proposed for C^3 system effectiveness analysis must be sufficiently broad and flexible so that it can accommodate change and can evolve over time.

Such a methodology is proposed in this paper. The analytical aspects of the methodology address mainly the relationships between component characteristics, system structure, and operating procedures to system availability and performance. Availability is defined as a probabilistic quantity dependent on the random failure characteristics of system components (whether or not due to enemy action). System performance denotes the ability to achieve appropriate operational goals for a given availability state [Fink, 1980]. It is assumed that the cost associated with any system realization and operation can be computed: the total cost may reflect the costs for developing and implementing the system and the costs for operating and maintaining it. Finally, the assessment of worth is left for the final, and subjective stage of the methodology, since worth is a relative measure that involves value judgements [Dersin and Levis, 1981].

In analyzing the effectiveness of a C^3 system, it is essential that the diversity of users and types of services demanded be taken into account. Also, the tolerances associated with each system characteristic or attribute must be established so that the adequacy of the service provided by a given system realization can be evaluated.

The basic premise of the methodology is that a C^3 system provides a variety of services (or supports a variety of functions). The complementary premise is that each user imposes on a C^3 system a load which is generated from a need for service that the system may or may not be able to satisfy. Thus, on one side, there is the C^3 system with a range of capabilities for providing service, while on the other is the military organization with its diverse needs for service. Therefore, the first step of the methodology is based on the ability to model the system's capabilities and the organization's requirements in terms of commensurate attributes. This and the other steps in the methodology are described in the next section. In the third section, an illustrative example is presented.

SYSTEM EFFECTIVENESS ANALYSIS

The methodology outlined in this section is based on six concepts: *system, mission, context, primitives, attributes,* and *measures of effectiveness.* The first three describe the problem, while the last three define the key quantities in the analytical formulation of the problem.

The *system* consists of components, their interconnection and a set of operating procedures. A naval communications system, a computer network or a testbed are typical systems. The system can be centralized (e.g., a testbed facility) or decentralized (e.g., a computer network).

The *mission* consists of a set of objectives and tasks that the military organization is assigned to accomplish. The description of the mission must be as explicit and specific as possible so that it can be modeled analytically. For example, a mission specification such as "to defend the West Coast of the US" is too broad, while a more useful specification would be "to detect enemy submarines off the cost of California".

The *context* denotes the set of conditions and assumptions, i.e., the environment, within which the mission takes place and the system operates. For example, the context may include specification of the geographical area, the time of the year, and the prevailing set of international agreements.

Primitives are the variables and parameters that describe the system and the mission. For example, in the case of a communications network, primitives may include the number of links and nodes, the capacity of each link, and the probability of failure of each link. Primitives of a mission may be the designation of origin-destination pairs, the data flow rate between these points, and the duration of each transmission. Let the system primitives be denoted by the set $\{x_i\}$ and the mission primitives by the set $\{y_j\}$.

Attributes are quantities that describe system properties or mission requirements. System attributes for a communications system may include reliability, average delay, and survivability. Mission attributes are expressed as requirements for the same quantities as the system attributes, e.g., minimum reliability, maximum average delay, or minimum survivability. The system attributes are denoted by the set $\{A_s\}$ and the mission attributes by $\{A_m\}$.

Measures of Effectiveness are quantities that result from the comparison of the system and mission attributes. They reflect the extent to which the system is well matched to the mission.

These six concepts are the key components of the methodology for analyzing and assessing the effectiveness of C^3 systems.

The first step of the methodology consists of the selection of the set of system primitives. By definition, the elements of the set are mutually independent. In this sense, the primitives are the independent variables in the analytical formulation of the methodology.

The second step consists of defining attributes for the system that characterize the properties that are of interest in the analysis. The attributes are expressed as functions of the primitives. The values of the attributes could be obtained from the evaluation of a function, from a model, a computer simulation, or from empirical data. Each attribute depends, in general, on a subset of the primitives, i.e.,

$$A_s = f_s\ (x_i,\ldots,x_k) \qquad (1)$$

Attributes may or may not be independent from each other. They are dependent, if they have primitives in common. A system realization results in the set of primitives taking specific values $\{\bar{x}_i\}$. Substitution of these values in the relationships (1) yields values for the attribute set $\{A_s\}$. Thus, any specific realization can be depicted by a point in the attribute space.

The third and fourth steps consist of carrying out a similar analysis for the mission: Selection of the primitives that describe the variables and parameters of the mission and definition of the mission requirements. Then models are selected that map the primitives y_j into the attributes:

$$A_m = f_m\ (y_j,\ldots,y_\ell) \qquad (2)$$

Some of the mission attributes may be interrelated through dependence on common primitives. It is also possible to introduce directly some constraints between the attributes, e.g., a trade-off relationship between delay and accuracy. However, it is preferable that such trade-off relationships be derived through the functions or models that define attributes or requirements in terms of the mission primitives. Specification of values for the mission primitives results in a point or region in the mission attribute space.

The two spaces, the system attribute space $\mathcal{A}_s$ and the mission attribute space $\mathcal{A}_m$, although of the same dimension, may be defined in terms of different attributes, or attributes scaled differently. Therefore, the fifth step consists of transforming the system and mission attributes into a set of common, commensurate attributes that define a common attribute space $\mathcal{A}$. For example, one of the system attributes may be vulnerability, while the corresponding mission attribute may be survivability. Since they both reflect the same concept -- the effect of hostile actions -- one of them may be chosen as the common attribute, say, survivability, while the other one will then be mapped into the first one. Once the common set of attributes has been defined, the two sets $\{A_s\}$ and $\{A_m\}$ are transformed into commensurate sets that can be depicted in the common attribute space $\mathcal{A}$.

A possible additional operation in this step is the normalization of the various commensurate attributes so that their values are in the range [0,1]. If all the attributes are normalized in this manner, then the common attribute space is the unit hypercube. This is very useful in depicting graphically the loci of the sets $\{A_s\}$ and $\{A_m\}$ and in analyzing their interrelationships.

The sixth step is the key one in analyzing the effectiveness of a C^3 system in view of the mission that is to be carried out. It consists of procedures for comparing the system and mission attributes through the geometric properties of two loci in the attribute space. Consider first all the allowable values that the primitives of a specific realization of the system may take. If the primitives are allowed to vary over their admissible ranges, then the variations define a locus L_s in the attribute space. Similarly, a mission locus L_m can be constructed. Both loci are defined in the unit hypercube. The geometric relationship between the two loci can take one of three forms:

(a) The two loci do not have any points in common, i.e., the intersection of L_s with L_m is null:

$$L_s \cap L_m = \phi \tag{3}$$

In this case, the system attributes do not satisfy the mission's requirements and one would define the effectiveness to be zero, regardless of which specific measure is used.

(b) The two loci have points in common, but neither locus is included in the other:

$$L_s \cap L_m \neq \phi \tag{4}$$

and

$$L_s \cup L_m > L_s \tag{5}$$

In this case, a subset of the values that the system attributes may take satisfies the mission requirements. Many different measures can be used to describe the extent to which the system meets the requirements. Each of these measures may be considered as a measure of effectiveness which, if normalized, takes values in the open interval (0,1). For example, let V be a measure in the normalized attribute space. Then an effectiveness measure can be defined by

$$E = V(L_s \cap L_m)/V(L_s) \tag{6}$$

which emphasizes how well matched the system is to the mission.

(c) The mission locus is included in the system locus:

$$L_s \cap L_m = L_m \tag{7}$$

In this case, it follows from (7) that L_s is larger then L_m and, consequently, the ratio defined by (6) will be less than unity. This result can be interpreted in two ways. First, only certain system attributes values meet the requirements of the mission. This is consistent with the interpretation given in case (b). The second interpretation is that the use of this system for the given mission represents an inefficient use of resources since the system capabilities exceed the mission requirements. Inefficiency, in turn, implies lower effectiveness.

If the system locus is included in the mission locus, then the system's effectiveness is identically equal to unity.

The measure of effectiveness given by (6) is one of many partial measures that can be defined in the common attribute space. Let these partial measures be denoted by $\{E_r\}$. To combine these partial measures into a single global measure, utility theory may be used [Debreu, 1958; Phlips, 1974]. The k partial measures $E_1,\ldots,E_k$ are now considered to be the arguments of a utility function u. However, for the valid application of utility theory, the arguments of u must belong to the positive orthant of $\mathcal{R}^k$, i.e., they should take values in $[0,+\infty)$. For this to happen, each E_r that takes values in [0,1) is mapped to an $\tilde{E}_r$ that takes values in $[0,+\infty)$. Many functions exist for transforming the bounded variables E_r to the unbounded ones; typical examples are

$$-\log(1-E)\ ;\ E/1-E\ ;\ \tanh^{-1}E$$

$$\tan(\pi E/2)\ ;\ E^{\alpha}/1-E^{\beta}$$

Each of these mappings tends to emphasize different segments of the range [0,1) and therefore weight in a different way the partial effectiveness measures E_r. Therefore, the subjective judgements of the system designers and the users can be incorporated directly into the methodology in three ways: (a) by choosing different partial measures, (b) by choosing the mapping function, and (c) by selecting a utility function. The global effectiveness measure is obtained, finally, from

$$\hat{E} = u(\tilde{E}_1,\tilde{E}_2,\ldots,\tilde{E}_k) \tag{8}$$

The seven steps of the methodology and their interrelationships are shown schematically in Figure 1. The diagram emphasizes that the system and the mission must be modeled and analyzed independently, but in a common context. The system capabilities should be determined independently of the mission and the mission requirements should be derived without considering the system to be assessed. Otherwise, the assessment is biased.

The methodology will be illustrated in the next section through application to a communications network operating in a hostile environment.

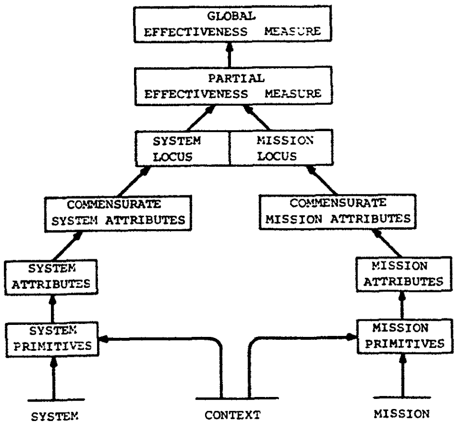


Figure 1. The Methodology for C^3 System Effectiveness Analysis

ASSESSMENT OF A C^3 NETWORK

Consider the communications network presented in Figure 2. It consists of seven nodes and thirteen links. The nodes represent information collection and transmission centers or decision centers or both. The network is assumed to be part of a C^3 system operating in a hostile environment. Specifically, it is assumed that the links are subject to jamming that disrupts communication between nodes. There are twenty-one possible origin-destination pairs in this network; only the pair (1,7) will be used because the subsystem it defines is equal to the whole network. Multiple pairs can be analyzed if each pair is considered as a subsystem.

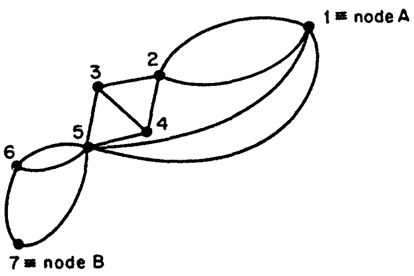


Figure 2. A Simple Communications Network.

The context for this network determines the environment in which the system will operate: geographical location, climatic conditions, enemy capabilities and resources. The context determines many of the primitives of both the system and the mission.

The mission is defined in terms of the objectives and tasks assigned to each node by the tactical plan. Let the aspect of the mission that is relevant to the pair (1,7), denoted by (A,B) from now on, be the collection of target information at node A and its transmission to node B where the weapon system is located.

Attributes

In one of the definitions of C^3 systems [AFM 1-1, 1979] it is stated that "Command and control systems must provide the commander with communications networks that are reliable rapid, survivable and secure." The first three requirements motivated the definition of the attributes for this example: *reliability*, *time delay*, and *survivability*. A fourth one that characterizes the amount of information that can be transmitted between A and B is the *input flow*.

The attribute *Reliability* denotes the capability of the system to deliver a message from A to B when only the intrinsic, physical characteristics of the components (links) are taken into account. The relevant system primitive is the probability of failure, 1-p, of each link, where it is assumed that link failures are independent events. *Survivability* is defined as the ability of the network to continue functioning in the presence of jamming. The system survivability depends on the probability that the enemy attempts to jam a link (or links) and the probability that the link is jammed when attacked. Both reliability and survivability are special cases of availability. The distinction between reliability and survivability is that the former reflects the failure characteristic of components and the effect of the environment, while the latter models the effect on the network of the enemy's electronic warfare capability.

The attribute *Time Delay* introduces the notion of timeliness of the transmitted information and the rapidity with which it is transmitted. This attribute is critical because in many instances target acquisition by a weapon system depends on the speed with which tracking information is received from distant sensors. For this network, time delay between nodes A and B is defined as the sum of the delays in each link of a path from A to B. The time delay is related to the capacity of each link. Therefore, link capacity is a system primitive.

Input Flow is defined as the amount of data transmitted from A to B. The underlying assumption is that as more tracking data are collected and transmitted to the weapon system, the target acquisition is improved.

The input flow that can be transmitted depends on the link capacities and the network topology; it depends also on the time delay, i.e., there is an interrelationship between delay and flow.

Let the mission be the protection of platforms located at the network nodes by weapon systems located at node B, where the sensors are located in a platform denoted by node A. Therefore, the objective of the platform at A is to detect and identify enemy targets and communicate that information to the weapon systems. The objective of node B is to destroy at least m percent of the enemy targets while suffering no more than n percent losses. Therefore, mission primitives are the level of forces of the two opponents, the single shot kill probabilitites, the time interval between salvos, the radius of uncertainty in locating a target, and the relative velocity between the targets and the weapon systems. With these primitives, it is necessary to determine conditions on the attributes that will imply the success of the assigned mission. Now that the overall situation has been described, the seven steps of the methodology can be applied.

Step 1 and 2: System Attributes

Structural analysis models based on engineering reliability theory [Barlow and Proschan, 1975] and network theory [Ford and Fulkerson, 1962] can be used to model the reliability attribute and compute its value.

Let x_i be a binary variable indicating whether link i is functioning ($x_i=1$) or has failed ($x_i=0$). Similarly, the binary variable ϕ specifies the state of the communication between nodes A and B. If the state of the communication is determined completely by the state of the links, then

$$\phi = \phi(x_1, x_2, \ldots, x_{13}) \qquad (9)$$

The function ϕ is called the structure function of the communication pair A,B. If p_i denotes the probability the link i is functioning, i.e.,

$$p_i = \text{prob}\ (x_i=1)$$

then the reliability index R is defined as the expected value of the structure function:

$$R = E\ [\phi(x_1, x_2, \ldots, x_{13})] \qquad (10)$$

For simplicity, let all the link failure probabilities be equal. Then the reliability index for the pair A,B in the network defined in Figure 2 is given as follows:

$$R = h_{m_4}(p)\ [1-(1-h_{m_2}(p))(1-h_{m_1}(p)\ h_{m_3}(p))] \qquad (11)$$

where

$$h_{m_1} = h_{m_2} = 1 - (1-p)^2$$

$$h_{m_3} = 2p^2 + 2p^3 - 5p^4 + 2p^5$$

$$h_{m_4} = 1 - (1-p^2(2-p)^2)\ (1-p).$$

The failure probabilities of the links are likely to vary with time. Since the p takes values in the interval [0,1], it follows that R, a continuous function defined on a closed set, takes minimum and maximum values. Furthermore, if p takes values in the subinterval [a,b] then R takes its minimum value R_{min} for p=a and its maximum R_{max} for p=b. Therefore, the reliability index R in eq. (11) is an increasing function of its argument p. If the bounds a and b are known, then the system reliability is bounded by

$$R_{min} \leq R \leq R_{max} \qquad (12)$$

While survivability depends on totally different primitives, the analysis is identical with that for reliability, but with the probability p replaced by

$$1 - e_i q_i$$

where e_i is the probability that the enemy attacks link i and q_i is the probability that link i is jammed when attacked. If the probability of survival of a link takes values in the interval [a',b'], it follows that the survivability index is bounded as follows:

$$S_{min} \leq S \leq S_{max} \qquad (13)$$

Queueing theory is used to model the time delay in the communications network. Specifically, the M/M/1 model [Schwartz, 1977] was used to determine the delay in transmitting packets from A to B. Let the capacity of each link in the network of Figure 2 be denoted by C_k, with k = 1,2,...,13. There are thirty different paths that can be chosen to transmit a packet from A to B and, therefore, thirty time delays, one for each path, can be computed. If path π_j is chosen, then the total delay along this path is [Bouthonnier, 1982]

$$T_{\pi_j} = \sum_{C_k \in \pi_j} \frac{1}{\mu C_k - F} \qquad (14)$$

where $1/\mu$ is the mean number of bits per packet and F is the input flow from A to B. Clearly, there will be a minimum and maximum delay over the thirty paths. So, depending on the routing algorithm chosen, the delay T may be bounded as follows:

$$T_{min} \leq T \leq T_{max} \qquad (15)$$

Now let all the link capacities be equal to C and let C vary between C_{min} and C_{max}. Then, for the network of Figure 2, the total delay from A to B satisfies [Bouthonnier, 1982]:

$$\frac{2}{\mu\, C_{max} - F} \leq T \leq \frac{6}{\mu\, C_{min} - F} \qquad (16)$$

The last condition relates two of the attributes, *Time Delay* and *Input Flow*. In order to normalize these attributes so that they vary between 0 and 1, the following scaling factors are introduced:

$$T^{*} = \text{maximum duration of mission}$$

$$F^{*} = \mu\, C_{max} \qquad (17)$$

Then the normalized attributes are

$$t = T/T^{*} \qquad (18)$$

and

$$K = F/F^{*} \qquad (19)$$

and relation (16) takes the form

$$\frac{2/T^{*}F^{*}}{1-k} \leq t \leq \frac{6/T^{*}F^{*}}{\dfrac{C_{min}}{C_{max}} - K} \qquad (20)$$

Thus, inequalities (13), (14), and (20) define the system locus L_s in the four-dimensional unit hypercube.

Steps 3 and 4: Mission Attributes

Let x(t) denote the number of blue forces and y(t) the number of red forces. The desirable conditions for blue are that at the end of the mission (at time T^{*}),

$$x(T^{*})/x(0) \geq n \qquad (21)$$

$$y(T^{*})/y(0) \leq m \qquad (22)$$

where n and m are positive numbers in the interval [0,1). A model is needed that describes the engagement. For this example, the Lanchester combat model that describes the "salvo fire" engagement was chosen for its simplicity rather than its realistic depiction of naval engagements. War games or extensive simulations could be used to analyze the mission in some detail and obtain realistic estimates of the requirements.

In the "salvo fire" engagement model each blue (red) unit fires every t_x (resp. t_y) time units at random at red (blue) units. Let $p_x (p_y)$ be the single shot probability of kill of a red (blue) unit by a blue (red) unit. If the single shot probabilities are small [Mangulis, 1980] then the Lanchester model reduces to

$$x = -\,(p_y/t_y)\; y = -\,ay \qquad (23)$$

$$y = -\,(p_x/t_x)\; x = -\,bx \qquad (24)$$

where the ratios of the kill probability to the interval between salvos denote the attrition rates a and b, respectively.

Solution of the differential equations (23), (24) leads to the "square-law" attrition process:

$$ay^2(t) - b\,x^2(t) = ay^2(0) - bx^2(0) \qquad (25)$$

Substitution of conditions (21) and (22) in (25) yields a condition on the attrition rates:

$$\frac{b}{a} \geq \frac{1-m^2}{1-n^2}\;\frac{y^2(0)}{x^2(0)} \qquad (26)$$

The attrition rate b was defined in (24) as p_x/t_x. Let r be the kill radius of blue's weapon system and let ρ denote the radius of uncertainty in locating red targets. Then

$$p_x = \pi r^2/\pi\rho^2 \qquad (27)$$

The value of ρ depends not only on the surveillance systems, but also on the ability of the network to transmit surveillance data about a moving target accurately and quickly to node B. The radius of uncertainty is assumed to be given by the following function of S,R,F, and T:

$$\rho = \frac{2}{S+R}\left[10c(1-.9F) + vT\right] \qquad (28)$$

where c is the radius of uncertainty due to the surveillance system alone and v is the relative speed to the red target.

Introduction of the normalized variables K and t, substitution of (27) and (28) in (26), and some algebraic manipulations yield the following requirement for the mission attributes:

$$S + R + c_1 K - c_2 t > c_3 \qquad (29)$$

where c_1, c_2, and c_3 are coefficients dependent on a,m,n,c,T*,F*,x(0) and y(0).

The mission locus, L_m, is defined then as the portion of the four-dimensional unit hypercube bounded by the hyperplane (29).

Step 5: The System and Mission Loci

In the previous four steps, the inequalities defining the two loci were derived. Numerical values must be selected now so that the loci can be specified completely and the assessment of effectiveness carried out.

Let the probability of a link failing, 1-p, range from 0.607 to 0.630 and let the probability that a link will be jammed vary over the same range. Then, inequalities (12) and (13) become:

$$0.4 \leq R \leq 0.45 \qquad (30)$$

$$0.4 \leq S \leq 0.45 \qquad (31)$$

while (20) becomes

$$\frac{0.1}{1-k} \leq t \leq \frac{0.3}{0.7-k} \qquad (32)$$

for T* F* =5 and C_{min}/C_{max} =0.7. Analysis of (32) shows that

$$0.1 \leq t \leq 1$$

and

$$0 \leq K \leq 0.9,$$

i.e., the normalized delay is at least 0.1 and the input flow cannot exceed 0.9. The locus L_s is depicted graphically in terms of three three-dimensional projections, Figs. 3, 4, and 5. In the third figure the unspecified axis is either S or R.

The mission locus, L_m, is defined by the inequality

$$S + R + K - t > 1$$

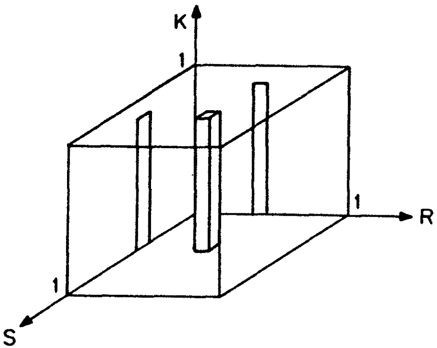


Fig. 3. Projection of L_s in (S,R,K) Space

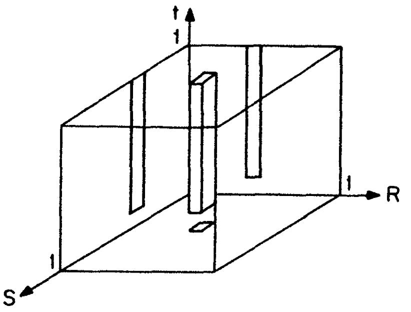


Fig. 4. Projection of L_s in (S,R,t) Space

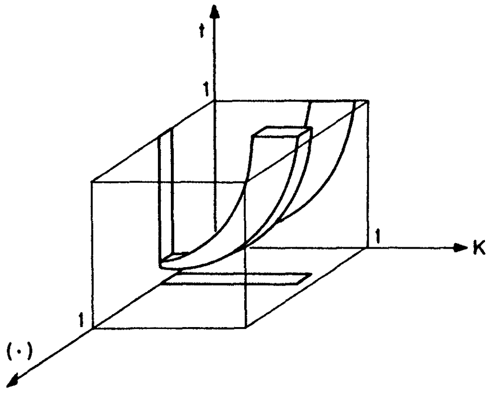


Fig. 5 Projection of L_s in (S,K,t) or (R,K,t) Space

where the coefficients c_1, c_2, and c_3 have been set equal to unity. The interrelationship between the system and the mission loci is shown in Figures 6, 7, and 8. Clearly, the two loci represent solids in four-dimensional space and, furthermore, the two solids intersect.

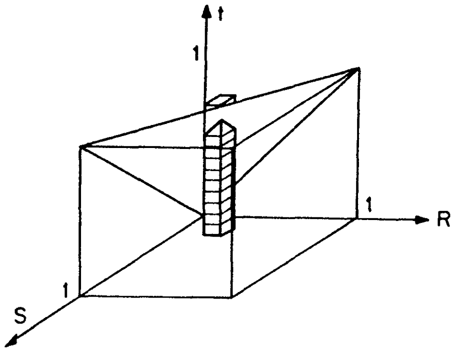


Fig. 6. Intersection of L_s and L_m in (S,R,t) Space

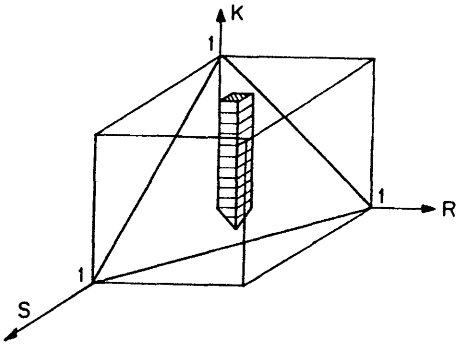


Fig. 7. Intersection of L_s and L_m in (S,R,K) Space

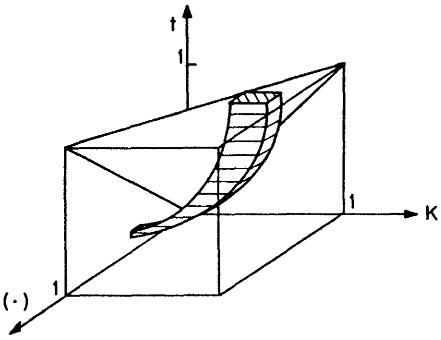


Fig. 8. Intersection of L_s and L_m in (S,K,t) or (R,K,t) Space

Step 6: Effectiveness Measures

Many different measures may be used to evaluate and compare the system and the mission locus. Let the first one considered be the volume:

$$V_1 = \iiiint dS\, dR\, dK\, dt$$

Then, the volume of the system locus can be computed analytically

$$V_1(L_s) = 0.131 \times 10^{-2}$$

The volume of the intersection of the system and mission loci can also be computed analytically

$$V_1(L_s \cap V_t) = 0.201 \times 10^{-3}$$

A class of measures based on volumetric comparisons is one defined by

$$V_2 = \int\int\int\int w(S,R,K,t)\ dS\ dR\ dK\ dt$$

Let

$$w(S,R,K,t) = (S+R)\ K$$

Then the system locus measure can be computed analytically

$$V_2(L_s) = 0.516 \times 10^{-3}$$

while the measure of the intersection is computed numerically

$$V_2(L_s \cap L_m) = 0.106 \times 10^{-3}$$

The partial measures of effectiveness are computed according to eq. (6):

$$E_1 = \frac{0.201 \times 10^{-3}}{0.131 \times 10^{-2}} = 0.15$$

$$E_2 = \frac{0.106 \times 10^{-3}}{0.516 \times 10^{-3}} = 0.205$$

Step 7: Systems Effectiveness

The two partial measures, E_1 and E_2, can be combined into a global measure of system effectiveness. First, however, the measures E_i should be mapped to the measures $\tilde{E}_i$ that range from zero to infinity. The function used is

$$\tilde{E} = \tan \frac{\pi}{2}\ E$$

Then

$$\tilde{E}_1 = 0.24 \quad ; \quad \tilde{E}_2 = 0.333$$

Finally, for the utility function

$$\hat{E} = A \cdot E_1^{\alpha_1} \cdot E_2^{\alpha_2}$$

with $A = 1$, $\alpha_1 = 0.5$, and $\alpha_2 = 0.5$, the global measure takes the value

$$\hat{E} = 0.283$$

Thus, all steps of the methodology were carried out and a measure of effectiveness for the specific communications network has been determined. If an alternative network is proposed, then the methodology can be applied to the second network and a measure of effectiveness obtained. Comparison between the two networks using the effectiveness measures (as well as the attributes) would be straightforward because both the attributes and the measures of effectiveness are commensurate.

CONCLUSIONS

A new approach for assessing the effectiveness of C^3 systems has been presented. The key idea is to relate, in a quantitative way, the capabilities of a C^3 system to the requirements of the mission(s) that the military unit or organization has been assigned to execute. Each step of the methodology (specification of system and mission primitives, definition of attributes, modeling the system and the mission, constructing the two loci) brings into sharper focus qualitative information on what the system is intended to do, where it is intended to be used, and how it is intended to be used. Posing and addressing these questions is essential for assessing C^3 systems which are complex, often large scale, service delivery systems.

REFERENCES

AFM 1-1 (1979). Functions and Basic Doctrine of the USAF, GPO, Washington, DC.

Barlow, R., and F. Proschan (1975). Statistical Theory of Reliability and Life-Testing: Probability Models, Holt, Rinehart and Winston, New York.

Bouthonnier, V. (1982). System Effectiveness Analysis for Command and Control. SM Thesis. Report LIDS-TH-1231. Laboratory for Information and Decision Systems, MIT, Camb., MA.

Debreu, G. (1958). Theory of Value: An Axiomatic Analysis of Economic Equilibrium. Wiley, New York,

Dersin, P. and A.H. Levis (1981). Large Scale Systems Effectiveness Analysis. Report LIDS-FR-1072. Laboratory for Information and Decision Systems, MIT, Cambridge, MA.

Fink, L. (1980). System Effectiveness Analysis: In Systems Engineering for Power, Program Report DOE-RA-0052-01, US Dept. of Energy, Washington, DC.

Ford, L.K. and D. K. Fulkerson (1962). Flows in Networks. Princeton University-Press, Princeton, New Jersey.

Mangulis, V. (1980). Lanchester-Type Equations of Combat which Account for Multiple Killing Hits. Oper. Res., 28, pp. 560-569.

Phlips, L. (1974). Applied Consumption Analysis, North-Holland/American Elsevier, New York.

Schwartz, M. (1977). Computer Communication Network Design and Analysis. Prentice-Hall, Englewood Cliffs, New Jersey.

MEASURES-OF-EFFECTIVENESS FOR COMMAND, CONTROL, AND COMMUNICATIONS COUNTERMEASURES

G.R. LINSENMAYER

Westinghouse Electric Corporation, Defense and Electronics Center
P.O. Box 746, MS 434
Baltimore, Maryland 21203

ABSTRACT

A number of operationally significant MOE's are suggested and defined for the evaluation of C^3CM techniques and systems. A primary aim has been to suggest MOE's which reflect the effectiveness and usefulness of C^3CM systems and equipments in relation to the engagements and operations in which they are used.

1. INTRODUCTION

The intent of this technical note is to suggest certain "measures of effectiveness" (MOE's) for the evaluation of C^3CM systems effectiveness. A primary aim has been to suggest MOE's which reflect the operational utility of C^3CM systems and equipments, indicating their effectiveness and usefulness in relation to the engagements and operations in which they are used. Such MOE's complement and extend other "measures of performance" i.e., measures of the technical performance of equipments being evaluated.

A C^3CM system can be viewed as consisting of information inputs, a C^3 structure, and C^3CM assets. The C^3 structure supports the data gathering, situation assessment, and command/control functions which provide (hopefully) for the effective use of these assets. Measures of effectiveness provide a basic conceptual reference for considering the goals and requirements of this C^3 structure, and for evaluating how effectively specific C^3 structures, and their associated C^3CM assets, perform as a C^3CM system.

Ideally, definitions of measures of effectiveness would lead techniques of system synthesis which would allow the design of optimal or nearly optimal systems. In practice, criteria of optimality which are mathematically tractable may not agree, or may agree only in certain aspects, with intuitive ideas of "effectiveness". Defined measures of effectiveness can be valuable in either guiding efforts toward deriving useful optimality criteria, or in providing for tradeoff and effectiveness evaluations of systems designed according to intuitively less satisfying but mathematically tractable goals.

The MOE's which are suggested here are based on a generalized "defensive C^3CM engagement" (section 3) which includes RED surveillance and targeting of BLUE targets, and weapon launch, midcourse, and terminal acquisition phases. It seems likely that similar MOE's could also be developed for other engagement situations. Only MOE's indicative of technical effects are considered here; that is, no attempt is made to predict intangible C^3CM effects such as "deceiving the enemy commander."

2. PERFORMANCE AND EFFECTIVENESS

In this memorandum, the term "performance" is used to indicate the technical performance of equipment, while the term "effectiveness" will be used to indicate some degree of operational utility of equipment or systems.

Thus the term "measure of performance" (MOP) denotes a measure of the technical performance of some particular equipment or system. Examples include "miss distance," "detection range," and "communication capacity."

True "measures of effectiveness" should reflect the operational utility of equipments and systems. They should attempt to measure qualities which a commander would perceive as having a more-or-less direct relation to the outcome of engagements and operations in which he is involved. The terms "operational utility" and "more-or-less direct" are somewhat nebulous, of course, and do not really define "measure-of-effectiveness;" however, they are useful working guides for attempting to suggest and define specific measures of effectiveness for C^3CM systems.

Measures of effectiveness can be defined at various levels of activity in an engagement or operation. Two levels are considered here.

At the highest level, we consider measures of the outcomes of particular engagements or operations, or of classes of engagements and operations. Measures of effectiveness at this level will be called "Engagement Outcome" MOE's. Numerous examples of such MOE's appear to be relatively easy to suggest, such as:

RED losses (numbers, types, loss rates, ...)

BLUE losses (numbers, types, loss rates, ...)

RED/BLUE loss ratios
Number or fraction of RED (BLUE) targets successfully attacked;
Value of such targets
Gain or loss of RED (BLUE) territory
Depth of penetration of RED (BLUE) strike forces
.
.
.

At a more detailed operational level, we may postulate "Operational-Level" MOE's* to measure qualities of an engagement or operation which could be of use to a commander in assessing his operational situation and in evaluating the effectiveness of possible actions such as C^3CM on that situation. A potential advantage of MOE's defined at this level, as compared with engagement-outcome MOE's**, is that they might be more easily evaluated quantitatively. Further, evaluating engagement-outcome MOE's may require making specific assumptions (e.g., force effectiveness, terminal CM effects, command decisions and tactics) which might not be required to evaluate operational-level MOE's. By requiring fewer assumptions for their evaluation, operational-level MOE's would be correspondingly more "reliable" in that sense, in their application to particular situations of interest.

3. C^3 COUNTERMEASURES (C^3CM) EFFECTS

As a prelude to presenting a set of suggested MOE's for C^3CM systems, we first consider a generalized defensive C^3CM engagement, to identify typical classes of effects which C^3CM equipments may have on enemy C^3 systems. Figure 1 illustrates the generalized defensive C^3CM engagement considered here. RED maintains surveillance and intelligence operations directed toward the goal of locating and targeting certain BLUE forces. This information is processed in a RED C^3I network. When BLUE forces have been located and targeted, commands will be issued to appropriate weapon launchers, and weapons will be launched. A midcourse phase is assumed as the weapons proceed toward their targets. Subsequently, a target acquisition phase occurs as each weapon attempts to acquire a specific target and convert to a terminal attack.

BLUE countermeasures employed during the phases prior to terinal target acquisition are considered here to be C^3CM. CM's employed directly during terminal target acquisition or during final weapon attack are considered to be terminal CM's (e.g., range gate stealers, or heavy noise jamming of a terminal seeker). However, the effects of certain C^3CM's may also appear during the terminal target acquisition phase, so that there is some overlap between C^3CM impacts and terminal CM effects, as shown in the figure.

The classes of C^3CM effects which are considered here are:

delay
targeting dilution
uncertainty
saturation
control degredation
tempo

These effects are discussed briefly in the paragraphs following. Specific C^3CM MOE's are then defined in section 4.

Delay. C^3CM applications by BLUE against RED* may delay RED in achieving specific operational capabilities. Such delays could result from delays caused by CM's against RED's communication subsystems, delays caused by excessively loading and/or saturating RED's information processing and command decision subsystems, and lost or misleading data from RED's surveillance sensor subsystem.

---

* These MOE's correspond to "2nd-order" MOE's as described by Wohl (MITRE report MTR-8217, 15 Jan. 1981). However, they are defined here in a C^3CM context, rather than in a TAC air operations context. They may also be related to Brandenburg's classes of information mobility and knowledge difference (NOSC TR 598, 5 Oct. 1980) as specific operational-level instances of these classes.

** The advantage of engagement-outcome MOE's is fairly obvious, in that they more directly correspond to possible operational goals and intentions of the engagement. However, the significance of any particular engagement-outcome MOE's may vary strongly depending on the operational goals of the engagement.

* BLUE, RED will be used to distinguish between the force applying C^3CM's and its opponent force.

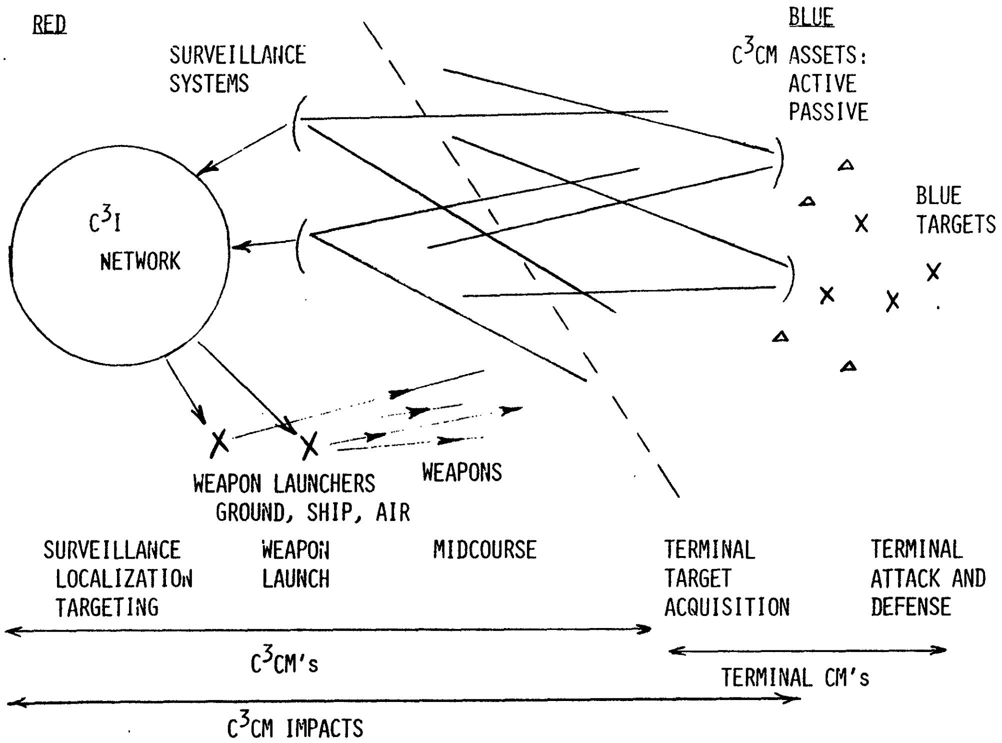


Figure 1. A Generalized Defensive C^3CM Engagement

Targeting dilution. The term "targeting dilution" will be used to indicate an increase in perceived numbers of BLUE targets, for example due to use of decoys, generation of false target signatures, or reduction of RED target ID capabilities, such that RED's available strike force effectiveness in relation to a set of desired BLUE targets is "diluted" through application against non-intended or false targets.

Uncertainty. C^3CM applications by BLUE against RED may cause other forms of uncertainty** in addition to target dilution. Three such categories of uncertainty to be considered are:

targeting uncertainty
uncertainty of BLUE force status and intentions
uncertainty in status and/or location of own (RED) forces

The application of CM's against RED surveillance subsystems can be a prime cause of such uncertainties. Such CM's can either operate directly (e.g., noise) or indirectly (e.g., saturation of RED tracking system capability). CM's against communications links from RED sensors to RED C^2 centers could also cause such uncertainties.

Saturation. Certain C^3CM systems may produce effects by saturating RED sensing, information processing, and/or command decision subsystems. In this memorandum, such saturation will be considered to be an intermediate effect, leading to other final effects such as delays and/or uncertainties as described previously.

Control degradation. "Control degradation" will be reserved to indicate a direct degradation or loss of control by RED over his own force elements, and will not include degradations in control or effectiveness of his force elements due indirectly to other effects such as targeting dilution and targeting uncertainty. Control degradation thus will be primarily an effect related to CM's against RED communications links between RED C^2 centers and subordinate centers and/or force elements. CM's against force element navigation systems

---

** RED may or may not be aware of this uncertainty; such awareness or lack of awareness would affect RED's tactical decisions. For example, knowing that his knowledge of BLUE location is uncertain, RED may allocate greater number of weapons or delay weapon commitment. Believing an erroneous location is true, RED may commit a weapon ineffectively.

and/or mid-course guidance systems can also produce control degradation.

Tempo*. The scenario assumed here is a "fixed batch" engagement, wherein a fixed number of BLUE forces are engaged by a "batch" of enemy weapons. Alternatively, a "continuing" scenario could be postulated, in which BLUE forces are continuously engaged by RED over some interval of time. In this case, degredations in the rate of RED operations (e.g., weapons launched per unit time) may be operationally significant. Such effects will not be considered here; however, it appears that the MOE's described in section 4 could be modified in appropriate ways to yield "per-unit-time" MOE's.

4. OPERATIONAL-LEVEL C^3CM MEASURES OF EFFECTIVENESS

This section suggests and defines a number of specific operational-level MOE's for C^3CM, based on the classes of effects described in section 3. In some cases, extensions and/or variations to the basic MOE's are also indicated.

4.1 DELAY

The time delay until specific levels of enemy C^3 capability are attained:

ΔT_{C3CAP}

Examples:

$\Delta T_{TAC.SIT}$ = Time delay from present until a defined "tactical situation" is obtained.

e.g.,

TAC.SIT = "Location of battlegroup forces not known"

"Location of CV approximately known"

"Targeting-quality location of CV known."

$\Delta T_{WPN.CAP}$(TGT.CLASS,WPN.SYS) = Time delay from present until a weapon system capability WPN.CAP (e.g., weapon launch, or terminal engagement) is attained against targets of TGT.CLASS using WPN.SYS.

In order to evaluate or estimate delays of this type, it is necessary to first determine the various components of delay and their interrelationships. The effects of each C^3CM considered on these components of delay must then be determined. Finally, the cumulative effects of the component delays must be evaluated to determine the total operational delay. PERT-type methods of analysis may be useful during this last step.

4.2 TARGETING DILUTION

Increase the apparent number of targets which a particular enemy weapons system must engage.

*

$$\mathrm{TD}\ (\mathrm{TGT.CLASS},\ \mathrm{WPN.SYS}) = \frac{N_2}{N_1}$$

TGT.CLASS is the class of potential targets (e.g., CV, HV).

WPN.SYS is the type of enemy weapons system assumed.

N_1 is the true number of platforms in TGT.CLASS

N_2 is the apparent number of platforms in TGT.CLASS.

Special classes -

TD_{ID}	The target dilution factor due to loss of or deceptive ID data, such that other platforms not in TGT.CLASS are indistinguishable from those in TGT.CLASS.
$TD_{FLS.TGT}$	The target dilution factor due to generation of apparent targets which are false targets.

The key to estimating or evaluating TD is the estimate of N_2. This estimate must be made by a technical evaluation of RED ID capability as affected by the particular C^3CM's to be considered.

4.3 UNCERTAINTY

BLUE force location uncertainty

At some future time, the ratio of area of possible BLUE force location to

---

* J.S. Lawson, "The Role of Time in a Command Control System", Proceedings of the Fourth MIT/ONR Workshop on Distributed Information and Decision Systems Motivated by Command-Control-Communications (C^3) Problems, Vol, IV (C^3 Theory), MIT LIDS-R-1159, October 1981.

* Here and in other MOE's defined, a variation could be to express "after" and "before" conditions as % change, rather than by ratio. Also, the absolute numbers N_2 and N_1 may be of interest, as well as their ratio.

total theater area; relative to baseline.

$$\text{BLU(TGT.CLASS,t)} = \frac{A_2/T}{A_1/T} = \frac{A_2}{A_1}$$

$A_2(t)$ = area of possible locations of BLUE force targets in TGT.CLASS at time t, for C^3CM condition being evaluated.

$A_1(t)$ = area of possible locations of BLUE force targets in TGT.CLASS at time t, for "baseline" C^3 condition

T = area of theater of operations.

An estimate or evaluation of BLU must include RED surveillance update frequency and BLUE force maneuver capability. Lawson* presents an example of such an analysis.

Target Concealment

Conceal BLUE targets from RED surveillance. This is the most complete form of uncertainty possible.

$$\text{TC (TGT.CLASS)} = \frac{N_1}{N_2}$$

N_1 is the true number of platforms in TGT.CLASS

N_2 is the apparent number of platforms in TGT.CLASS detected by RED surveillance.

To estimate or evaluate TC a technical analysis or estimate must be made of the capability of RED surveillance/intelligence to detect and identify TGT.CLASS platforms. Significant factors to be included in such an analysis include not only C^3CM's to be considered, but also scenario environment (e.g., rain, clutter/background, ...).

Targeting Allocation Effectiveness Loss

A measure of the decrease in effective allocation of RED forces against BLUE targets. Causes include concealment of BLUE targets and RED uncertainty of BLUE-force status, own-force status and/or locations.

A simple measure may be defined as:

$$\text{TAEL (TGT,CLASS, WPN.SYS)} = \frac{N_{A1}}{N_{A2}}$$

N_A = number of platforms in TGT.CLASS against which weapons of WPN.SYS. are allocated.

1,2 refer to without and with C^3CM

Note that TAEL may include target concealment factors, but may also include other target degradation factors as well.

However, the effectiveness of weapons allocation depends upon the numbers of weapons allocated to the targets, as well as the number of targets against which weapons are allocated. To consider this factor, let S_A be the set of TGT.CLASS targets against which weapons are allocated. Let subscripts 1,2 designate without and with C^3CM; and let X designate any particular enemy allocation control concept assumed. Let Υ represent any particular element of S_{A1} or S_{A2}. Usually, $S_{A2} \subseteq S_{A1}$. Let $n_{xi}(\Upsilon)$ represent the number of weapons allocated against BLUE target Υ by the RED C^3 system, for any given X and for i = 1, 2. Finally, let P_{ssk} be the conditional single-shot probability of kill given a valid allocation of WPN.SYS weapon against TGT.CLASS target.

Then, the expected number of TGT.CLASS targets destroyed, without or with C^3CM, is:

$$\bar{N}_{ki} = \sum_{\Upsilon \in S_{Ai}} \left[1 - (1 - P_{ssk})^{n_{xi}(\Upsilon)}\right]$$

We could then define TAEL as:

$$\text{TAEL'(TGT.CLASS, WPN.SYS.)} = \frac{\bar{N}_{k1}}{\bar{N}_{k2}}$$

In order to evaluate TAEL', assumptions must be made about P_{ssk} and X. Assumptions about X must define both the sets S_{Ai} and the numbers of weapons allocated, $n_{xi}(\Upsilon)$; in contrast, for TAEL, only the numbers N_{Ai} must be defined. Because of these complexities, TAEL might be preferred.

A more simple definition of TAEL, chosen to include consideration of the number of weapons which RED launches, is

$$TAEL'' = \left(\frac{W_1}{W_2}\right)^{\alpha} \left(\frac{N_{A1}}{N_{A2}}\right)^{\beta}$$

where W_1, W_2 are the assumed total numbers of weapons allocated and launched, corresponding to baseline and C^3CM cases, and α and β are weighting factors to be determined.

Intuitively, for α and $\beta > 0$, a C^3CM which causes RED to be able to successfully allocate fewer weapons ($W_2 < W_1$) or to be able to allocate against fewer targets ($N_{A2} < N_{A1}$) will degrade RED effectiveness and thus improve BLUE's position (TAEL'' > 1).

---

* J.S. Lawson, Op.Cit.

The estimation or evaluation of TAEL requires estimating how RED will perform weapon allocation, both without and in the presence of C^3CM's being considered. Based upon such an understanding, technical estimates must be made of the W and N_A parameters. If TAEL' is chosen as a definition of TAEL, kill probabilities must also be estimated.

Targeting conversion failure

Increase the uncertainty of the predicted target position at the termination of weapon midcourse, so that the weapon fails to convert to a terminal attack.*

$$\text{TCF(TGT.CLASS, WPN.SYS)} = \frac{P_{c1}}{P_{c2}}$$

P_{c1} = geometric** probability of successful conversion without C^3CM.

P_{c2} = geometric** probability of successful conversion with C^3CM.

This definition may not be sufficiently complete from an operational point of view. To see this, consider as an example a case where a "single-shot" geometric probability of conversion is $P_{c1} = .9$, and where RED achieves a "multi-shot" value of 0.99 by using two weapons. Suppose C^3CM degrades P_{c1} to .6; thus TCF = .9/.6 = 1.5. However, the "multi-shot" value, assuming two shots, is only 0.99/0.84 = 1.18. Furthermore, suppose that RED knows that C^3CM is employed and decides to fire 3 shots; then the ratio would become only 0.99/0.936 = 1.06.

An alternate definition of TCF is

$$\text{TCF'(TGT.CLASS,WPN.SYS)} = \frac{(N_1/N_{A1})}{(N_2/N_{A2})}$$

N_1 = number of platforms in TGT.CLASS which would be successfully attacked⊕ for "baseline" C^3 conditions.

N_2 = number of platforms in TGT.CLASS which would be successfully attacked⊕ for C^3CM condition being evaluated.

The ratios (N_i/N_{Ai}) are analagous to P_{ci}, except that they include consideration of allocations of multiple weapons per target.

In the example above, TCF' = .99/.84 = 1.18. If RED can fire 3 shots per target, and is assumed to choose to do so, TCF' = .99/.936 = 1.06.

TCF and TCF' may be estimated based upon geometric models.

4.4 CONTROL DEGREDATION

Weapon Delivery Effectiveness Loss

A measure of a decrease in effective control by RED over allocated forces such that reduced numbers are able to reach terminal phase.* This is analagous to TCF, except that control degredation factors are considered instead of geometric factors. Thus,

$$\text{WDEL(TGT.CLASS, WPN.SYS)} = \frac{P_{c1}}{P_{c2}}$$

$$\text{WDEL'(TGT,CLASS, WPN.SYS)} = \frac{(N_1/N_{A1})}{(N_2/N_{A2})}$$

where P_{ci} and N_i reflect control degredation factors rather than geometric factors.

5.0 CONCLUSIONS

A number of "first-cut" suggestions have been made of MOE's for the evaluation of C^3CM operational effectiveness.

Only brief consideration is given here to computing quantitative values for these MOE's. In particular situations, several of the MOE's suggested (e.g., TD, BLU) appear to be reasonably straightforward to evaluate. In other cases (e.g., TAEL), the complexities of underlying scenario situations, and the need for assumptions such as kill probabilities and/or RED tactics, make evaluation relatively complex.

The next step required in developing these MOE's is to postulate specific scenarios and investigate detailed specific definitions and evaluation methods, in order to develop an understanding of the practicalities and potential usefulness of these suggested MOE's.

---

* Terminal-phase CM's are not considered here.

** i.e., control degredation factors (see WDEL) should be ignored here.

⊕ Consider only geometric factors here; i.e., control degredation factors (see WDEL) and terminal kill probabilities should be ignored.

* Terminal CM's are not considered here.

DECISIONMAKING ORGANIZATIONS WITH ACYCLICAL INFORMATION STRUCTURES*

A. H. Levis

K. L. Boettcher

Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, Mass., USA

Abstract. An analytical model of a team of well-trained human decisionmakers executing well-defined decisionmaking tasks is presented. Each team member is described by a two-stage model consisting of a situation assessment and a response selection stage. An information theoretic framework is used in which bounded rationality is modeled as a constraint on the total rate of internal processing by each decisionmaker. Optimizing and satisficing strategies are derived and their properties analyzed in terms of organizational performance and individual workload. The results are applied to the analysis of two three-person organizations. The relevance of this approach to the design and evaluation of alternative information structures for command control and communications (C^3) systems is discussed.

INTRODUCTION

A command control and communications (C^3) system is defined as the collection of equipment and procedures used by commanders and their staff to process information, arrive at decisions, and communicate these decisions to the appropriate units in the organization in a timely manner. Implicit in this definition is the notion that the role of the human decisionmaker is central to the design of organizations and the C^3 systems that support them. A basic model of an interacting decisionmaker, appropriate for a narrow but important class of problems, was introduced by Boettcher and Levis (1982). In a second paper, Levis and Boettcher (1982) considered the modeling of organizations consisting of two decisionmakers that form a team. In this paper, the methodology is extended to the analysis and evaluation of teams with acyclical information structures. Two three-person organizations are used to illustrate the approach.

The basic assumption in designing organizations is that a given task, or set of tasks, cannot be carried out by a single decisionmaker because of the large amount of information processing required and the severe time constraints present in a tactical situation. In designing an organizational structure for a team of decisionmakers, two issues need to be resolved: who receives what information and who is assigned to carry out which decisions. The resolution of these issues depends on the limited information processing rate of individual decisionmakers and the tempo of operations. The latter reflects the rate at which tasks are assigned to the organization and the interval allowed for their execution.

An information theoretic framework is used for both the modeling of the individual decisionmaker and of the organization. Information theoretic approaches to modeling human decisionmakers have a long history (Sheridan and Ferrell, 1974). The basic departure from previous models is in the modeling of the internal processing of the inputs to produce outputs. This processing includes not only transmission (or throughput) but also internal coordination, blockage, and internally generated information. Consequently, the limitations of humans as processors of information and problem solvers are modeled as a constraint on the total processing activity. This constraint represents one interpretation of the hypothesis that decisionmakers exhibit bounded rationality (March, 1978).

The task of the organization is modeled as receiving signals from one or many sources, processing them, and producing outputs. The outputs could be signals or actions. Implicit in this model of the organization's function is the hypothesis that decisionmaking is a two-stage process. The first is the assessment of the situation (SA) of the environment, while the second is the selection of a response (RS) appropriate to the situation.

The input signals that describe the environment may come from different sources and, in general, portions of the signals may be received by different members of the organization. It has been shown by Stabile, Levis

---

*This work was supported by the Air Force Office of Scientific Research under grant AFOSR-80-0229.

and Hall (1982) that the general case can be modeled by a single vector source and a set of partitioning matrices that distribute components of the vector signal to the appropriate decisionmakers within the organization. This model is shown in Fig. 1, where the input vector is denoted by X and takes values from a finite alphabet $\mathscr{X}$. The partitions x^i may be disjoint, overlapping or, on occasion, identical.

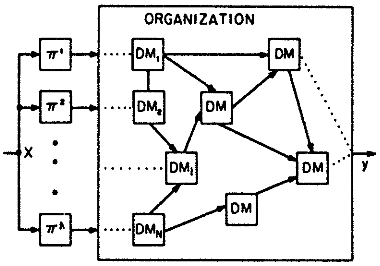


Fig. 1 The problem of information structures for organizations.

Many classes of organizational structures can be represented by Fig. 1. Consideration in this paper will be restricted to structures that result when a specific set of interactions is allowed between team members. In this case, each team member is assigned a specific task, whether it consists of processing inputs received from the external environment or from other team members, *for which he is well trained and which he performs again and again for successively arriving inputs.* In general, a member of the organization can be represented by a two-stage model as shown in Fig. 2. First, he may receive signals from the environment that he processes in the situation assessment (SA) stage to determine or select a particular value of the variable z that denotes the situation. He may communicate his assessment of the situation to other members and he may receive their assessments in return. This supplementary information may be used to modify his assessment, i.e., it may lead to a different value of z. Possible alternatives of action are evaluated in the response selection (RS) stage. The outcome of this process is the selection of a local action or decision response y that may be communicated to other team members or may form all or part of the organization's response. A command input from other decisionmakers may affect the selection process. A further restriction is introduced in that the information structures be acyclical.

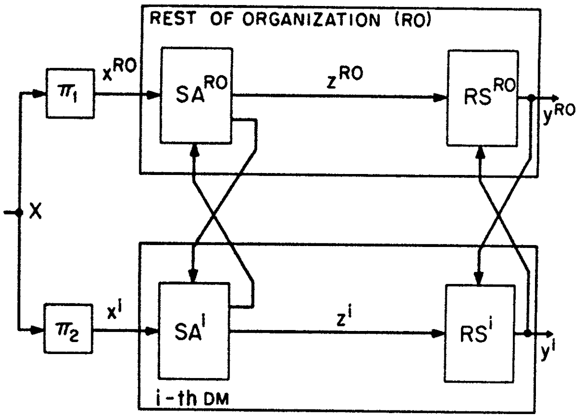


Fig. 2 Allowable team interactions

The overall mapping between the stimulus (input) to the organization and its response (output) is determined by the internal decision strategies of each decisionmaker. The total activity of each DM as well as the performance measure for the organization as a whole are expressed then in terms of these internal decision strategies. For each set of admissible internal decision strategies, one for each DM, a point is defined in the performance-workload space. The locus of all such points is characteristic of the organizational structure. Once the locus has been constructed, it is then possible to analyze the effects of the bounded rationality constraints on the organization's performance when either optimizing or statisficing behavior is assumed.

In the next section, the model of the interacting organization member is reviewed. In the third section the model of a team with acyclical information structures is described analytically. In the fourth section, the optimal and the satisficing decision strategies for the two three-person organizations are obtained and analyzed.

MODEL OF THE ORGANIZATION MEMBER

The complete realization of the model for a decisionmaker (DM) who is interacting with other organization members and with the environment is shown in Fig. 3. The detailed description and analysis of this model, as well as its relationship to previous work, notably that of Drenick (1976) and Froyd and Bailey (1980), has been presented in Boettcher and Levis (1982). Therefore, only concepts and results needed to model the organization are described in this section. The presentation is similar to that in Levis and Boettcher (1982).

Let the organization receive from the environment a vector of symbols, X'. The DM receives x which is a noisy measurement of a portion, x', of X'. The vector x takes values from a known finite alphabet according to the probability distribution p(x). The quantity

$$H(x) = -\sum_{x} p(x) \log_2 p(x) \qquad (1)$$

is defined to be the entropy of the input (Shannon and Weaver, 1949) measured in bits per symbol generated. The quantity H(x) can also be interpreted as the uncertainty regarding which value the random variable x will take. If input symbols are generated every τ seconds on the average, then τ, the mean symbol interarrival time, is a description of the tempo of operations (Lawson,1981)

The conditional entropy is defined as

$$H_x(z) = -\sum_x p(x) \sum_z p(z|x) \log_2 p(z|x) \qquad (2)$$

The situation assessment stage consists of a finite number U of procedures or algorithms f_i that the DM can choose from to process the measurement x and obtain the assessed situation z. The internal decisionmaking in this stage is the choice of algorithm f_i to process x.

Therefore, each algorithm is considered to be active or inactive, depending on the internal decision u. In this paper, it is assumed that the algorithms f_i are deterministic. This implies that once the input is known and the algorithm choice is made, all other variables in the first part of the SA stage are known. Furthermore, because no learning takes place *during* the performance of a sequence of tasks, the successive values taken by the variables of the model are uncorrelated, i.e., the model is memoryless. Hence, all information theoretic expressions appearing in this paper are on a per symbol basis.

The vector variable $\underline{z}'$, the supplementary situation assessment received from other members of the organization, combines with the elements of z to produce $\bar{z}$. The variables z and $\bar{z}$ are of the same dimension and take values from the same alphabet. The integration of the situation assessments is accomplished by the subsystem S^2 which contains the deterministic algorithm A.

If there is no command input vector $\underline{v}'$ from other organization members, then the response selection strategy $p(v|\bar{z})$ specifies the selection of one of the algorithms h_j that map $\bar{z}$ into the output y. The existence of command input $\underline{v}'$ modifies the decisionmaker's choice v. A final choice $\bar{v}$ is obtained from the function $b(v,\underline{v}')$. The latter defines a protocol according to which the command is used, i.e., the values of $\bar{v}$ determined by $b(v,\underline{v}')$ reflect the degree of option restriction effected by the command. The overall process of mapping the assessed situation $\bar{z}$ and the command input $\underline{v}'$ into the final choice $\bar{v}$ is depicted by subsystem S^3 in Fig. 3. The result of this process is a response selection strategy $p(\bar{v}|\bar{z}\underline{v}')$ in place of $p(v|\bar{z})$.

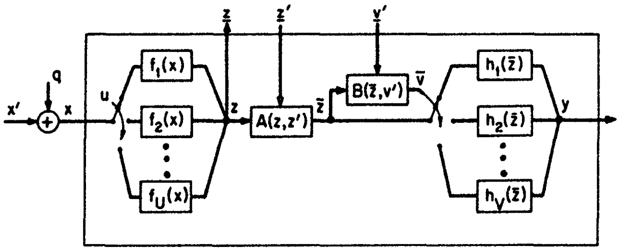


Fig. 3 Single interacting decisionmaker model

The model of the decisionmaking process shown in Fig. 3 may be viewed as a system S consisting of four subsystems: S^1, the first part of the SA stage; S^2; S^3; and S^4, the second part of the RS stage. The inputs to this system S are $x, \underline{z}'$, and $\underline{v}'$ and the outputs are y and the situation assessment transmitted to other DMs. The second output consists of a set of z_i vectors, one for each interacting DM. For notational simplicity, these vectors will be denoted by a single vector $\underline{z}$ consisting of the concatenation of the z_i's. Furthermore, let each algorithm f_i contain α_i variables denoted by

$$W^i = \left\{ w_1^i, w_2^i, \ldots, w_{\alpha_i}^i \right\} \qquad i = 1,2,\ldots,U \quad (3)$$

and let each algorithm h_j contain α_j' variables denoted by

$$W^{U+j} = \left\{ w_1^{U+j}, \ldots, w_{\alpha_j'}^{U+j} \right\} \quad j = 1,2,\ldots,V \qquad (4)$$

It is assumed that each algorithm has a self-contained set of variables and that when one algorithm is active, all others are inactive. Consequently,

$$W^i \cap W^j = \emptyset \quad \text{for } i \neq j$$

$$\forall i,j \in \{1,2,\ldots,U\} \text{ or } \{1,2,\ldots,V\} \qquad (5)$$

The subsystem S^1 is described by a set of variables

$$S^1 = \{u, W^1, \ldots, W^U, \underline{z}, z\};$$

subsystem S^2 by

$$S^2 = \{W^A, \bar{z}\};$$

subsystem S^3 by

$$S^3 = \{W^B, \bar{v}\};$$

subsystem S^4 by

$$S^4 = \{W^{U+1}, \ldots, W^{U+V}, y\}.$$

The mutual information or transmission or throughput between inputs $x, \underline{z}'$, and $\underline{v}'$ and outputs y and $\underline{z}$, denoted by $T(x,\underline{z}',\underline{v}':y,\underline{z})$ is a description of the input - output relationship of the DM model and expresses the amount by which the outputs are related to the inputs:

$$\begin{aligned} G_t &= T(x,\underline{z}',\underline{v}':y,\underline{z}) \\ &= H(x,\underline{z}',\underline{v}') + H(y,\underline{z}) - H(x,\underline{z}',\underline{v}',y,\underline{z}) \\ &= H(\underline{z},y) - H_{x,\underline{z}',\underline{v}'}(\underline{z},y) \qquad (6) \end{aligned}$$

A quantity complementary to the throughput G_t is that part of the input information which is not transmitted by the system S. It is called blockage and is defined as

$$G_b = H(x,\underline{z}',\underline{v}') - G_t \qquad (7)$$

In this case, inputs not received or rejected by the system are not taken into account.

In contrast to blockage is a quantity that describes the uncertainty in the output when the input is known. It may represent noise in the output generated within S or it may represent information in the output produced by the system. It is defined as the entropy of the system variables conditioned on the input, i.e.,

$$G_n = H_{x,\underline{z}',\underline{v}'}(u,W^1,\ldots,W^{U+V},W^A,W^B,\underline{z},z,\bar{z},\bar{v},y) \tag{8}$$

The final quantity to be considered reflects all system variable interactions and can be interpreted as the coordination required among the system variables to accomplish the processing of the inputs to obtain the output. It is defined by

$$G_c = T(u:w_1^1:\ldots:w_{\alpha_V'}^{U+V}:w_1^A:\ldots w_{\alpha_B}^B:z:\bar{z}:\bar{v}:\underline{z}:y) \tag{9}$$

The Partition Law of Information (Conant, 1976) states that the sum of the four quantities G_t, G_b, G_n, and G_c is equal to the sum of the marginal entropies of all the system variables (internal and output variables):

$$G = G_t + G_b + G_n + G_c \tag{10}$$

where

$$G = \sum_{i,j} H(w_i^j) + H(u) + H(z) + H(\bar{z}) + H(\underline{z}) + H(\bar{v}) + H(y) \tag{11}$$

When the definitions for internally generated information G_n and coordination G_c are applied to the specific model of the decisionmaking process shown in Fig. 3 they become

$$G_n = H(u) + H_{\bar{z}}(v) \tag{12}$$

and

$$\begin{aligned} G_c = &\sum_{i=1}^{U} [p_i g_c^i(p(x)) + \alpha_i \mathscr{H}(p_i)] + H(z) \\ &+ g_c^A(p(z)) + g_c^B(p(\bar{z})) \\ &+ \sum_{j=1}^{V} [p_j g_c^{U+j}(p(\bar{z}|\bar{v}=j)) + \alpha_j' \mathscr{H}(p_j)] + H(y) \\ &+ H(z) + H(\bar{z}) + H(\bar{v},\bar{z}) + T_z(x':\underline{z}') \\ &+ T_{\bar{z}}(x',\underline{z}':\underline{v}') \end{aligned} \tag{13}$$

The expression for G_n shows that it depends on the two internal strategies $p(u)$ and $p(v|\bar{z})$ even though a command input may exist. This implies that the command input $\underline{v}'$ modifies the DM's internal decision after $p(v|\bar{z})$ has been determined.

In the expressions defining the system coordination, p_i is the probability that algorithm f_i has been selected for processing the input x and p_j is the probability that algorithm h_j has been selected, i.e., u=i and $\bar{v}$=j. The quantities g_c represent the internal coordination of the corresponding algorithms and depend on the distribution of their respective inputs. The quantity $\mathscr{H}$ is the entropy of a random variable that can take one of two values with probability p:

$$\mathscr{H}(p) = -p\log p - (1-p)\log(1-p) \tag{14}$$

If there is no switching, i.e., if for example p(u=i)=1 for some i, then $\mathscr{H}$ will be identically zero for all p_i and the only non-zero term in the first sum will be

$$g_c^i(p(x))$$

Similarly, the only non-zero term in the second sum will be

$$g_c^{U+j}(p(\bar{z}|\bar{v}=j))$$

The quantity G may be interpreted as the total information processing activity of system S and, therefore, it can serve as a measure of the workload of the organization member in carrying out his decisionmaking task.

TEAMS OF DECISIONMAKERS

In order to define an organizational structure, it is necessary to specify exactly the interactions of each decisionmaker with the organization. A decisionmaker is said to interact with the environment when he receives inputs directly from sources or when he produces outputs that are all or part of the organization's output. The internal interactions consist of receiving inputs from other DMs, sharing situation assessments, receiving command inputs, and producing outputs that are either inputs or commands to other DMs. If these interactions are shown graphically in the form or a directed graph, then the organizational forms being considered have directed graphs which do not contain any cycles or loops. The resulting decisionmaking organizations are defined as having acyclical information structures. This restriction in the structure of the organizations is introduced to avoid deadlock and also messages circulating within the organization. It is also consistent with the assumptions inherent in the use of this particular information theoretic framework based on entropy rather than entropy rates. The organizations being considered then are multiechelon systems (Mesarovic et al, 1970) where the composition of the echelons and their ordering follow the path defined by the acyclical information structure (see Fig. 1) from inputs to the organization to the organization's output.

The typical decisionmaker in such an organization may receive inputs directly from the environment and will process them to obtain his situation assessment. He will transmit his assessment to some other DMs belonging to the same or "downstream" echelons, but not "upstream" echelons. He may also receive situation assessments from decisionmakers in the same echelon or from upstream ones; he will combine this information to produce his modified situation assessment. Note that the interaction allowed with DMs in the same echelon is of the result-sharing type; the exchange is simultaneous and consists of the original assessments, not the ones modified by the shared situation assessment information. He then proceeds to select a response. He may receive command inputs from DMs in upstream or the same echelon that can modify his response selection. He then produces outputs that are transmitted to downstream DMs or to the environment. He can also produce commands for DMs in the same or downstream echelons. The dominant constraint is that he cannot send commands to DMs from which he is also receiving commands. Such a situation will create loops and result in deadlock. Note that this restriction was not necessary in the situation assessment stage when the simultaneous sharing of information was allowed.

The types of information-processing and decisionmaking organizations that can be modeled and analyzed are exemplified by the two three-person organizations A and B shown in Figures 4 and 5, respectively. Three-person organizations were chosen because they require relatively simple notation. The approach applies to n-person organizations, however. Let the three decisionmakers be denoted by DM^1, DM^2, and DM^3. Their corresponding variables are superscripted 1, 2, and 3, respectively. The notation z^{12} indicates that variable z is generated by DM^1 and is received by DM^2.

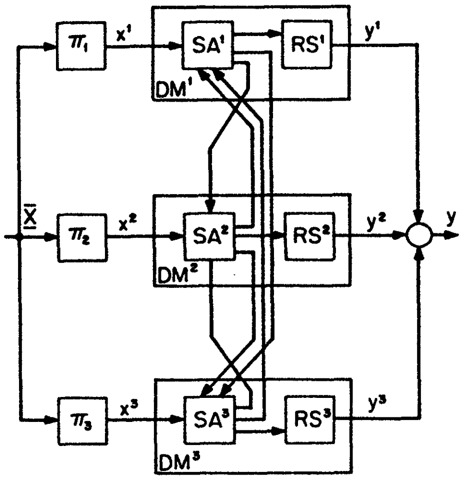


Fig. 4 Three person organization
A: Parallel Structure

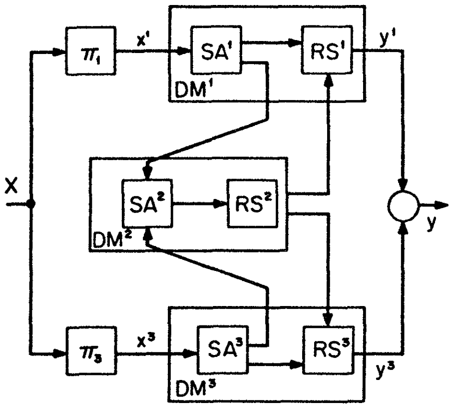


Fig. 5 Three person organization
B: Hierarchical Structure

In the first case, A, all three decisionmakers receive signals from the environment, process them to assess the situation as perceived by each one and then share their situation assessments. Each revises his assessment and proceeds to select a response. There are no command inputs; the organizational output is the combined outputs of the three DMs. This is a pure parallel structure: the task has been divided into three subtasks done in parallel. However, there are lateral links - the sharing of situation assessment information - between the three DMs that constitute a single echelon.

The second organizational structure, Fig. 5, is more complex. The task is divided into two subtasks. The first and third DMs receive the external inputs and assess the situation. They transmit the assessments to the second DM who processes them and generates commands that he then transmits to the other two DMs. These commands restrict the options in selecting responses by DM^1 and DM^3. The two produce the outputs which constitute the organization's output. The second decisionmaker has, clearly, a supervisory role, even though he is in the same echelon.

The four quantities that characterize the total information-processing and decisionmaking activity G of each DM in organizations A and B are obtained directly by specializing equations (6), (7), (12) and (13). In organization A, all decisionmakers have an identical structure although the specific algorithms f_i and h_j in the SA and RS stages, respectively, may differ. The expressions are presented for DM^1; the expressions for DM^2 and DM^3 are identical in form but with the appropriate superscripts.

Organization A: Decisionmaker 1 (or 2, or 3)

$$G_t^1 = T(x^1, z'^{21}, z'^{31} : y^1) \qquad (15)$$

$$G_b^1 = H(x^1, z'^{21}, z'^{31}) - G_t^1 \tag{16}$$

$$G_n^1 = H(u^1) + H_{\bar{z}'^1}(v^1) \tag{17}$$

$$\begin{aligned} G_c^1 = & \sum_{i=1}^{U^1} [p_i^1 g_c^{1i}(p(x^1)) + \alpha_i^1 \mathscr{H}(p_i^1)] \\ & + H(z^1) + H(z'^{12}) + H(z'^{13}) \\ & + g_c^{1A}\,(p(z^1, z'^{21}, z'^{31})) \\ & + g_c^{1B}\,(p(\bar{z}^1)) \\ & + \sum_{j=1}^{V^1} [p_j^1 g_c^{1j}(p(\bar{z}^1 | \bar{v}^1 = j)) + \alpha_j'^1 \mathscr{H}(p_j^1)] \\ & + H(y^1) + H(z^1) + H(\bar{z}^1) + H(\bar{z}^1, \bar{v}^1) \\ & + T_{z^1}(x^1 : z'^{21}, z'^{31}) \end{aligned} \tag{18}$$

In organization B, decisionmakers DM^1 and DM^3 serve identical roles and, therefore, the expressions for the four terms are similar. Only those for DM^1 are presented; those for DM^3 are obtained by substituting the apropriate superscripts. The second decisionmaker acts as a coordinator and supervisor, and does not receive inputs directly from the environment. This is reflected in the expression for coordination.

<u>Organization B: Decisionmaker 1</u> (or 3)

$$G_t^1 = T(x^1, v'^{21} : z'^{12}, y^1) \tag{19}$$

$$G_b^1 = H(x^1, v'^{21}) - G_t^1 \tag{20}$$

$$G_n^1 = H(u^1) + H_{\bar{z}^1}(v^1) \tag{21}$$

$$\begin{aligned} G_c^1 = & \sum_{i=1}^{U^1} [p_i^1 g_c^{1i}(p(x^1)) + \alpha_i^1 \mathscr{H}(p_i^1)] \\ & + H(\bar{z}^1) + H(z'^{12}) + g_c^{1B}(p(\bar{z}^1, v'^{21})) \\ & + \sum_{j=1}^{V^1} [p_j^1 g_c^{1j}(p(\bar{z}^1 | \bar{v}^1 = j)) + \alpha_j'^1 \mathscr{H}(p_j^1)] \\ & + H(y^1) + H(\bar{z}^1) + T_{\bar{z}^1}(z'^{12} : v'^{21}) \end{aligned} \tag{22}$$

<u>Organization B: Decisionmaker 2</u>

$$G_t^2 = T(z'^{12}, z'^{32} : y^{21}, y^{23}) \tag{23}$$

$$G_b^2 = H(z'^{12}, z'^{32}) - G_t^2 \tag{24}$$

$$G_n^2 = H(u^2) + H_{\bar{z}^2}(\bar{v}^2) \tag{25}$$

$$\begin{aligned} G_c^2 = & \, g_c^{2A}(p(z'^{12}, z'^{32})) + g_c^{2B}(p(\bar{z}^2)) \\ & + \sum_{j=1}^{V^2} [p_j^2 g_c^{2j}(p(\bar{z}^2 | \bar{v}^2 = j)) + \alpha_j'^2 \mathscr{H}(p_j^2)] \\ & + H(v'^{21}) + H(v'^{23}) + H(\bar{z}^2) + H(\bar{z}^2, \bar{v}^2) \end{aligned} \tag{26}$$

It follows from expressions (15) to (26) that the interactions affect the total activity G of each DM. At the same time, these interactions model the control that is exerted by the DMs on each other. These controls are exerted either directly through the command inputs v' or indirectly through the shared situation assessment z'.

All decisionmakers in Fig. 4 are subject to indirect control. The supplementary situation assessments z' modify the assessments z to produce the final assessment $\bar{z}$. Since $\bar{z}$ affects the choice of output, it follows that each DM is influenced by the assessments of the other DMs.

Direct control is exerted in organization B, Fig. 5, through the command inputs from DM^2 to the other two members. The variables v' modify the response selection strategies $p(v|\bar{z})$ of DM^1 and DM^3. Note that both types of controls, direct (v') and indirect (z'), can improve the performance of a decisionmaker, but can also degrade it.

The value of the total processing activity, G, of each decisionmaker depends on the choice of the internal decision strategies adopted by him, but also on those of the other members of the organization with whom he interacts directly or indirectly.

Let an internal decision strategy for a given decisionmaker be defined as pure, if both the situation assessment strategy p(u) and the response selection strategy $p(v|\bar{z})$ are pure, i.e., an algorithm f_r is selected with probability one and an algorithm h_s is selected also with probability one when the situation is assessed as being $\hat{z}$:

$$D_k = \{p(u=r) = 1; \;\; p(v=s|\bar{z}=\hat{z}) = 1\} \tag{27}$$

for some r, some s, and for each $\hat{z}$ element of the alphabet $\bar{Z}$. There are n possible pure internal strategies,

$$n = U \cdot V^M \tag{28}$$

where U is the number of f_i algorithms in the SA stage, V the number of h_j algorithms in the RS stage and M the dimension of the set $\bar{Z}$. All other internal strategies are mixed (Owen, 1968) and are obtained as convex combinations of pure strategies:

$$D(p_k) = \sum_{k=1}^{n} p_k D_k \tag{29}$$

where the weighting coefficients are probabilities.

A triplet of pure strategies, one for each DM, defines a pure strategy for the organization:

$$\Delta_{k,\ell,m} = \{D^1_k, D^2_\ell, D^3_m\} \qquad (30)$$

Independent internal decision strategies for each DM, whether pure or mixed, induce a behavioral strategy (Owen, 1968) for the organization

$$\Delta = \{D^1(p_k), D^2(p_\ell), D^3(p_m)\} \qquad (31)$$

Given such a behavioral strategy, it is then possible to compute the total processing activity G for each DM:

$$G^1 = G^1(\Delta) \ ; \ G^2 = G^2(\Delta) \ ; \ G^3 = G^3(\Delta) \qquad (32)$$

This interpretation of the expressions for the total activity is particularly useful in modeling the bounded rationality constraint for each decisionmaker and in analyzing the organization's performance in the performance-workload space.

BOUNDED RATIONALITY AND PERFORMANCE EVALUATION

The qualitative notion that the rationality of a human decisionmaker is not perfect, but is bounded, has been modeled as a constraint on the total activity G:

$$G^i = G^i_t + G^i_b + G^i_n + G^i_c \leq F^i \tau \qquad (33)$$

where τ is the mean symbol interarrival time and F the maximum rate of information processing that characterizes decisionmaker i. This constraint implies that the decisionmaker must process his inputs at a rate that is at least equal to the rate with which they arrive. For a detailed discussion of this particular model of bounded rationality see Boettcher and Levis (1982).

As stated earlier, the task of the organization has been modeled as receiving inputs X' and producing outputs y. Now, let, Y be the desired response to the input X' and let L(X') be a function or a table that associates a Y with each member of the input X'.

The organization's actual response y can be compared to the desired response Y using a function d(y,Y) which assigns a cost to each possible pair (y,Y). The expected value of the cost can be obtained by averaging over all possible inputs. This value, computed as a function of the organization's decision strategy Δ, can serve as a performance index J. For example, if the function d(y,Y) takes the value of zero when the actual response matches the desired response and the value of unity otherwise, then

$$J(\Delta) = E\{d(y,Y)\} = p(y \neq Y) \qquad (34)$$

which represents the probability of the organization making the wrong decision in response to inputs x; i.e., the probability of error. The procedure for evaluating the performance of an organization is shown in Fig. 6.

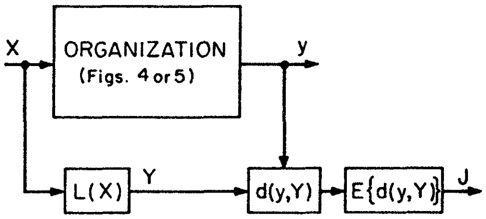


Fig. 6 Performance evaluation of an organization.

The information obtained from evaluating the performance of a specific organizational structure and the associated decision strategies can be used by the designer in defining and allocating tasks (selecting the partitioning matrices π_i), in changing the number and contents of the situation assessment and response selection algorithms and in redesigning the interaction between the DMs.

In order to do this, the designer can formulate and solve two problems: (a) the determination of the strategies that minimize J and (b) the determination of the set of strategies for which $J \leq \bar{J}$.

The first is an optimization problem while the latter is formulated so as to obtain satisficing strategies with respect to a performance threshold $\bar{J}$. Since the bounded rationality constraint for all DMs depends on τ, the internal decision strategies of each DM will also depend on the tempo of operations. The unconstrained case can be thought of as the limiting case when $\tau \to \infty$.

The solutions of the optimization and satisficing problems can be depicted graphically in the N+1 dimensional performance-workload space $(J, G^1, G^2, \ldots G^N)$. The locus of the admissible (N+1)-tuples is determined by analyzing the functional dependence of the organizational performance J and the total activity G^i of each decisionmaker i on the organization's strategy Δ.

For organization A and B the performance workload space is four dimensional, namely (J, G^1, G^2, G^3). The G^i of each decisionmaker is a convex function of the Δ, eq. (31), in the sense that

$$G^i(\Delta) \geq \sum_{k,\ell,m} G^i(\Delta_{k,\ell,m}) p_k\, p_\ell\, p_m \qquad (35)$$

where $\Delta_{k,\ell,m}$ is defined in eq. (30). Note that an alternate representation of Δ can be obtained from eqs. (30) and (31):

$$\Delta = \sum_{k,\ell,m} \Delta_{k,\ell,m}\, p_k\, p_\ell\, p_m \qquad (36)$$

The result in eq. (35) follows from the definition of G^i as the sum of the marginal entropies of each system variable, eq. (11), and the fact that the possible distributions $p(w)$, where w is any system variable, are elements of a convex distribution space determined by the organization decision strategies, i.e.,

$$p(w) \in \{p(w) \mid p(w) = \sum_{k,\ell,m} p(w \mid \Delta_{k,\ell,m}) p_k p_\ell p_m\} \qquad (37)$$

The performance index of the organization can also be obtained as a function of Δ. Corresponding to each $\Delta_{k,\ell,m}$ is a value $J_{k,\ell,m}$ of the performance index. Since any organization strategy being considered is a weighted sum of pure strategies, eq. (36), the organization's performance can be expressed as

$$J(\Delta) = \sum_{k,\ell,m} J_{k,\ell,m}\, p_k\, p_\ell\, p_m \qquad (38)$$

Equations (35) and (38) are parametric in the probabilities p_k, p_ℓ and p_m. The locus of all admissible (J,G^1,G^2,G^3) quadruplets can be obtained by constructing first all binary variations between pure strategies; each binary variation defines a line in the four dimensional space (J,G^1,G^2,G^3). Then successive binary combinations of mixed strategies are considered until all possible strategies are accounted for. The resulting locus can be projected on the two-dimensional spaces (J,G^i) as shown in Boettcher and Levis (1982) in order to analyze the performance of a single decisionmaker. For organizations A and B, projection of the locus on the three dimensional space (J,G^1,G^2) is practical and convenient because in both cases the properties of DM^3 are analogous to those of DM^1.

The bounded rationality constraints, eq. (33), can be realized in the performance-workload space by constructing planes of constant G for each DM. For example, the constraint for DM^i is defined by a plane that is normal to the G^i axis and intersects it at $G^i_r = F^i\tau$. For fixed values of F^i, the bounded rationality constraint is proportional to the tempo of operations. As the tempo becomes faster, i.e., the interarrival time τ becomes shorter, the G^i_r becomes smaller and, consequently, a smaller part of the locus satisfies the constraint.

The solutions of the satisficing problem can be characterized as the subset of feasible solutions for which the performance measure $J(\Delta)$ is less than or equal to a threshold value $\bar{J}$. This condition also defines a plane in the performance-workload space that is normal to the J axis and intersects it at $\bar{J}$. All points on the locus on or below this plane which also satisfy the bounded rationality constraint for each decisionmaker in the organization are satisficing solutions.

The method of analysis presented thus far is illustrated in the next section through a simple example in which the two organizational forms, A and B, are compared.

EXAMPLE

A simple example has been constructed based on aspects of the problem of organizing batteries of surface to air missiles. Let a trajectory of a target be defined by an ordered pair of points located in a rectangle that represents a two-dimensional (flat) sector of airspace. From the ordered pair, the speed and direction of flight of the target can be determined. On the basis of that information, the organization should respond by firing either a slow or a fast surface-to-air missile or by not firing at all. The size of the sector and the frequency of the arrival of targets is such that three units are needed.

The first organizational structure, corresponding to Organization A, is defined as follows. The rectangular sector is divided into three equal subsectors and a decisionmaker is assigned to each one. Each DM is capable of observing only the points that appear in his subsector. He can assess the situation, i.e., estimate the trajectory, and select the response, i.e., which weapons to fire, for targets with trajectories totally within his subsector. This is the case when both points that define the target are within his subsector. Since it is possible for trajectories to "stradle" the subsector boundaries, it is necessary that situation assessment information be shared. Thus, DM^1 and DM^2 share information that relates to their common boundary. Similarly, DM^2 and DM^3 share information that relates to targets that cross their common boundary. To keep the computational effort small and the resulting loci uncomplicated, the situation assessment stages of DM^1 and DM^3 are assumed to contain a single algorithm f; that of DM^2 contains two algorithms, f^2_1 and f^2_2. In contrast, the response selection stage of DM^2 contains a single algorithm h, while the RS stages of DM^1 and DM^3 contain two algorithms h^i_1 and h^i_2, i=1,3. Therefore, the internal decision strategies are $p(u^2)$, $p(v^1 \mid \bar{z}^1)$ and $p(v^3 \mid \bar{z}^3)$. The detailed structure of this organization is shown in Figure 7.

The second organizational structure, corresponding to Organization B, is defined as follows. The rectangular sector is divided

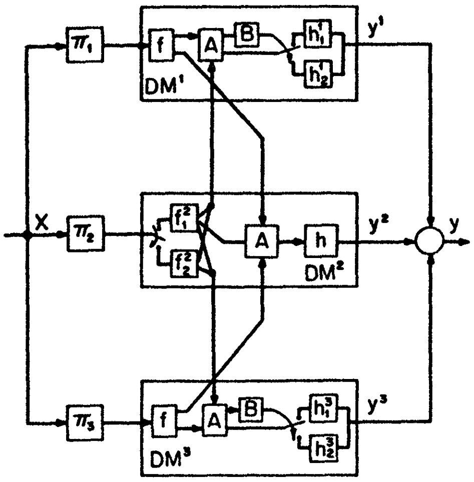


Fig. 7 Organization A in example

into two equal subsectors for which DM^1 and DM^3 are responsible for assessing the situation and selecting a response. The two DMs do not share situation assessment between themselves; however, data from the area adjacent to the boundary between DM^1 and DM^3 is transmitted to the coordinator or supervisor, DM^2, who resolves conflicts and assigns targets either to DM^1 or to DM^3, as appropriate. This is accomplished through command inputs v'^{21} and v'^{31} from the coordinator to the two commanders. They, in turn, exercise their response selection stage to determine response y^1 and y^3, respectively. Again, for computational simplicity, it is assumed that DM^1 and DM^3 have a single algorithm f for their SA stage and two algorithms h_1^i and h_2^i for the RS stage. The coordinator, DM^2, has an algorithm A for processing the assessed situations z'^{12} and z'^{32} and two algorithms, h_1^2 and h_2^2 , in the RS stage. The internal decision strategies are $p(v^1|\bar{z}^1)$, $p(v^2|\bar{z}^2)$ and $p(v^3|\bar{z}^3)$. The structure of this organization is shown in Figure 8.

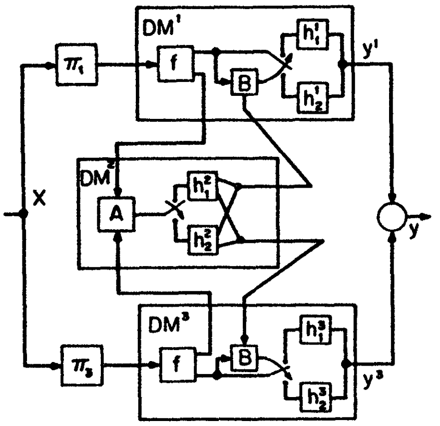


Fig. 8 Organization B in example

In order to compute the performance J of each organization and total activity G^i of each DM^i, it is necessary to specify the probability distribution of the targets, all the algorithms f and h, the algorithms A and B and a table of correct responses for each possible target. Then, each admissible pure strategy of the organization is identified. The construction technique described in the previous section is used to obtain the locus of all the feasible (J,G^1,G^2,G^3) quadruplets.

Consider first the performance-workload locus for each DM in each one of the two alternative organizational structures. The three loci for each organization are obtained by projecting the (J,G^1, G^2, G^3) locus on each on the three (J,G^i) planes respectively. The results for organization A of the example are shown in Figures 9-a,b,c; those of Organization B in Figures 10-a,b,c. The index of performance J measures the probability of error and is expressed in percentage. The total activity G^i is measured in bits per symbol. The two sets have been drawn at the same scale to allow for direct comparisons.

In organization A, the probability that an incorrect response (error) will be made in processing an input ranges from 3.5 percent to 4.6 percent. Decisionmakers DM^1 and DM^3 have very similar, but not identical loci. The difference in the loci is due to asymmetries in the input, i.e., $H(x^1) \neq H(x^3)$. Note, however, that their total activity G ranges between 22 and 35 bits per symbol.

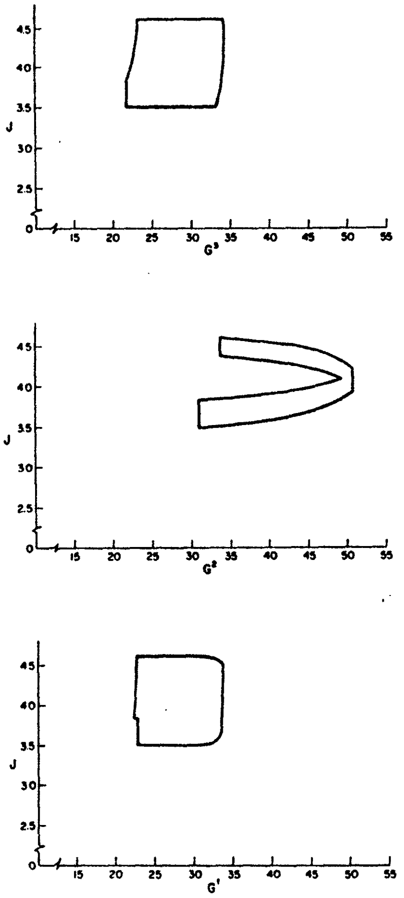
The performance-workload locus of DM^2, however, is quite different: the G ranges from 31 to 51 bits and, for a fixed G, there are, in general, two ranges of possible values of J.

The loci of all three DMs exhibit the properties discussed in Boettcher and Levis (1982). The optimal (minimum error) performance is achieved with a pure strategy when there are no bounded rationality constraints. The existence of such a constraint would be shown by a line of constant G^i with all feasible loci points to the left (lower G) of the line. If, for example, the constraint was the same for all three DMs, namely,

$$G^i \leq G_r = 40 \quad \text{bits/symbol}$$

then none of the admissible organization strategies would overload DM^1 and DM^3; however, DM^2 would be overloaded for some of the strategies. Therefore, only the organization strategies that do not overload any one of the organization's members are considered feasible.

Comparison for the three loci for the decisionmakers in Organization B indicates that their loci are very similar: the organization's probability of error ranges between 2.4 and 4.0 percent. The total



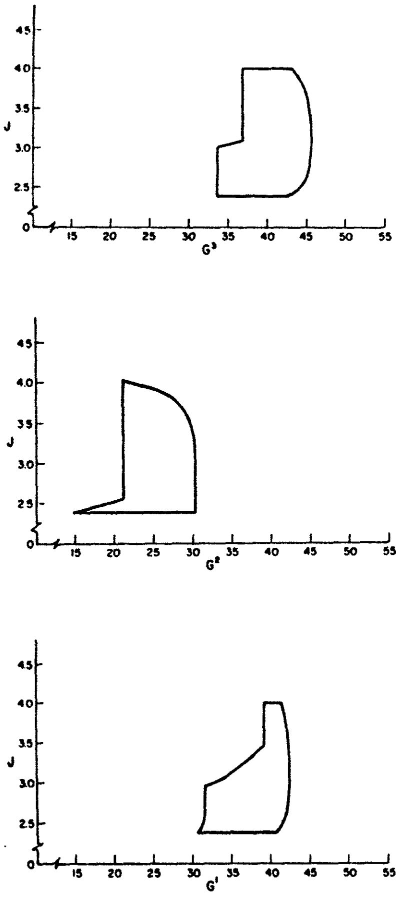


Fig. 9 Performance-Workload projection for DM^3, DM^2, and DM^1, respectively, in Organization A,

Fig. 10 Performance-Workload projection for DM^3, DM^2, and DM^1, respectively, in Organization B.

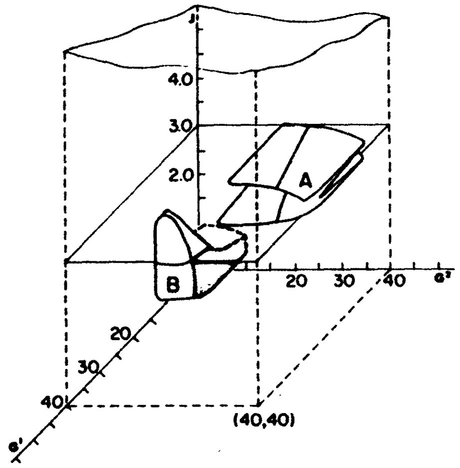


Fig. 11 Organizational Performance versus Individual Workload. Projection of four-dimensional (J,G^1,G^2,G^3) loci in three-dimensional space (J,G^1,G^2) for Organizations A and B.

activity level for DM^1 and DM^3 is between 30 and 45 bits/symbol. Again, the differences in the two loci are due to asymmetries in the tasks (inputs) assigned to each DM. The coordinator, DM , has a much lower workload: his total activity ranges between 15 and 30 bits per symbol. This is a consequence of not having to process either external inputs (no algorithms f) or command inputs (no algorithm B). In this case, if the bounded rationality constraints are set at G_r=40, they will restrict the choice of strategies by DM^1 and DM^3 and hence the organization's strategies.

If the two sets of loci are compared with each other, it becomes apparent that Organization B has the ability to perform better, i.e., make fewer errors, then Organization A. In the absence of bounded rationality constraints, B would be the preferred design. This would be especially true, if there were a satisficing constraint that required the organization's performance to be much that the error probability be less than a given value, such as three percent.

These results could be seen best by considering the comparison of the two (J,G^1,G^2,G^3) loci and the associated bounded rationality constraints. Since the performance-workload characteristics of DM^1 and DM^3 are essentially the same in each organization, the four-dimensional locus was projected in the (J,G^1,G^3) space. The two loci, L^A and L^B, are shown in Figure 11. The satisficing condition, $J \le \bar{J}$ is shown as a plane parallel to the (G^1,G^2) plane intersecting the J axis at 3.0. The bounded rationality constraints for DM^1 and DM^2 are planes parallel to the (J,G^2) and the (J,G^1) plane at 40 bits/symbol.

It is clear from the figure that the choice of preferred organizational structure to carry out the assigned task depends on the values of the bounded rationality constraints and the satisficing threshold $\bar{J}$. If the satisficing constraint is $\bar{J}$=3.0, then the design represented by Organization A is not an effective one: the organization cannot perform the task. However, there are many strategies that the decisionmakers in Organization B can use to carry out the task without overload.

The evaluation of the two designs has been carried out in a qualitative manner using the geometric relationships between the various loci in the performance-workload space. A quantitative approach to the evaluation and comparison of alternative designs is the subject of current research.

CONCLUSIONS

An analytical methodology for modeling and analyzing structures of information-processing and decisionmaking organizations has been presented. The approach was applied to the design of three person organizations assigned to execute a well defined task. Implicit in the design of the organizational form in the C^3 system required to support the information processing and decisionmaking activity.

ACKNOWLEDGMENT

The authors wish to thank Gloria Chyen and Vincent Bouthonnier for their help in developing the example.

REFERENCES

Boettcher, K. L., and A. H. Levis (1982). Modeling the interacting decisionmaker with bounded rationality. IEEE Trans. Sys.,Man & Cybern. (USA), SMC-12, May/June 1982.

Drenick, R.F. (1976). Organization and Control in Y.C. Ho and S.K. Mitter (Eds.) Directions in Large Scale Systems, Plenum Press, N.Y.

Froyd, J., and F. N. Bailey (1980). Performance of capacity constrained decisionmakers. Proc. 19th IEEE Conf. Decision & Control Albuquerque, N.M.

Levis, A.H. and K.L. Boettcher (1982). On Modeling teams of interacting decisionmakers with bounded rationality. Proc. IFAC/IFIP/IFORS/IEA Conf. on Analysis Design and Evaluation of Man Machine Systems, Pergamon Press, London, September 1982.

March, J.G, (1978). Bounded rationality, ambiquity, and the engineering of choice. Bell J. Ecomc. (USA) Vol. 9, pp, 587-608.

Mesarovic, M.D., D. Macko and Y. Takahara (1970). Theory of hierarchical, multilevel systems.Academic Press, N.Y.

Owen, G. (1968). Game Theory. W.B. Saunders Co. Philadelphia, PA.

Shannon, C.E. and W. Weaver (1949). The Mathematical Theory of Communication. The Uni. of Illinois Press, Urbana, IL.

Sheridan, T.B. and W.R. Ferrell (1974). Man-Machine Systems. The MIT Press,Cambridge MA.

Stabile, D.A., A.H. Levis and S.A. Hall (1982) Information Structures for Single Echelon Organizations. LIDS-P-1980, Laboratory for Information and Decision Systems, M.I.T.

Lawson, J.S., (1981). The Role of Time in a Command Control System. Proc. Fourth MIT/ONR Workshop on C^3 Systems, LIDS-R-1159, MIT, Cambridge, MA.

Conant, R.C. (1976). Laws of Information which Govern Systems. IEEE Trans. on Sys., Man & Cybernetics, Vol. SMC6, pp. 240-255.

A MATHEMATICAL FRAMEWORK FOR THE STUDY OF BATTLE GROUP POSITION DECISIONS

Dr. D. A. Castanon Dr. J. R. Delaney Dr. L. C. Kramer Prof. M. Athans

ALPHATECH, Inc.
3 New England Executive Park
Burlington, Mass. 01803

Dept. of Elect. Eng.
MIT
Cambridge, Mass.

1. INTRODUCTION

One of the principal decisions to be made by the command structure of a US Navy Battle group concerns the proper positioning of their surface platforms to improve the overall defensive effectiveness against expected enemy threats. The threats to Battle Groups (BG) are typically composite threats, with air, surface and subsurface components capable of attacking the BG from different approach sectors. Furthermore, each surface platform in the BG carries assets which can contribute defensive capabilities against all types of threats. Hence, the overall decision problem of ship positioning to enhance overall defensive effectiveness is a difficult coordination problem, which involves evaluating the contributions of all ships against the composite threat.

In this paper, we develop a mathematical framework which evaluates the decisions of ship positioning in terms of overall defensive effectiveness. This model is based on a probabilistic description of the expected enemy threats, and a probabilistic description of the capabilities of each ship against each kind of threat. The expected enemy threats are grouped into three classes: Surface, Subsurface and Air threats. For the sake of simplicity, we restrict ourselves to air and subsurface threats only. Based on this mathematical framework, we define an optimization problem to find the optimal ship positions for the expected enemy threat.

For this optimization problem, we develop a numerical algorithm for finding the optimal ship positions. This algorithm is an extension of a nonlinear programming algorithm developed by Bertsekas [1]. We apply the algorithm to study the optimal ship positions in a simple scenario. This illustrates the capability of the mathematical formulation to provide suggestions for ship-positioning decisions.

The rest of this paper is organized as follows: In section 2, we develop the mathematical framework which relates ship positions, perceived enemy threats and defensive effectiveness. In section 3, we describe the optimization problem to be solved in ship-positioning. Section 4 contains a brief description of the numerical algorithm, and the results for a simple scenario. Section 5 is a brief conclusion.

2. A MATHEMATICAL FRAMEWORK FOR DEFENSIVE EFFECTIVENESS

Our mathematical framework for defensive effectiveness has two principal interacting components: the perceived enemy threat, and the positions of all the friendly ships. The perceived enemy threat is described by a probabilistic model which assigns a probability distribution to the space of possible <u>attack paths</u>. We will describe this probability distribution for air and subsurface threats and the interactions of these attack paths with the friendly ships.

Consider the scenario depicted in Fig. 1. The x's in the figure indicate the position of friendly ships. The origin is defined for reference as the position of a given Carrier (CV) in the BG. We assume that, initially, all threats must originate outside of a volume V around the BG. This assumption is easy to relax in our framework. To keep geometry simple, we will neglect altitude, projecting all trajectories into a two-dimensional surface.

Let C(V) denote the space of all continuous trajectories on the surface V which start at the edge of V, over the time interval [0,T]. Let N_a be a finite index set, ennumerating the possible air threats which are within reach of the BG. A perceived <u>air threat</u> E_a is a pair (P_a, P_{na}) where

$P_a: N_a \dashrightarrow [0,1]$ is the probability that na is a threat.

P_{na} is a probability distribution on the Borel subsets of C(V) for each na.

Our definition of E_a includes intelligence information on what the threat is, and what are its likely attack patterns. A generalization of the definition will permit the air threat to represent branching trajectories, such as a missile separating from an airplane. This extension is similar to the representation used in the subsurface threat model discussed next.

Let N_s be a finite index set, ennumerating the possible subsurface threats which are within reach of the BG. A perceived <u>subsurface threat</u> E_s is a quintuple

$(P_s, P_{ns}, P(\ ;ns,x), P_{ns}(\ ;x), P_{na}(\ ;x)$

where

$P_s : N_s \longrightarrow [0,1]$ is the probability that ns will be a threat,

P_{ns} is a probability distribution on the Borel subsets of C(V) for each ns in N_s,

$P(;ns,x) : N_a \cup N_s$ --- [0,1] is the probability that a trajectory for ns which reaches x will give birth to air threat na or subsurface threat ns at x,

$P_{ns}(;x)$ is a probability distribution on the Borel subsets of C(v) starting at x, describing posible trajectories for ns.

$P_{na}(;x)$ is a probability distribution on the Borel subsets of C(v) starting at x, describing possible trajectories for na.

Note particularly that subsurface threats can become air threats after part of their trajectory has been completed. We can illustrate our subsurface threat model with the following example:

Let w denote a sample path for submarine 1 penetrating to pint x in V, where this submarine will surface and launch a cruise missile of type 2. This missile will follow an attack trajectory v as a sample path. The probability that w is the trajectory, followed by a cruise missile trajectory v is given by

$$P(1,w,2,v) = P_1(w)\ P(2;1,x)P_2(v;x)$$

where the first term is the probability that w was the trajectory for submarine 1, the second term is the probability that submarine 1 fires cruise missile 2 at x, and the third term is the probability that cruise missile 2 flies trajectory v starting at x.

The probabilistic threat models E_a and E_s generate a distribution of trajectories for the objects in $N_a x N_s$. These distributions can depend in general on the location of the friendly force positions. In particular, these distributions should depend continuously on the position of the Battle Group ships. Denote this position by the vector $\underline{x}$ in R^{2m} where m is the total number of ships present. Then, all of the elements of E_a and E_s are assumed to be continuously differentiable in $\underline{x}$.

The threat models E_a and E_s describe the threat trajectories assuming that no friendly ship takes any action against them. In any model of defensive effectiveness, one needs to model the potential outcomes of any actions which the ships may take against the incoming threats. As before, we use probabilistic modeling.

For each ship position, we describe its capability for detecting a subsurface threat and for prosecuting successfully an air threat or a detected subsurface threat. Let N_f be an index set denoting the m friendly ships. The subsurface detection model is based on classical models in surveillance theory [3], where a ship at position x contributes a search effort to detecting a subsurface target at position y. In general, this search effort is represented in functional form, as g(x,y;ns,nf), which is the probability that subsurface target ns at y is detected between times 0 and dt by ship nf at x, given that it was undetected by time 0. We assume that the overall search effort using several ships is independent among ships, and cumulative. Then, for any trajectory c(t), t in [0,T], of target ns, the probability of remaining undetected by time T is

$$P = \exp\left\{ \int_0^T \sum_i g(x_i,c(t);ns,i)dt \right\}$$

Typically, the search effort functions will depend only on the distance between ship and target. The only other restriction on g is that it should be continuously differentiable as a function of x, for each ns and nf.

The capability of each ship to prosecute successfully an air threat is described probabilistically as a function of the ship position and the trajectory of the air threat. The basic principle behind this function is that the probability that an air threat survives prosecution by a ship depends on the nature of the air threat, the nature of the ship, and the amount of prosecution time available. This time depends on the trajectory of the air threat, relative to the ship's position, as illustrated in figure 2. Mathematically,

$$\text{Prob}\ \{\text{Air threat na survives prosecution by ship i}\} = f_i(na,\ |t_1 - t_0|)$$

where the function $f_i : N_a x R^+ \longrightarrow [0,1]$ is a continuous, monotone-decreasing function of $|t_1 - t_0|$ and $f_i(na,0) = 1$.

When several ships prosecute a single air threat simultaneously, we assume that the events of surviving prosecution by each ship are mutually independent. Hence, the probability that an air threat na survives prosecution by all ships is given by

$$P = \prod_i f_i(na,\ |t_1(i) - t_0(i)|)$$

The final interaction which takes place between arriving enemy threats and a ship positioned at x is the prosecution of subsurface detections. In this model, we assume that a subsurface threat has been detected at position y; the probability of this event is computed from the subsurface surveillance model. The quantity which must be computed is the probability that this detected subsurface threat will reach its terminal state before it is prosecuted successfully. Otherwise, at this terminal state, the subsurface threat will give birth to other threats, air and subsurface, which must be prosecuted.

The subsurface attrition model is based on time available for prosecution. Let t_0

be the time that subsurface threat ns was detected at x. Let tf be the time at which ns reaches its destination xf. Denote by c(t), t in $[t_0,tf]$ the trajectory connecting x to xf. Assume that ship i can move to prosecute ns at a speed v. The subsurface attrition model is based on computing ti, the time of earliest intercept from ship i at trajectory c(t). Then, the probability of surviving prosecution is given by a function h_i, where

$$\text{Prob }\{\text{ns reaches xf under prosecution from ship i}\} = h_i(\text{ns},\text{tf} - \text{ti})$$

When several ships are prosecuting ns at the same time, the probability of successful escape from each ship is assumed independent of every other ship, so that

$$\text{Prob }\{\text{ns reaches xf under prosecution from ships 1 to n}\} = \prod_i h_i(\text{ns},\text{tf}-\text{ti})$$

The subsurface attrition model is illustrated in figure 3.

Based on the mathematical model described above, we can construct a quantitative measure of defensive effectiveness as follows: To each enemy threat n in $N_a \cup N_s$ we assign a nonzero value k(n,xf) to the event that the threat reaches its destination xf. For example, if the destination xf is a particular carrier, and n is a particular cruise missile, there is a value associated with the potential damage that such a missile will cause the overall battle group if it reaches the carrier. This value depends on the destination of the missile, as indicated by the functional dependence of k. The overall defensive effectiveness of a particular battle group position is given by

$$\text{D.E.} = E\{ \sum_n k(n,xf) \, \mathbf{I} \, \{\text{threat n reaches xf}\}\}$$

That is, defensive effectiveness is defined as the expected value of all the threats which reach their targets. This expectation refers to the average over the threat distribution and over the interactions between threats and BG ships.

3. PROPERTIES OF THE MODEL

The problem of ship positioning in the Battle Group can be described mathematically by the following optimization problem:

$$\text{Select } x_i,\ i = 1,N, \text{ to maximize D.E.}$$

This optimization problem is a nonlinear programming problem, where the objective function is the expected value of defensive effectiveness, given the threat distributions and the probabilistic interactions between threats and surface platforms. In this section, we highlight some of the mathematical features of the nonlinear programming problem.

First of all, it is important to note that numerical evaluation of the objective function for particular ship positions is a complex operation, requiring a probabilistic average over a space-time distribution of threat-surface platform interactions. Hence, any search algorithm which requires extensive function evaluations, such as those using second-order Hessian information, will be prohibitive in computational cost. Furthermore, the objective function will not be twice differentiable in most cases. This is a property of the parametrizations used in the threat-platform interaction models described in the previous section.

Consider the diagram in figure 4. A surface platform x is interacting with 3 air threats, whose trajectories are indicated in the figure. These trajectories are all aimed at a common target T. The change in the amount of time within weapon range of x for each threat varies in a smooth way as long as the target T is either strictly outside or strictly inside weapon range. However, when the target is on the boundary, the rate of change is discontinuous. This is due to the fact that, while the target was strictly outside, all of the threats had to cross the entire weapon range circle, whereas if the target is inside, the threats will only cross part of this circle. Hence the derivative of the objective function is discontinuous at these types of points. These discontinuities can be smoothed out if a more complex air threat prosecution model is adopted.

Rather than developing a smoother version of the air threat prosecution model, and the other discontinuities present in the subsurface detection and prosecution models, we developed an algorithm which can work with nondifferentiable cost functions. This algorithm is described in the next section, and it permits us to use the intuitive parametrizations which we used in our model.

4. AN ALGORITHM FOR NUMERICAL SHIP POSITIONS

The algorithm in this section deals with the nondifferentiability problems mentioned in the previous section by constraining the search away from the points or surfaces of nondifferentiability. This algorithm is based on a self-scaling quasi-Newton search described in [2], with a modification to incorporate simple constraints as described in [1]. A proof of convergence is included in [1]. We provide an outline of the algorithm below.

1. Select the initial ship positions x(0) away from the nondifferentiable regions. Let n be the dimension. Set k = 0.
2. Evaluate the gradient of the objective function, denoted g(k).
3. Initialize the matrix S(k) to the identity if k is an integer multiple of n.
4. Select search direction d(k) =-S(k)g(k).
5. Search for optimal step size along d(k). If the suggested position violates constraints, project this position to the admissible set. Denote this point x(k+1).
6. Evaluate the gradient g(k+1) at x(k+1)

7. Define $p(k)=x(k+1)-x(k)$, $q(k) = g(k+1) - g(k)$.

8. Check that $p(k) \cdot q(k)$ is positive. Otherwise, set $S(k+1) = S(k)$ and go to step 12.

9. Set $S(k+1)$ as

$$S(k+1) = \left\{ S(k) - \frac{S(k)q(k)q^t(k)S^t(k)}{q^t(k)S(k)q(k)} \right\} \cdot \frac{p^t(k)q(k)}{q^t(k)S(k)q(k)} + \frac{p(k)p^t(k)}{p^t(k)q(k)} .$$

10. If $x(k+1)$ is in the boundary of the constraint set, reduce $S(k+1)$ to the identity on directions normal to the constraint surface.

11. Change k to k+1 and return to 3.

The algorithm outlined in steps 1-12 is based on computing an approximation to the inverse of the Hessian of the objective function, based on gradient evaluations. Hence, its convergence rate is superlinear. The constraints which avoid the regions of non-differentiability are included in steps 5 and 10. In polar coordinates, these constraints can be stated simply in terms of the radial distance of the ship positions.

We programmed the above algorithm for the following attack scenario: Consider a BG consisting of one carrier and two surface ships, denoted S and A. Associated with each surface ship and the carrier, we defined the appropriate functions h_i, g_i and f_i. We assumed that these functions were equal for each submarine, or each air threat, in the scenario. As an indication of these functions, the weapon ranges of S and A against air threats were 10 and 15 miles, respectively. The subsurface detection ranges were 50 and 5 miles. The carrier air defense system was assumed to shoot down missiles with a probability of .4. Carrier and ship-based assets for prosecuting submarines were assumed to be 4 times faster than the submarines themselves.

The components of the threat scenario were as follows: On the sector from 30° to 75 ° based on the carrier, 40 cruise missiles were quadratically distributed. On the sector from 0° to 45°, five submarines would penetrate to a launch range of 150 miles, where they would surface and launch 6 cruise missiles each towards the carrier. All of the trajectories were assumed to be straight line, minimum time trajectories. The threat scenario is illustrated in fig. 5.

The numbers we chose for the scenario and the capabilities of the BG ships bear little resemblance to the actual capabilities of real ships. These numbers were selected casually to illustrate how the various ship parameters influence the optimal positions.

The algorithm of steps 1-11 was coded in 300 lines of Fortran code, and used to search for the optimal ship positions. This search required about 3 seconds of computation. The algorithm was started with a wide variety of initial conditions, and only two local minima were found. These are described below.

The first local minimum is illustrated in figure 6. Essentially, the ASW-capable ship S goes out and tries to detect and prosecute submarines before they reach their launch range. The other ship A sets a picket against air missiles. The optimal positions of the ships are: A is located 15 miles from the carrier at a bearing of 50°. S is located 163 miles from the carrier at a bearing of 20°. Note that the positions of S and A are not in the centers of their sectors. This is because of the implicit coordination which requires that A shoot at any missiles that are launched from submarines which escape S. In this position, only 1.5 of the possible 30 missiles carried by the submarines reach the carrier, whereas 14.2 of the 40 air-launched missiles reach the carrier.

The second local minimum is illustrated in figure 7. Both the ASW-capable ship S and ship A are placed close to the carrier. The submarines are allowed to launch their missiles unmolested, and the effort is placed on setting a more effective air-missile intercept near the carrier. The optimal positions for the ships are now S at 10 miles and bearing 39.6°, and A at 15 miles and bearing 39.6°. Under this double coverage, the expected number of submarine-launched missiles which reach the carrier increases to 5.6, but the number of air-launched missiles is reduced to 6.85. Hence, this configuration reduces the total number of missiles that reach the carrier.

5. CONCLUSION

The main result of this paper has been to develop a general mathematical framework for describing the problem of ship positioning as a normative optimization problem, and to demonstrate how numerical procedures can be designed which will provide optimal ship positions for uncertain, random descriptions of the enemy threats. In order to be of tactical use, the crude model described in this paper needs to be extended and refined. If such refinements are incorporated, the resulting mathematical model may be used for the analysis of suggested ship positions by commanders, or as the basis for an interactive decision aid. Such applications should be considered in future research.

ACKNOWLEDGEMENT

This work was supported by the Naval Electronic Systems Command under Contract N00039-81-C-0243. The authors are indebted to Dr. J. Lawson and Mr. J. Machado for their helpful suggestions.

REFERENCES

1. D. P. Bertsekas, "Projected Newton algorithms with simple constraints", SIAM J. Control, March, 1982.
2. D. G. Luenberger, Introduction to Linear and Nonlinear Programming, Addison-Weley, Reading, Mass. 1973.
3. L. D. Stone, Theory of Optimal Search, Academic Press, New York, 1975.

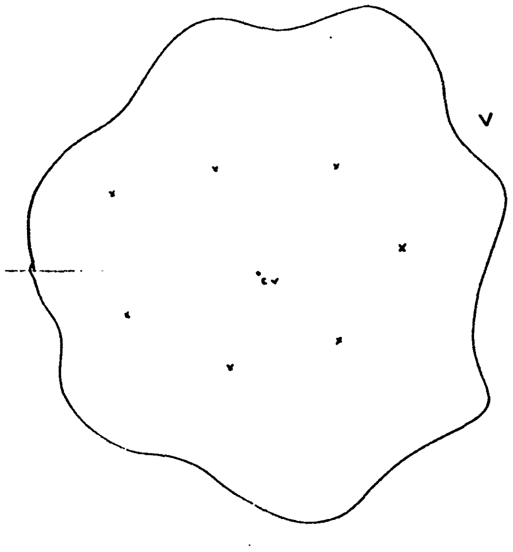


Figure 1.

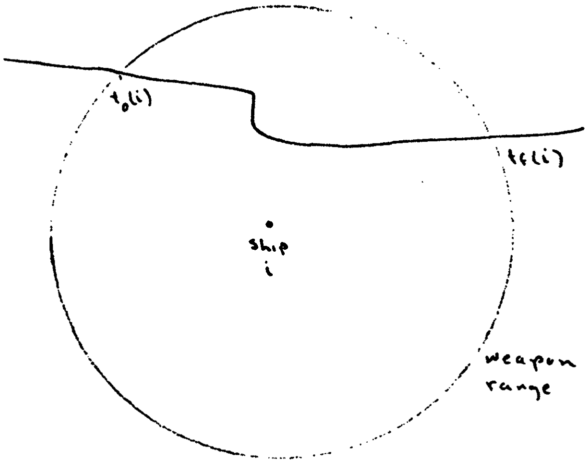


Figure 2

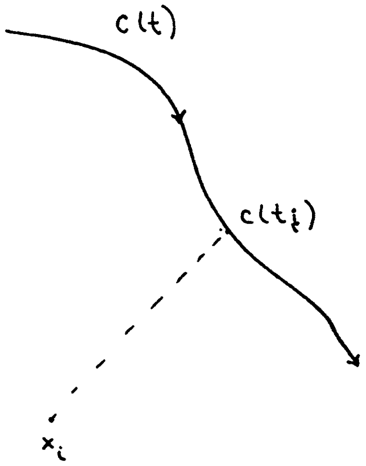


Figure 3.

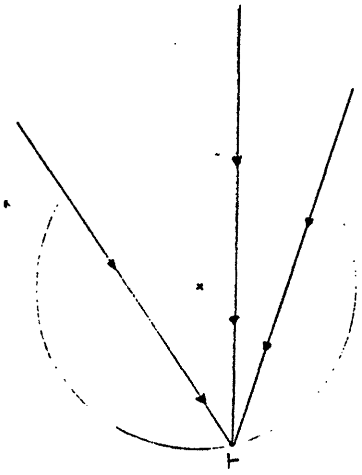


Figure 4.

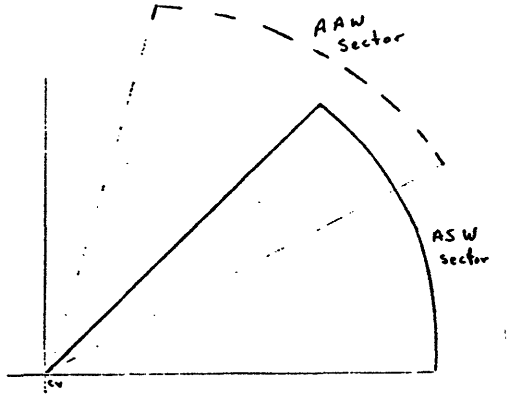


Figure 5

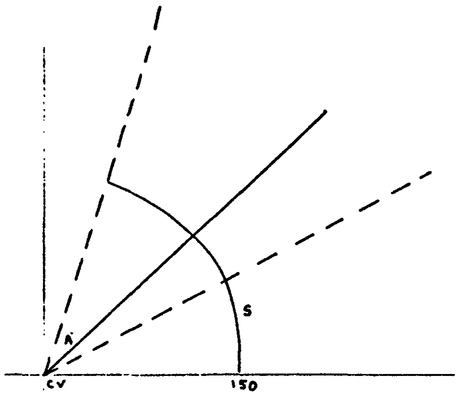


Figure 6.

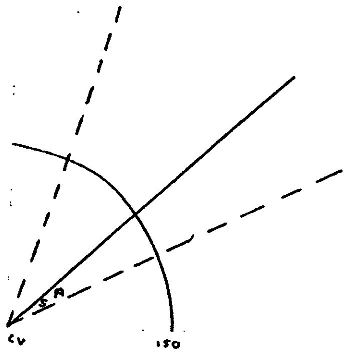


Figure 7.

DYNAMIC SEQUENCE ASSIGNMENT IN AIRCRAFT MISSION PLANNING

V.G. Rutenburg, R.P. Wishner, J.M. Abram[1]
Advanced Information & Decison Systems
Mountain View, CA 94040

Edison Tse[2]
Department of Engineering-Economic Systems
Stanford University, Stanford, CA 94305

Abstract. In this paper we consider the problem of aircraft mission planning. We discuss an interactive, analytical decision aid for this planning problem. We begin by describing analytical models of aircraft mission planning and the decision making structure involved. We then present some of the analytical algorithms that we have developed for the solution of this class of problems. These include a new form of dynamic programming (DP)that has several advantages over the traditional DP approach to this problem.

INTRODUCTION

Naval force engagements are sufficiently complex that commanders need analytical tools to better plan attacks that involve use of multiple resources against multiple targets. Our effort concerns itself with developing an automatic dynamic planning aid for assisting in the aircraft mission planning process.

Current mission planning techniques emphasize manual approaches and make heavy use of heuristic techniques. There now exist evaluation tools that can be used to evaluate the goodness of manually derived trajectories. Research has been done on calculating optimal trajectories for a single aircraft penetrating enemy defenses. No significant work that we are aware of has addressed the overall mission planning problem involving multiple targets, multiple aircraft with multiple weapons and the interactions and tradeoffs among the alternative utilization of these resources in achieving an optimal overall mission plan. Significant improvements in overall force effectiveness should result from the development of an analytically based dynamic mission planner.

The ultimate goal of our research effort is to develop such an interactive analytical planning aid for aircraft mission planning. This decision aid can assume the functions dealing with all the low-level procedural, computational and search tasks, enabling the decision makers to concentrate on the important high-level planning issues. Thus, this decision aid will provide for a successful symbiosis between the man and the computer, utilizing the best capabilities of each of them.

The overall control of the decision process will be in the hands of the human decision maker, who will be in charge of providing the overall strategic guidelines. He will use the decision aid for a) complex mathematical computation of the best plans within the given strategic guidelines, and b) evaluation and graphic simulation of strategic plans.

MATHEMATICAL FORMULATION

The current formulation is a high-level model of the surface-to-land battle situation. The forces involved in the battle are friendly aircraft carriers with airplanes and/or surface-to-land cruise missiles on board poised against enemy land targets and enemy defenses protecting those targets.

The objective of an attack mission is to inflict the maximum possible damage to the enemy targets, subject to survivability constraints. Planning of this mission involves jointly making the following decisions:

1) which high-valued targets to attack,

2) which defensive threats should be destroyed to facilitate the attack.

3) which attacking units to assign to each objective,

4) the sequential order of these attacks, and

[1] Research partially supported by ONR Contract No. N00014-82-C-0085.

[2] Research partially supported by ONR Contract No. N00014-75-C-0738.

5) how to optimize the performance of each attack mission (e.g., path optimization, use of EW resources, etc.).

The rules of the battle are as follows:

There are n targets, each target T having its own initial military value V(T) assigned to it. There are m defenses protecting the approaches to the targets. Friendly forces consist of q aircraft carriers with NAi airplanes on board a given airplane i.

Each airplane A will carry out an assignment against one of the enemy objects and is allowed to choose the safest path for reaching that object within the available fuel allowance fi. If an airplane A attacks a target T, then A will destroy T with probability PK(A → T). If an airplane A attacks a defense D, then A will destroy D with probability PK(A → D). If an airplane A flies near a defense, then it will be shot down with probability PK(D → A), which depends on the proximity and the duration of the exposure. All single events are assumed independent. For example, if an airplane flies over two defenses, D1 and D2, then its probability of survival PSur(A) will be equal to

$$\mathrm{PSur(A)}=[1-\mathrm{PK(D1}\rightarrow \mathrm{A)}]\cdot[1-\mathrm{PK(D2}\rightarrow \mathrm{A)}] \qquad (1)$$

The objective is to find the sequence of assignments that maximizes the expected success E(L) of the attack, which is equal to

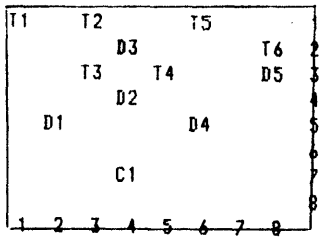
$$\mathrm{E(L)}=\sum_{i=1}^{Nn} \mathrm{V(Ti)}\cdot(1-\mathrm{PS(Ti)}) \qquad (2)$$

where PS (Ti) stands for the probability that Ti survives all the attacks directed against it. The value of PS(Ti) depends on the the number of airplanes attacking it, on the probability that these airplanes will avoid all the defensive threats along their paths, and on the probability of their success against the target.

Example 1:

Figure 1 introduces a small example of our scenario. This example will be referred to several times throughout this paper. Figure 1 represents an aerial view of the battlefield. The field is divided into an 8 x 8 square grid. There are six targets, T1 through T6, in the top half of the field. On the right we can see the values V(T) associated with each target T. For example, target T3 is the most valuable with 350 points while T1 is the least valuable with only 100 points. Five enemy defenses protect the approaches to the targets, D1 through D5.

There is one aircraft carrier C1, with twelve airplanes on it. For simplicity of discussion, all the airplanes are presumed to have identical capabilities, as do all the enemy defenses. Each airplane has capacity for f = 17 units of fuel. It takes two units of fuel to fly from the center of one square to the center of the adjacent one if one is flying vertically or horizontally; it takes three units of fuel if one is flying diagonally, as seen in Fig. 2.



	P_E		Value:
D1	.99	T1	100
D2	.99	T2	300
D3	.99	T3	350
D4	.99	T4	300
D5	.99	T5	150
		T6	200

Airplanes: A1 A2 A3 A4 A5 A6 A7 A8 A9 A10 A11 A12 (all on board C1)

Fig. 1 Example scenario

An allowable flight path is thus a chain of vertical, horizontal and diagonal transitions whose "fuel length" is no greater than 17.

The following are the values assigned to various probabilities of kill: PK(A → T) = .8; PK(A → D) = .56 probability of kill by a defense D against an airplane A (i.e., PK(D→ A), that is flying by is computed accordingly to the following rule (see Fig. 3): every time an airplane transitions into one of the eight squares next to defense D, the airplane will be shot down with probability of 20%, while a transition into the central square, where D is situated, entails a 30% probability of kill.

MODELS OF PLANNING AND CONTROL

There can be many ways in which attacks can be executed. For example, they could be executed in parallel all at once, or they could be executed one at a time, or in several attack waves. The execution structure of a particular attack in real life depends on the situation in general and on the time limitations and intelligence capabilities in particular.

As our initial models of control we chose the two most representative situations. They come from the opposite extremes of the spectrum of all available controls. They are called Closed -and Open-Loop Control Models, for the reasons that will be described subsequently.

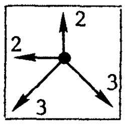


Fig. 2

Fuel consumption

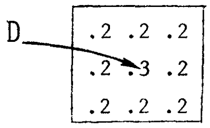


Fig. 3

Defensive threat probabilities

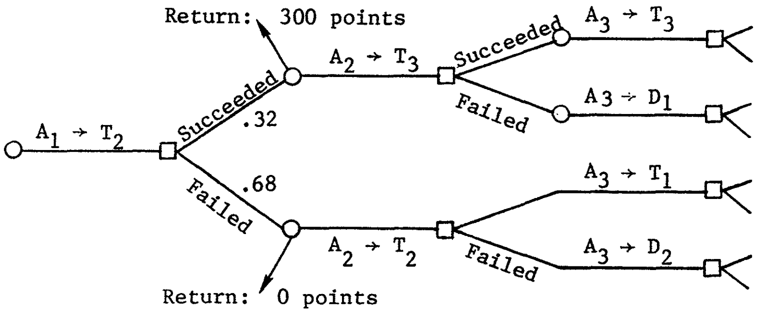


Fig. 4 Decision tree for the closed-loop scenario

Closed-Loop Control Scenario

Missions are executed sequentially, each new one starting after the completion of the preceding one, with the knowledge of its outcome. Thus we have a perfect feedback situation.

Example 2:

Let us consider the situation in Example 1 (see Fig. 1). A decision maker may decide to start his attack by sending one of his airplanes, say A1, against T2. The optimal path computation routine, discussed subsequently in this paper, predicts that A1 has a 32% chance of destroying T2. In the case that A1 is successful, the new battle position is obtained from that of Fig. 1 by deleting A1 and T2 from consideration. Notice that we have accumulated 300 points as a result of destroying T2. In this situation we may decide that the next target to attack would be T3, and send A2 against it.

On the other hand, our first assignment (A1→ T2) could have failed with probability 68%. Then we would have faced the same position as in Figure 1, except that now A1 is no longer available. No points have been accumulated, of course. This situation requires its own new decision, and we decide to send A2 against the T2, also.

Notice that different outcomes can thus entail different future assignments. For each of the two possible outcomes of the first decision , there will be a second decision with two possible outcomes, and so on.

Example 2 illustrates the point that a complete solution to the closed-loop problem has to be in the form of a decision tree, for example that in Figure 4.

In the closed-loop scenario the equation (2) for the expected return function becomes:

$$E(L)=\sum_{i\varepsilon OC}\ \sum_{j=1}^{Nn} V\,(Tj)\cdot I_{Tj}\cdot(i)\cdot Prob(i) \qquad (3)$$

where OC is the set of all final outcomes , and Prob (i) is equal to the probability of final outcome i, and

$$I_{T_j}(i)=\begin{cases} 1 - \text{if } T_j \text{ has been destroyed} \\ \quad\ \text{in outcome } i \\ 0 - \text{otherwise} \end{cases}$$

Open-Loop Control Scenario

In this scenario all missions are executed sequentially, but the outcomes of earlier missions do not get fed back to the controller and thus do not influence the future assignments. In the open-loop case, the probabilistic outcome of each attack is predicted before its execution.

Consequently, there is no advantage in waiting for the outcome of one mission before proceeding with the next one, and the whole attack sequence can be planned in advance.

Example 3:

Let us see how the open-loop control will work for the scenario in Example 1. A particular open-loop attack plan is displayed in Figure 5.

This plan assigns five airplanes to the task of creating holes in the enemy defensive line, as a first wave of attack. In the second wave, the remaining seven airplanes will then proceed with their attacks against the targets: target T3, the most valuable one, is attacked twice, while the other five targets are attacked once each.

After the first wave attack is completed, all the enemy defenses still have to be taken into consideration, but the three defenses that were subject to our attack now have a very low, probability of existence (PE) as seen in Fig. 6.

$A_1 \rightarrow D_1$
$A_2 \rightarrow D_1$
$A_3 \rightarrow D_3$
$A_4 \rightarrow D_4$
$A_5 \rightarrow D_4$
$A_6 \rightarrow T_2$
$A_7 \rightarrow T_3$
$A_8 \rightarrow T_4$
$A_9 \rightarrow T_5$
$A_{10} \rightarrow T_6$
$A_{11} \rightarrow T_1$
$A_{12} \rightarrow T_3$

Fig. 5

Attack sequence for the open-loop scenario

D1	.08
D2	.99
D3	.47
D4	.99
D5	.99

Fig. 6 Probabilities of existence of defenses after first wave of attack

T1	25	75
T2	97	203
T3	72	278
T4	136	164
T5	37	113
T6	104	96

Fig. 7

Final values of targets

Attacks against the targets are also executed in the "probability of existence" sense. Figure 7 summarizes the outcomes of the second wave attack. The numbers in the second column stand for how much value we managed to extract from each target, and the numbers in the first column stand for its remaining value. For example, target T3, which had 350 points in the beginning, is now worth only 72 points, with 278 having been already extracted by airplanes A7 and A12.

The reason the two control scenarios described above are so important is that almost all real-life feedback command and control situations can be viewed as being a combination of those two.

The optimal scores for the two scenarios also provide upper-and lower-bounds for the performance of any "combination" scenario. The closed-loop case provides much more flexibility and responsiveness to the decision maker than the open-loop case, and thus supplies the higher performance bound (Rutenburg,1982a).

Because of the importance of each of the two control scenarios, we have worked on modeling and solving each of the two. The same mathematical ideas apply to both scenarios, although the algorithmic details are different.

MATHEMATICAL SOLUTIONS

In this section we present some of our results in developing efficient mathematical algorithms for finding optimal solutions to mission planning problems.

The common denominator in our approach to the solution of problems of this class lies in decomposition of a problem into a hierarchy of sub-problems. In particular, the basic fundamental scenarios that we are dealing with in this paper can be decomposed into two levels of optimization subproblems. The top level is responsible for finding the optimal overall sequence of high-level mission assignments, while the lower level is responsible for finding the optimal use of resources (fuel, for example) within each flight mission and for evaluating the likelihood of success of that mission.

The idea that ties the two levels together is that finding a good high-level plan requires the ability to estimate the needs and consequences of individual missions, which is exactly what the low-level subproblem is responsible for. Therefore, the lower level can be viewed in this context as an evaluation subroutine for the top level.

Our initial approach to path optimization is that of dynamic programming, with remaining fuel representing the stage variable and position coordinates being the state variables. It follows the spirit of earlier works in this area (for example, Wishner and Payne, 1979).

The low-level sub-problem, which in our scenario involves path optimization, has been studied by many researchers over the years. The higher-level problem of optimal mission assignment has not, to our knowledge, been properly addressed in the past. Only highly ad hoc and heuristic solution methods reported in the literature (Callero, Jamison, and Waterman,1982; Case and Thibault,1977; Engelman, Berg, and Bischoff, 1979). It is therefore a major goal of our research to develop a systematic mathematical theory of, and algorithmic solutions to, global mission planning.

DYNAMIC PROGRAMMING APPROACH TO MISSION PLANNING

Mission planning is inherently a very complex computational problem. Trying to solve even an average-sized problem using methods of direct enumeration of all possible attack sequences proves to be an impossible task due to the faster-than-exponential growth of the problem size.

A more efficient approach is to apply the dynamic programming method to the higher-level sequence assignment problem.

Dynamic Programming Formulation for Open-Loop

The current state of the battle under this scenario is characterized by the number of aircraft available on each carrier, by the probability of existence of each enemy defense, and by the current value of each target. Thus, a state S can be defined as

$$S=(NA1,\ldots,Naq,\ PE(D1),\ldots,PE(Dm),\ PE(T1),\ldots,\ PE(Tn)), \quad (4)$$

where NAi is the number of airplanes currently available on board carrier i., and PE(X) stands for the current probability of existence of object X.

The stage (i.e., decision point) corresponds to the total number of airplanes already assigned, being equal to

$$\sum_{i=1}^{q} (\overline{NAi}-NAi), \quad (5)$$

where $\overline{N}Ai$ is the initial number of airplanes on board carrier i.

An action u in this formulation corresponds to assigning one of the available airplanes from one of the carriers on a mission against one of the enemy forces. The outcome of this action will take us to the next stage and a new state with one fewer airplane available and with the targeted enemy force having a reduced probability of existence. If that enemy force happens to be a target, say Ti, then we derive a reward L(S,u) equal to

$$L(S,u)=V(Ti)\cdot\Delta PE(Ti), \quad (6)$$

where V(Ti) is the initial value of Ti, and ΔPE (Ti) corresponds to the change in probability of existence of Ti due to the current action.

The dynamic programming equation is then represented by

$$J(S(i))=\max_{u\varepsilon Ui}\{L(S(i)),u)+J(S(i+1))\} \quad (7)$$

where J(S) is the optimal total reward that can be derived starting at state S, Ui is the set of all possible actions at state S(i), S(i+1) is the state resulting from applying action u to state S(i).

<u>Dynamic Programming Formulation for Closed-Loop</u>

A state S in this scenario is similar to that in the open-loop scenario, the difference being that an object is not represented by a probability of existence anymore; it either exists or doesn't. Thus,

$$S=(NA1,\ldots,\ NAq,\ I_{D1},\ldots,\ I_{Dm},\ I_{T1},\ldots,\ I_{Tn}), \quad (8)$$

where

$$I_X = \begin{cases} 1 \text{ - if object X exists} \\ 0 \text{ - otherwise} \end{cases}$$

The stage variable is the same as that for the open-loop case and is given by (5). An action u corresponds to the same type of a mission assignment as in the open-loop case. If the mission u is directed against an enemy target Ti then the reward L(S,u) is given by

$$L(S,\ u)=V(Ti)\cdot PS(u), \quad (9)$$

where PS(u) is equal to the probability of success of mission u, (computable by path optimization DP).

Unlike the open-loop case, which corresponds to deterministic dynamic programming, the closed-loop case gives rise to stochastic dynamic programming, due to the random nature of mission outcomes. With probability equal to PS(u) the outcome will be successful and will bring us to a new state S(i+1) with one fewer airplane available, and the indicator variable for the destroyed enemy object now reset from 1 to 0. On the other hand, with probability equal to 1 - PS(u) we will fail and result in a state $\overline{S}(i+1)$, again with one fewer airplane available, but with no enemy object having been destroyed. The dynamic programming equation now becomes

$$J(S(i))=\max_{u\varepsilon U_i}\{L(S(i),u)+PS(u)\cdot J(S(i+1)) +(1-PS(u))\cdot J(\overline{S}(i+1))\} \quad (10)$$

DEPTH-FIRST DYNAMIC PROGRAMMING ALGORITHM

In our efforts to develop an efficient implementation of the dynamic programming (DP) ideas, we discovered a new generation version of the dynamic programming algorithm. This algorithm combines the best features of both dynamic programming and of depth-first tree search algorithms. That is why it has been named Depth-First Dynamic Programming (DFDP) (Rutenburg,1982b).

DFDP is related to depth-first tree search (DFTS) by the order in which the states in the state-space are explored. Like DFTS (Winston,1979) it follows an initial "path" through the state-space from the initial state to one of the final states, thus finding an initial solution path. It then backtracks to the previous state and explores other actions available at that state, then backtracks again, and so on. In the process of the search, the algorithm computes optimal rewards for all the visited states and also updates the currently best known global solution path.

DFDP is similar to dynamic programming in its use of the underlying telescoping effects to reduce the complexity of the search involved. By telescoping effects we mean that the structure of the state-space is not a tree but rather a lattice, with many different paths leading to the same resulting state. By making the optimal reward for any state S (computed for a particular path leading to S) available for all the other paths leading to state S, DFDP algorithm reduces the computational complexity of the depth-first search from exponential down to polynomial of the order

$$O(Ns \cdot Nu), \tag{11}$$

where Ns - total number of states, Nu - average number of actions per state. This is the same complexity as that of the traditional DP algorithm, which can be described as a backwards breadth-first search technique.

USES OF DFDP

Although similar in spirit, DFDP acquires several advantages over regular DP from its depth-first search approach. Its first advantage lies in its satisficing capabilities. This refers to situations when (for example, due to time pressures), it is not necessary to find the optimal solution, but rather a satisficing one, i.e., a solution whose value exceeds a given threshold on performance. Because DFDP starts with particular solution and then keeps on improving it, this algorithm can terminate at the moment when one satisficing solution is found. This contrasts with the regular DP approach which doesn´t find any solution almost until the very end of computation.

A similar situation occurs when the algorithm is used in a real-time situation, when it is not known in advance how much time can be spent on solving a given problem. With DFDP, a good, currently best solution can be produced at any moment a solution is needed, a capability unavailable from regular DP.

Another advantage of DFDP lies in its ability to use branch-and-bound techniques to prune down the search space. Branch-and-bound is a well-known technique, used, for example, in integer programming (Nemhauser and Garfinkel,1972) and in AI tree search (Winston,1979). DFDP implementation makes the branch-and-bound ideas applicable for the first time (to our best knowledge) in dynamic programming.

The branch-and-bound approach involves computing an upper bound UB(S) on the optimal return J(S) for a given state in the state space. As we mentioned earlier, DFDP keeps track of the current optimal performance (COP) for the given problem throughout the computation. A state S need not be considered when

$$F(S)+UB(S)<COP \tag{12}$$

where F(S) is the score obtained by arriving to state S along the currently followed path. An example used in our presentation at the conference gave a good illustration of the power of this branch-and-bound approach to greatly reduce the search space. Unfortunately, space limitations do not allow us to include such an example in this paper.

One may wonder how successful the satisficing and branch-and-bound methods of DFDP are going to be. That depends greatly on the problem at hand and on the quickness with which near optimal solutions are going to be found. The latter is a function of the order in which various actions originating from a given state, are explored. A good order can be achieved by the use of a heuristic guidance function h(S) to estimate the value of the optimal performance function J(S) for a given state S. It is easy to see that if h(S) very closely approximates J(S), then the optimal solution is going to be found almost immediately, and that the algorithm efficiency decreases as h(S) becomes less reliable, just like in tree search situations, (Nilsson, 1980). Finding good heuristic guidance is a hard task from the realm of artificial intelligence, and we are currently involved in developing such heuristic functions for our mission planning problem.

DFDP has one weak point, in that the current version has on the average higher storage requirements. This fact hasn´t caused any important inconveniencies so far, but it is important to try to find implementations of DFDP with more efficient use of storage space.

<u>Example 4</u>:

Let us apply the DFDP algorithm to the open-loop scenario described in examples 1 and 3 (see Fig. 1). Figure 8 shows how the current optimal value was changing during the computation. The first solution found was worth 802 points, then the algorithm found a better solution, worth 870 points, and so it went until the optimal solution, worth 1097 points, was found. This is significantly better than the 925 points obtained from manual planning in Example 3. Figure 9 presents the summary of the optimal plan. It involves sending two airplanes to destroy defense D2, then sending two airplanes against defense D3, thus blowing a hole in the enemy defensive line and also freeing the most valuable targets, T2, T3 and T4, from the heavy defensive protection. After the two defenses are severely impaired, the plan asks for two airplanes to be sent against targets T2 and T3, and for one airplane to be sent against each of the other targets.

```
802.0592
870.2677
911.0408
960.5516
972.1386
1049.802
1094.562
1097.316
```

Fig. 8 Sequence of currently optimal scores

```
         best score is  1097.316

defense number 1 gets 0 airplanes
defense number 2 gets 2 airplanes
defense number 3 gets 2 airplanes
defense number 4 gets 0 airplanes
defense number 5 gets 0 airplanes

 target 3, VALUE is  256.3687
 target 4, VALUE is  219.7446
 target 2, VALUE is  214.6912
 target 5, VALUE is  109.8723
 target 6, VALUE is  93.75768
 target 1, VALUE is  73.24819
 target 3, VALUE is  68.58326
 target 2, VALUE is  61.05016
```

Fig. 9 Optimal plan

MYOPIC ALGORITHMS

One way of heuristically aiding DFDP in its branch-and-bound search can be achieved through the use of myopic algorithms. These algorithms are very fast, but don't guarantee finding an optimal solution. However, they almost always find a near-optimal one. Their traditional use is in the satisficing situations when a solution is needed immediately and there is not enough time to run any of the optimal algorithms. However, viewed in the same framework with DFDP (which also has satisficing capabilities) these algorithms are better used for quickly computing tight lower bounds on the optimal performance function, thus guiding DFDP in its search and also providing excellent bounds for successful branching-and-bounding.

Here is a brief description of the Basic Myopic (Greedy) Sequential Assignment (MSA) algorithm for the Open-Loop Scenario.

This algorithm has NA iterations, where NA is the number of assignments to be made. At the end of each iteration exactly one new assignment is made. It is chosen for being the "currently best" assignment. Here is how it is found.

For each as yet unassigned airplane we determine a preferred objective: that target or defense from which the airplane can derive the maximum expected value, conditioned on the assignments already made. To determine the value derived from a defense D by any given airplane, consider all assignments made previously to targets. Calculate the value that could be derived from these assignments if the given airplane were to attack defense D before those targets were attacked. The amount of improvement is the value derived from defense D. To determine the value derived from a target T, presume that all the previous assignments to enemy defenses and targets have been executed and all the involved probabilities of existence were updated. Then the value derived from T can be computed using equation (6).

Once each airplane has determined its preferred objective, the airplane capable of deriving the largest expected value is given its preferred assignment, and that assignment is added to the list. After that we start a new iteration.

The MSA algorithm performed suprisingly well in our trial runs, almost always finding near optimal solutions. However, it has a weak point which comes into play in situations when an airplane gets initially assigned to an objective, only to discover later after some other assignments have been made that it could now derive more value by attacking a different objective. This observation gives rise to an improvement: MSA Algorithm with Re-Assignment, (MSAR). This algorithm behaves similarly to the basic MSA algorithm, but, in addition, at each iteration already assigned airplanes compute how much improvement they can derive by reassigning to a different objective. If it can achieve a large improvement, it gets re-assigned.

The closed-loop scenario also has its version of the greedy algorithm. This algorithm works by myopically choosing the action to take at each decision point.

None of the myopic algorithms described above guarantee optimal solutions for all scenarios, but they seem to perform well, and do guarantee optimality for some important situations. In particular, in an open-loop case with a fixed set of defenses to be attacked, and with all the airplanes being of the same type and in the same location, the MSA algorithm is optimal. Thus, MSA can be used as a fast subroutine in an optimal algorithm for solving subproblems having the above structure. What is more interesting is that a similar result is true for the closed-loop case as well (Rutenburg,1982a). In such situations, the assignment rules become especially simple: Order all the targets in the order of decreasingly expected return value, computed using equation (9). Attack the first (highest value) target from that list. If attack fails, attack that target again and again, until we succeed. Then erase the destroyed target from the list and start attacking the new "highest value" target. Continue this procedure until all the available airplanes have carried out an attack.

Table 1 shows some familiar autonomous weapon and game-playing systems. To say to the readers of these proceedings that these systems can be effective is to preach to the choir! Nevertheless, we cannot resist mentioning two instances in which this effectiveness has been forcefully demonstrated:

* the EXOCET air-sea cruise missile which was so devastating of British ships in the Falklands war (see (Silverman, 1982) for an interesting discussion of battles between autonomous computer-driven systems);

* the SIDEWINDER air-air missile which, having been a mainstay in the US arsenal for over twenty-five years, again proved its worth against Libyan Mig-23s over the Meditteranean.

Nor can we resist mentioning that commercially available chess-playing algorithms are now good enough to win major tournaments. Indeed it has been conjectured (see (Hapgood, 1982)) that the world champion chess player may soon be a computer-hosted algorithm employing the same types of control algorithms that we envision using to control combat aircraft. We hope to do as well.

Table 1. Existing Systems of Interest

* Vehicluar Systems

Name	Launch/Target Mode
Sidewinder	Air/Air
Copperhead	Ground/Ground
Cruise Missiles	All/All
Homing Torpedoes	Sea/Sea

* Other Systems

Automatic Chess-Playing Algorithms
Certain Video Games

RELEVANT THEORY

We feel that the theory of optimal control of dynamic systems is well suited for application to the case of controlling an aircraft in combat. More specifically, the theory of optimal control of deterministic differential games in discrete time is appropriate for this purpose, at least for starters.

Deterministic differential games in discrete time are conceptually, if not mathematically, very simple (see, e.g., (Isaacs, 1965) and (Ho, 1965)). Briefly, these games model dynamic systems whose "state" evolves over time in response to the purposeful inputs or "controls" that are applied to the system at each time by each of several (usually two) players or "controllers", as well as to non-purposeful inputs from the outside world. Each controller has an opportunity to guide the state of the system toward a desired goal state by applying suitable, or perhaps even "optimal" (see note 2), controls. Of course, the two controllers usually have different objectives in mind. Figures 1 and 2 illustrate this situation.

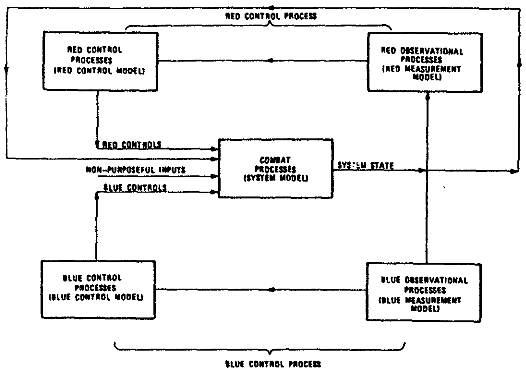
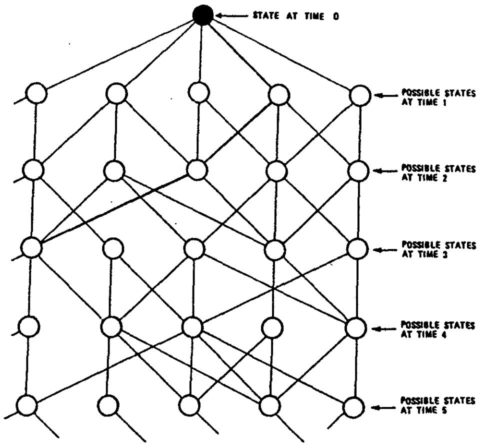


Fig 1. A Two-Sided Deterministic Differential Game



Nodes represent system states

Arcs represent transitions from one state to another corresponding to control choices by both players.

Fig. 2. Differential Game Network.

There are several reasons why we feel that the theory of optimal control of deterministic differential games in discrete time is appropriate, at least at the outset, to apply to the case of controlling an aircraft in combat:

* the dynamics of motion of aircraft are well understood, so that we do not need any "system noise";

* we are willing to assume, at least at

A TEST BED FOR THE DEVELOPMENT OF CONTROL ALGORITHMS FOR ROBOTIC COMBAT AIRCRAFT

Michael H. Moore

System Development Corporation, San Diego, California 92110

Abstract. We are interested in automating the activities of air combat vehicles. We begin by indicating the reasons why we have taken this objective. We also mention some examples of other combat systems whose control has been automated in various ways. Next, we describe the theory that is relevant to our purpose. Finally, we describe a computational testbed that we are developing to facilitate experimenting with the theory. We also mention a few of the questions and conjectures that we would like to resolve through the use of the testbed.

INTRODUCTION

We want to develop algorithms for automatically controlling the maneuvers and other activities of an autonomous aircraft in aerial combat (see note 1).

By an "autonomous aircraft", we mean an aircraft whose maneuvers are completely under the control of a computer-hosted algorithm. We envision the computer that hosts this algorithm as being aboard the aircraft itself.

There are three reasons why we think that the objective we have set for ourselves is appropriate. First, the need for an autonomous air combat vehicle seems obvious:

* pilotless combat aircraft may well be superior to their manned counterparts in several important attributes:
    - better performance (e.g., more rapid changes of direction and speed);
    - simpler/smaller design (e.g., no life support systems are necessary);
    - lower costs (lower initial cost, lower cost of maintenance);
    - higher tooth-to-tail ratio (smaller manpower intensiveness all along the tail).
* conventional manned combat aircraft may no longer have a reasonable chance of survival against modern surface-air or air-air weapons (such as the one we hope to develop ourselves as discussed in this article!);
* even without completely removing human pilots from combat aircraft, it would be desirable (see National Research Council, 1982) to automate some of their chores, such as maneuvering the aircraft in combat.

Secondly (and what probably motivates us the more), our objective seems timely. More specifically, our objective seems timely because computers that may be powerful enough to perform the computationally intense task of autonomous decisionmaking in real time, and compact enough to fit aboard an aircraft, have recently become available at reasonable cost. Furthermore, even more powerful and compact computers using VLSI/VHSIC technology will become available a few years hence (see, e.g., (Bishop, 1981)).

Finally, we have set this objective for ourselves because we think that developing automatic control algorithms for combat aircraft is interesting and fun. Indeed, this objective is just the simplest interesting case of a more ambitious objective: that of developing control algorithms for arbitrary weapon systems and military forces (see (Moore, 1981)).

---

1. It is interesting to note that what we propose to do for combat aircraft -- namely, develop algorithms that can play well (better, hopefully, than human pilots) -- is similar to what has been done in recent years for the game of chess. It is also interesting to note that chess may actually be a more complex game than aerial combat. In chess, there are multiple pieces of different types which, besides supporting one another, often get in each other's way. In aerial combat, this situation does not occur (not, at least, to the same degree).

CONCLUSIONS

Mission planning is an important and a very complicated area of decision making. Therefore, modern decision makers need a flexible, user-controlled decision aid. Analytical algorithms are needed for the solution of analytical facets of such decision aid. In this paper we have presented a basic analytical theory of mission planning, together with several analytical algorithms for the solution of mission planning problems. This theory needs to be elaborated and extended to more complex scenarios. Computational efficiency of the developed algorithms also deserves further study. In addition, ways for incorporating human guidance into the algorithms need to be investigated.

REFERENCES

Callero, M., Jamison, L., Waterman, D.A., (1982). TATR: An Expert Aid for Tactical Air Targeting. Rand Corporation, Santa Monica, CA

Case, K.E., and Thibault, H.C.,(1977).A Heuristic Allocation Algorithm with Extensions for Conventional Weapons for the Marine Integrated Fire and Air Support System. School of Industrial Engineering and Management, Oklahoma State University, Stillwater.

Engelman, C., Berg, C., Bischoff, M., (1979). KNOBS: An Experimental Knowledge Based Tactical Air Mission Planning System. Proc. of VI, International Conference on Artificial Intelligence., pp. 247-249.

Larson, R.E., and Casti, J.L., (1978). Principles of Dynamic Programming, Part 1. Marcel Debber, Inc.

Marsh, J.P., and Grossberg, M., (1978). Research Report for the Advanced Weapons Management System (AWMS). Systems Control, Inc.

Nemhauser, G., and Garfinkel, R., (1972). Integer Programming. John Wiley, New York.

Nilsson, N.J., (1980). Principles of Artificial Intelligence. Tioga Publishing Company, Palo Alto, CA

Rutenburg, V., (1982a). Depth-First Dynamic Programming Algorithm. Technical Report TR-1026-2, AI&DS, Mountain View, CA

Rutenburg, V., (1982b). Dynamic Sequence Assignment: Scenarios and Algorithmic Solutions. Technical Report TR-1026-3, AI&DS, Mountain View, CA

Slagle, J., Cansone, R., Halpern, E., (1982). BATTLE - An Expert Decision Aid for Fire Support Command and Control. NRL Report 4847, Washington, D.C.

Winston, P.H. (1979). Artificial Intelligence. Addison-Wesley, Reading, MA

Wishner, R.P., and Payne, J.R., (1979). Resource Allocation for Naval Platforms and Weapons. MIT/ONR Workshop on distributed Information and Decision Systems.

the outset, that the two controllers have complete and perfect information about the state of the combat system at all times (i.e., we do not need any "measurement noise");

* we are also willing to assume at the outset that controls can be applied to the combat system without any distortion (i.e., there is no "control noise);

* discrete time, which can be as near to being continuous as we like, is convenient for computation.

Later, we may decide to "go stochastic" (i.e., apply the theory of stochastic differential games) for the following reasons:

* to account for detail that would otherwise go unmodeled (e.g., winds, sensor errors, control slop);

* to intentionally trade detail of representation away (without ignoring it completely) for computational simplicity.

For differential games of practical interest, the number of control policies for the players -- that is, the number of paths through the game network of fig. 2 -- is combinatorially large. Thus, to try to find an optimal control policy by exhaustively considering all possible control policies one by one is to die an ugly computational death. Indeed, it is usually infeasible to find an optimal control policy even when we use such computation-reducing techniques as dynamic programming (an intelligent breadth-first search of the game network), or branch and bound (sometimes called "alpha-beta pruning"). Therefore, we must be content with the best control policies that we can obtain with a reasonable amount of time and effort.

There are several techniques for finding suboptimal, but hopefully "good", control policies. One such technique, which we might call the "lookahead-in-bounded-moving subnetworks" technique (we will not call it that!) is as follows:

(1) form a subnetwork of the game network that begins at the current state (which we are assuming is always known) but which is bounded in breadth and depth;

(2) find a "subnetwork-optimal" control policy -- that is, a sequence of controls that is optimal in the context of the subnetwork;

(3) apply the first control of this sequence;

(4) observe the resulting new state; and go back to (1).

Figures 3 and 4 illustrate this technique. This procedure (which, incidentally, is used in all chess-playing algorithms) may be described as repeatedly re-solving the problem of finding a good first move.

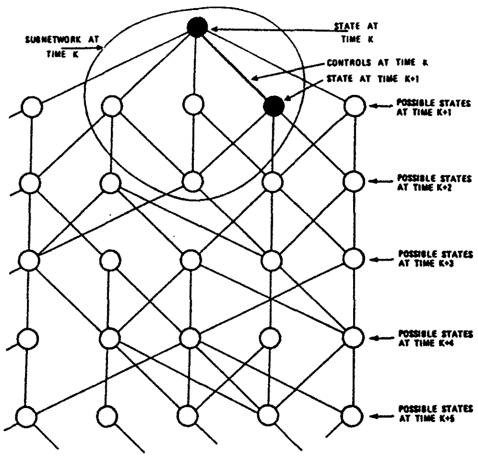


Fig. 3. Lookahead in bounded moving subnetworks.

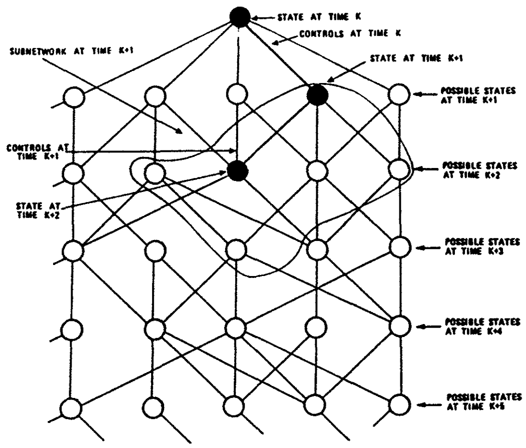


Fig 4. Lookahead in bounded moving subnetworks.

Another technique for finding suboptimal control policies for the players is the "heuristically guided search" or "best first search" technique of artificial intelligence (see Nilsson (1980)). This technique can be applied either to the entire game network or to the subnetworks in the procedure described above. Of course, the quality of the control policies that result from these searches depends on the quality of the heuristics that are used. Hence the need for a testbed with which to look for good heuristics.

A COMPUTATIONAL TESTBED

We are developing a computational testbed for experimenting with the theory described above. The testbed exists (on an unofficial basis) at the Advanced Command/Control Architectural Testbed (ACCAT) at the Naval Ocean Systems Center in San Diego, California.

A user of the testbed can specify at running time various models and parameters, such as:

- the dynamics of motion of an aircraft (as a function of the controls that may be applied to it);
- the control spaces for each aircraft
  - turning capabilities
  - capability to change speed
  - minimum and maximum speeds
- figure of merit specifications for each aircraft (i.e., information about how each aircraft values various combat states).

The user may also select at running time the method by which he wishes to define and search subnetworks: exhaustive search; dynamic programming; heuristically-guided search.

The most serious deficiency of the testbed at this point is that we have only a minimal graphic output capability (printer plots only). We feel that we, and other users of the testbed, should have the very best in graphic display capabilities -- including color graphics and animation -- as an aid to finding the control algorithms that we seek.

We plan to use the testbed to try to resolve such questions as:

- What is the utility of one aircraft being able to look ahead (in a subnetwork) more broadly or more deeply in time than the other?
- What is the effect of an aircraft not having correct information about the capabilities of his own and the other aircraft?

We also plan to use the testbed to verify or lay to rest a few conjectures:

- high speed and high maneuverability guarantee a win;
- high speed alone is enough to guarantee a win (here, we are recalling what happens in the classic "homicidal chaffeur" example of a deterministic differential game (see Isaacs, 1965));
- high maneuverability is useful only when speed capabilities of the two aircraft are comparable;
- high speed can permit us to replace our complicated control algorithms that search subnetworks with simple fast-running control algorithms that merely solve pursuit/evasion problems.

Currently, we are using the testbed to gain experience in defining and searching subnetworks. We are making a series of runs in which we are exhaustively searching subnetworks having a particular "hand-shape", as illustrated in fig. 5. For computational simplicity at this early stage, we are using a very simple model of the dynamics of aircraft motion in which, once the velocity vectors of the aircraft are known, the position vectors are determined by simply dead-reckoning them ahead. We are also taking the control spaces for each aircraft at each time as being a finite set of perturbations of the aircraft's velocity vector at that itme. Later, after we gain experience in searching subnetworks, we will refine these models -- possibly all the way to a fully detailed representation of the dynamics of aircraft motion with six degrees of freedom.

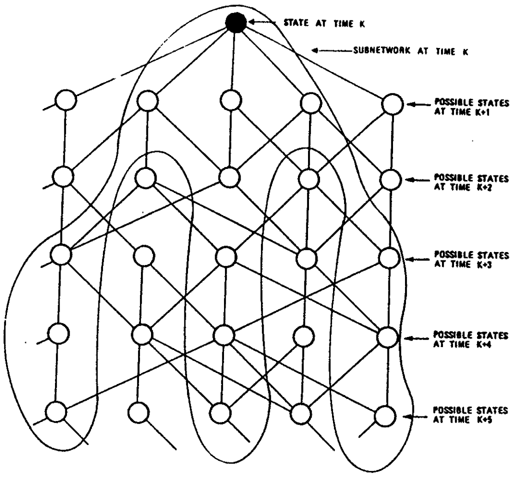


Fig. 5. Finding control strategies for autonomous combat aircraft will require looking ahead deeply in time.

REFERENCES

Jerry G. Bishop, "Can a Supercomputer be Built? Team at IBM Grows More Confident. A Critical Step: Cramming 1000 Circuits Onto Quarter-Inch Chip. The Japanese Could Be First", Wall Street Journal feature article, 27 February 1981.

National Research Council (Committe on Automation in Combat Aircraft), "Automation in Combat Aircraft", National Academy Press, Washington D.C., 1982, 136 pp.

REFERENCES (continued)

Fred Hapgood, "Wired to Win: A Chess-Playing Computer Program May Soon Be the World Champion. But is it Really Smart", Technology Illustrated 2-5 (October 1982), pp 91-96.

Y. C. Ho, "Review of R. Isaacs' Book on Differential Games", IEEE Transactions on Automatic Control AC-10/4 (October 1965), pp 501-503.

Rufus Isaacs, "Differential Games", John Wiley & Sons, New York, 1965 (reprinted in 3rd edition by Robert E. Kreiger Publishing Company, New York, 1975).

Michael H. Moore, "On the Automatic/Optimal Control of Weapon Systems and Military Forces", presented at the 48th Symposium of the Military Operations Research Society, Monterey California, December 1981.

Nils J. Nilsson, "Principles of Artificial Intelligence", Tioga Publishing Company, Palo Alto California, 1980, 476 pp.

Burt Silverman, "Battle of the Microchips", Time Magazine, 17 May 1982, pp 26.

MODELING HUMAN DECISION PROCESSES IN COMMAND AND CONTROL

Joseph G. Wohl
Vice President, Research and Development

Elliot E. Entin
Member of the Technical Staff

ALPHATECH, Inc., 3 New England Executive Park, Burlington, Massachusetts 01803

Abstract. A normative-descriptive model is described as a way to understand the decisionmaking activities of Naval commanders. This model incorporates the individual's "mental model" of the situation which can act to stimulate hypotheses and a "perception processor" for analysis of incoming data. The mathematized model represents the decisionmaker as stochastic controller. Hypotheses held for consideration are evaluated by parallel Kalman filters which output state estimates and error sequences. The later is then passed to a likelihood function calculator which in turn inputs a Bayes-Rule calculator. Output of this block is used as input to a tree production processor for option and response planning.

BACKGROUND

The long-term goals of this ONR-sponsored project are (1) to better understand how a commander, as part of a Naval command structure, conducts his decisionmaking activities in attempting to achieve certain military and organizational goals; and (2) to reflect this understanding in a logico-mathematical representation of the process. The modeling approach is based on the postulate that a well trained (i.e., "expert") human tries to develop a more or less optimal solution to his decision problem, subject to his own psychophysical and cognitive limitations and to the uncertainties inherent in the problem. Our approach is thus based on a blend of the normative, optimization-based concepts of optimal control and those of dynamic decisionmaking theory drawn from cognitive and engineering psychology.

The first step was to describe the decisionmaking processes employing Wohl's (1981) Stimulus-Hypothesis-Option-Response (SHOR) paradigm. The SHOR paradigm is basically an extension of the stimulus-organism-response (SOR) paradigm of classical behaviorist psychology to provide explicitly for the necessity to deal with two realms of uncertainty in the decisionmaking process: (1) information input uncertainty, which creates the need for hypothesis generation and evaluation; and (2) consequence-of-action uncertainty, which creates the need for option generation and evaluation.

PROGRESS

A major task during the first year has been the elaboration of the SHOR paradigm in light of current cognitive science literature and the particulars of the Naval Anti-Submarine Warfare (ASW) problem. The original SHOR paradigm held that hypotheses are generated in response to some stimulus. In turn, the hypothesis ultimately selected stimulates a set of option alternatives from which a response is selected. To this conception we have added the concept of a "mental model" - the individual's internal representation of the problem situation which represents his current understanding of the variables involved and the functional relationships believed to exist among them (see Fig. 1). Thus, if we conceive of an individual's mental model as serving much like a "theory," it can function as a hypotheses generator in the same way theories are used to derive hypotheses in science. We also see the mental model, serving in the additional capacity of a "problem classifier," functioning in analogical reasoning. Given a current representation of the problem situation (mental model) and a general classification of the problem area (problem classifier), an individual can search his knowledge base for a similar or analogical situation, which might suggest possible solutions.

A "perception processor" is assumed to intervene between the input stimulus and the remaining processes of the paradigm. Its major

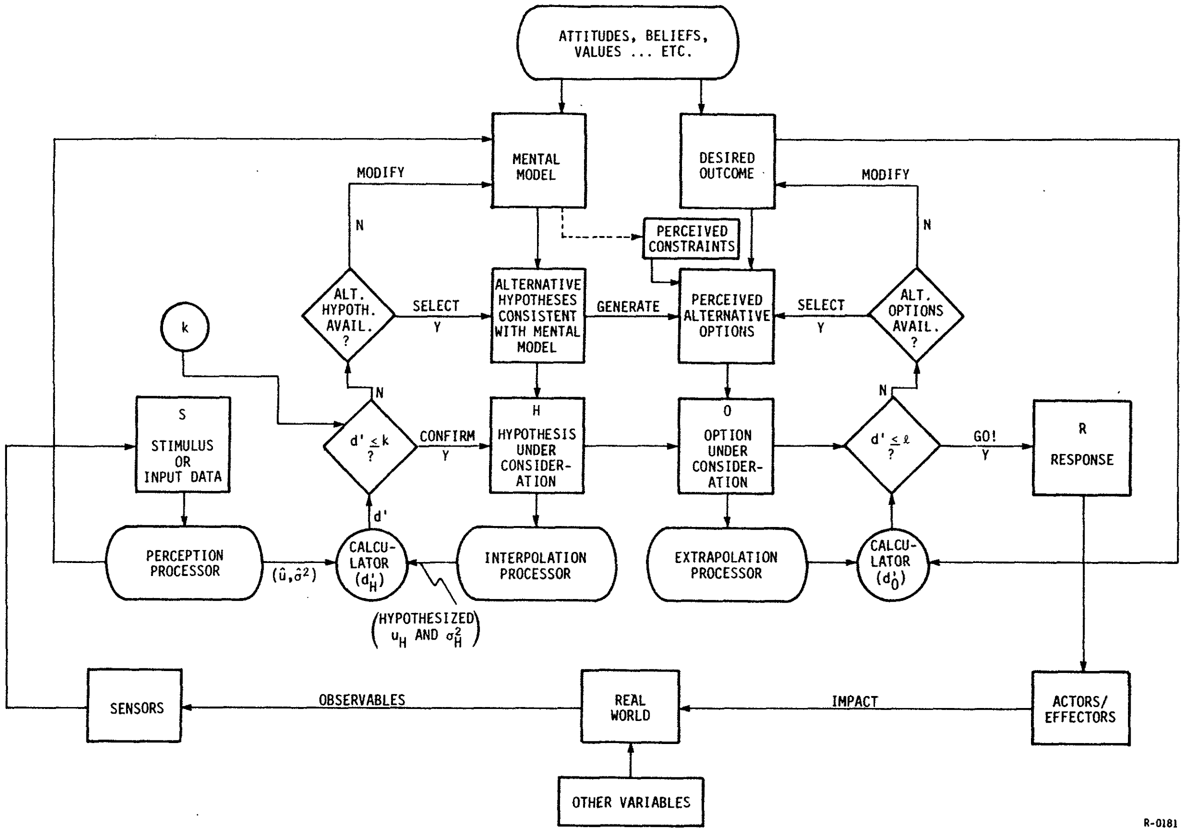


Fig. 1. Elaborated SHOR Paradigm

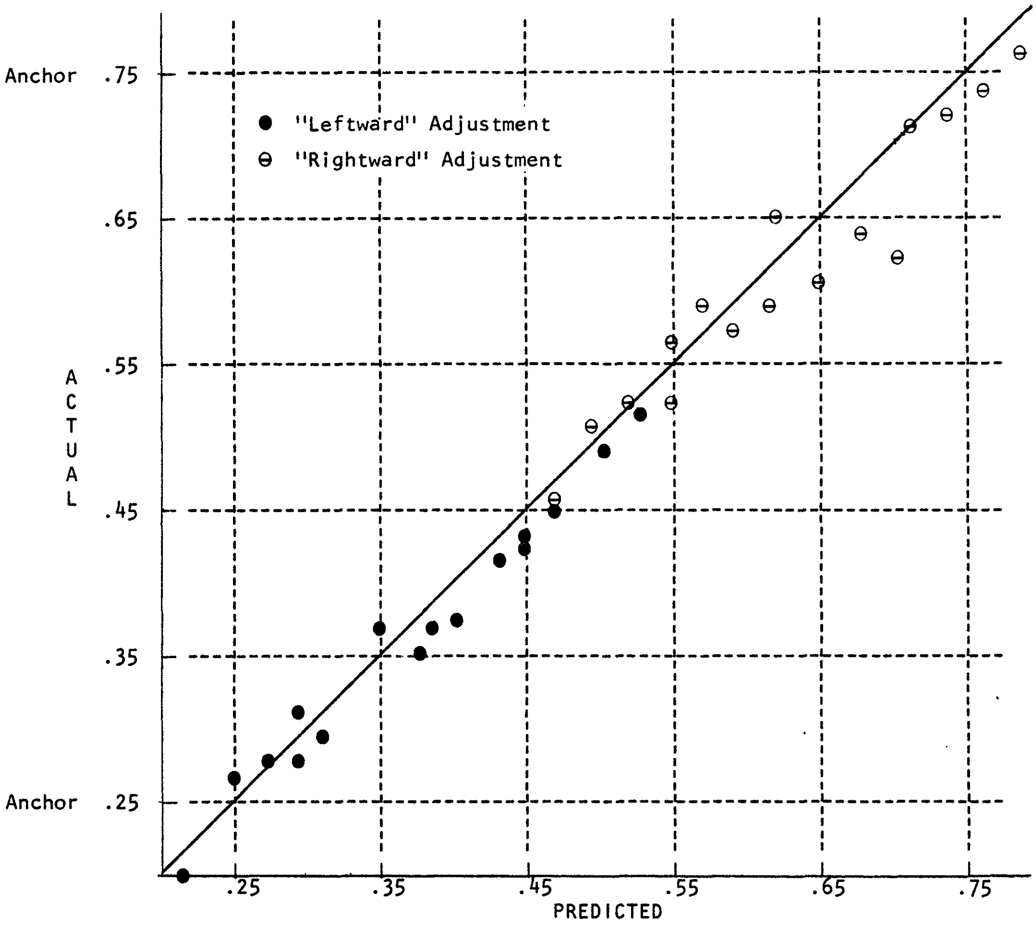
functions are identification and interpretation of incoming data; i.e., data processing. The introduction of two psychological constructs - attitudes and cognitive styles - allows for consideration of individual differences in perception processing, problem solving, action selection, and error commission.

The concept of parallel or simultaneous processing is indirectly addressed in the elaborated SHOR paradigm by considering that multiple hypotheses and options are dealt with in parallel, with constant interactive comparing among them.. Also, our assumption of a distributed knowledge base is consistent with the possibilities of parallel processing.

Most recently the elaborated SHOR paradigm has served as a foundation from which to develop a mathematized version of the model. This model represents the human decisionmaker as a stochastic controller, i.e., a controller operating in an uncertain environment, with multiple hypotheses about "what is going on" in the world. The input stimuli are measurements made of the real world; state estimates and probabilities are then formulated about the real world, and control actions are selected to affect the real world.

Option consideration and selection are reasonably well researched areas in the decision-making literature, but hypothesis consideration and selection are not. Thus, much of the development time spent on the model has been focused on the hypothesis component. The model component proposed for the hypothesis evaluation function stipulates that the data is passed to a state-estimator, which is composed of several parallel Kalman filters (one for each hypothesis). Each Kalman filter provides two key outputs; the first is a state estimate based on the incoming data, and the second is the error sequence, which is the difference between the measurement data and the filters' prior estimates of what the measurement should be. The error sequence is an indication of how well the data matches the filters' expectations, and thus provides key information about how well the data supports each hypothesis. Each error sequence is then passed to a likelihood function calculator, which computes the probability of a data sequence being observed given the hypothesis is true (and evaluated for the actual data received). This function is used as the input to a Bayes Rule calculator, which computes the desired posterior or post-measurement probability of the hypothesis being true. The output of this block is then used as an input to a tree production and search processor for option evaluation and response planning.

Ability of the Kalman filter to represent actual data from psychological experiments on probability estimation and to account for anchoring and adjustment biases (Lopes (1981)) is indicated by the results shown in Fig. 2.



$$P = \alpha S_1 + (1-\alpha) S_2 \left\{ \begin{array}{ll} \alpha & = 0.30 \\ S_1 & = \text{first sample} \\ S_2 & = \text{second sample} \\ P & = \text{predicted adjustment} \end{array} \right.$$

SP237

Fig. 2. Actual Lopes Data From Exp. #2, Heterogeneous Pairs

These results are quite encouraging, and will be followed by an attempt to extend them to reflect individual differences due to cognitive style.

Another activity has been the completion of several ASW scenarios modeled after the "ONRODA problem." We intend to follow an ASWC's decisionmaking processes through these scenarios employing the SHOR paradigm to analyze the decisionmaking and problem solving processes. This approach should, if successful, lead to a normative-descriptive model of human decisionmaking which is amenable to logico-mathematical representation, and can ultimately be tested against naval anti-submarine warfare scenarios and compared with the decision processes of real commanders faced with the same scenarios.

REFERENCES

Wohl, J.G., "Force Management Decision Requirements for Air Force Tactical Command and Control," IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-11, No. 9, September 1981, pp. 618-639.

Lopes, Lola L., "Averaging Rules and Adjustment Processes: The Role of Averaging in Inferences," WHIPP #13 (Wisconsin Human Information Processing Program), University of Wisconsin, December 1981.

HUMAN MEMORY LIMITATIONS IN MULTI-OBJECT TRACKING

Frank L. Greitzer, Richard T. Kelly, & Ramon L. Hershman

Navy Personnel Research and Development Center, San Diego, CA 92152

Abstract. Basic performance data were obtained on the effect of critical task variables on the ability of unaided observers to retain the tracks of multiple moving objects. Observers viewed computer-generated displays in which objects moved in random linear trajectories with no track history. Immediate recall tests on object trajectories showed that unaided humans can keep track of up to about seven moving objects, but performance improves as the interval between display updates is increased. Mathematical models of human memory were developed to account for the observed performance. The analysis of human information processing limitations should be extended to more complex operational tasks to support system designers with quantitative estimates of operator performance.

INTRODUCTION

Most C^2 functions in the Combat Direction Center rely heavily on humans to process information and make decisions. These personnel interact with hardware and software subsystems (by hooking targets, entering data, etc.) and thereby contribute to the overall knowledge base of the system. If we ask "What is this base of knowledge" or "What does the system know," we might obtain the answer from the computer's database on the various tracks of interest and their attributes.

However, our chosen focus is on limits in the operator's processing and on what they know, particularly on what humans can remember about the tactical environment when limited to their own resources. For this purpose, consider a primitive tracking problem that lacks all external aids and the computer support that would normally provide target symbology, identification of track numbers, etc. Consider also an operator freed from disruptive auxiliary tasks, such as communications. Our concern in this environment is with a single aspect of tactical knowledge: the unaided ability to remember the movements of multiple targets.

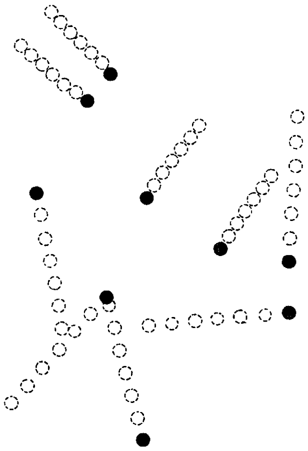
The objective of the research was to establish a basic understanding of the information processing and memory requirements for the tracking of moving objects by the unaided observer. The approach was: (i) to obtain human performance data on the effect of critical variables on unaided multi-object tracking; and (ii) to develop and assess mathematical models that quantify human performance limits in this basic task.

EXPERIMENT

Tracking Task

The tracking task was implemented on a Tektronix 4027 color graphics terminal driven by a Tektronix 4051 microcomputer. The display represented a series of seven snapshots of idealized target objects as they moved on linear paths across the CRT. The objects were identical 6mm pale yellow discs on a black background and were initially positioned at random locations on the screen. At regular intervals, all object positions were updated simultaneously. Except for the brief (.5 sec.) persistence of the phosphor, there was no display of track history. Thus, in order to track the objects, the observer had to first derive their courses by integrating successive displays. Since no aids such as grease pencil, pencil and paper, etc. were allowed, this information had to be maintained in memory until tested.

Figure 1 illustrates a nine-object tracking problem to which we have added track history (the dashed circles) for the benefit of the reader. Note that the observer viewed only the solid discs, which corresponded to the objects' present positions. Each object maintained a constant but randomly selected heading and moved at a constant display speed of either 6mm or 12mm per update. On the average, half of the objects moved at each speed. All objects were constrained to stay on the screen for at least eight updates. The paths of the objects were permitted to cross, but their positions could not overlap. The display then loosely resembled a noise-free radar screen with no external tracking aids.



Note. The dashed circles indicate previous object positions; only the current positions, shown as the solid discs, were visible to the observer.

Fig. 1. Trajectories in a representative nine-object display.

A sequence of seven object-position snapshots comprised a single trial, after which the screen was erased. This was a cue for the observer to respond on a hard copy facsimile of the most recent display. Observers were instructed to place an "X" at the position that each object would occupy if the display had been updated once more. They were encouraged to be as accurate as possible but to guess when uncertain. No feedback was provided. After 1 minute a warning tone was presented, signaling the imminent start of the next trial.

Performance Measurement. A simple, three-valued scoring rule was used to measure tracking accuracy. If the observer's predicted position X was within ± 15° of the true trajectory, two points were scored. If the absolute error in X was greater than 15° but not greater than 45°, then one point was assigned. A score of zero was given if the error in X exceeded 45°. The mean score for all objects was then taken as the observer's performance for the given trial.

Design

Six civilian observers were tested individually in six 30-minute sessions, one per day on consecutive weekdays. The first session served as practice, and these data were not analyzed. Each session consisted of 12 trials that were constructed from the factorial combination of two factors: number of objects (5, 7, or 9) and the time between updates, or interupdate interval (IUI = 5, 8, 13, or 18 sec). The order of the 12 types of trials was random in each session and for each observer. All observers viewed identical stimulus displays but in a different order. Speed of the objects was also varied to complete the 3 (number of objects) x 4 (IUI) x 5 (test session) x 2 (speed) design. All statistical analyses employed the .01 level of significance.

Effects of Major Variables

Detailed analyses are given in Greitzer, Kelly & Hershman (1982). Mean tracking accuracy exceeded chance performance in all conditions. The speed variable was not a significant factor, nor did it interact with any of the other variables; hence it was eliminated as a factor and the collapsed data were reanalyzed.

The effects of IUI and the number of objects to be tracked are shown in Fig. 2. The number of objects had a significant effect (F(2,10)=18.42); as expected, a memory limitation is evident within the range of five to nine objects. The effect of IUI was also significant (F(3,15)=9.95); in general, longer IUIs facilitated recall--presumably by permitting greater opportunity for rehearsal. However, the effect of IUI interacted with the number of objects (F(6,30)=4.96). IUI had virtually no effect with five objects, a moderate facilitative effect with seven objects, and a large facilitative effect with nine objects. When 18 seconds were available for rehearsal, the effect of the number of objects vanished. The additional time apparently helped compensate for the increased processing load.

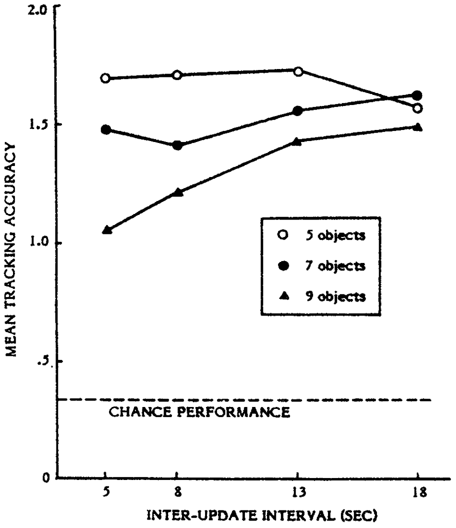


Fig. 2. Tracking accuracy as a function of number of objects and inter-update interval.

MODELS OF MULTI-OBJECT TRACKING

Information Processing Framework

It is convenient to distinguish among three functional types of memory without regard to their possible structural properties (see Fig. 3): (i) Sensory memory, which refers to information carried in "raw" or "precategorical" form, i.e., prior to the attachment of any linguistic category to the stimulus information. The sensory memory for the visual modality is called iconic memory, while the corresponding memory for audition is called echoic memory (Neisser, 1967). (ii) Primary memory, which refers to a mental process whereby a small amount of information can be held readily accessible for various operations, such as rehearsal. Rehearsal is crucial for maintenance of information in primary memory. (iii) Secondary memory, which refers to information held in more permanent form, enriched by semantic content or associations. Access to such information is not immediate as in the case of primary memory, but instead requires active retrieval processes.

We use the term "Executive" for the general management function that supervises the various strategies employed in processing the information, such as processes that transform sensory information into selected encodings, the process of rehearsal, and a variety of elaborate operations that build up or retrieve long-term representations in secondary memory.

In the specific context of the tracking task, the processes of attending, encoding, and learning are extremely rapid. The bulk of each update is presumably devoted to rehearsal, that is, the maintenance in primary memory of the learned trajectories. Only about 7 ±2 object trajectories can be retained (Miller, 1956), and unless these are maintained by a rehearsal process, the information is lost within about 20 seconds (Peterson & Peterson, 1959). Rehearsal is presumably effected by sub-vocalizations and is mediated by echoic memory. The set of object trajectories is presumed to be stored in a sequence, then rehearsed in that sequence, and later retrieved in that same sequence (Sperling, 1963). The locus of the rehearsal process is referred to as the rehearsal buffer (Atkinson & Shiffrin, 1968). If the demand for processing exceeds the capacity of the buffer, the executive must decide whether to forestall further processing of new items or to replace old items in the buffer with newly encoded ones. In either case, performance will suffer if the rehearsal capacity is exceeded.

A family of four mathematical models of the observer's processing was developed and evaluated. Each model included an encoding process, a learning process, a rehearsal process, and a response process. The models make identical assumptions for the learning and response processes. Two alternative assumptions are introduced for the number of new objects that are encoded at each update. Also, two different assumptions are considered for rehearsal.

Encoding

As the observer attends to the display, information first enters iconic memory. Further processing requires encoding, which

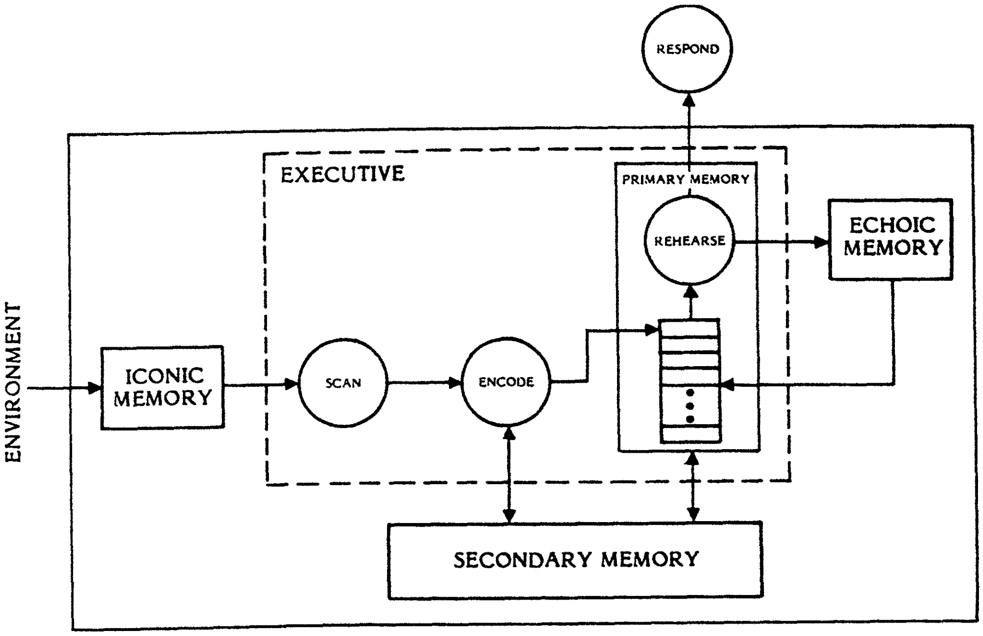


Fig. 3. Schematic model of information-processing framework.

is the transformation of this "raw" data into a form compatible with primary memory. This entails the invoking of a familiar representation for the visual datum. In general, the objects to be tracked need not all be encoded at once. First, this could prove a large processing burden at a single update, and second, the limited rehearsal capacity might be inadequate to maintain them in storage.

Definitions: Let there be Z objects in the tracking task. At the *k*th update, define E_k to be the number of objects previously encoded and A_k to be the number of additional objects that can be accomodated in the rehearsal buffer. Let $N_k = \min(A_k, Z-E_k)$.

Assumption E1: At update *k* the number of new objects encoded is precisely N_k. That is, the observer always keeps the rehearsal buffer filled to capacity.

Assumption E2: At update *k*, the number of new objects encoded is a Poisson-distributed random variable with parameter λ, truncated at zero. With this distribution it is theoretically possible to sample more objects than the N_k that can be accomodated in the buffer; in such cases the number encoded is limited to this maximum of N_k objects.

Learning

While knowledge about an object's trajectory can vary continuously from no information to virtually perfect information, it is convenient to reduce this continuum to discrete states. Consistent with the 3-valued scoring procedure used in the experiment, the proposed models assume three learning states: State U--Unencoded (nothing is known about the object's trajectory); State 1--Partial learning ($15^\circ < |\text{tracking error}| \leq 45^\circ$); State 2--Refined learning ($|\text{tracking error}| \leq 15^\circ$).

All objects start and remain in state U unless encoded. Any changes in state occur only in the initial moments of each update as objects' new positions are detected. At such times, the observer may encode an object for the first time or refine the information for a previously-encoded object. This improvement in knowledge is taken to be stochastic and is formally defined by the following transition matrix:

$$\begin{array}{cc|ccc} & & & \text{New State} & \\ & & 2 & 1 & 0 \\ \hline & 2 & 1 & 0 & 0 \\ \text{Prior State} & 1 & \beta & 1-\beta & 0 \\ & 0 & \alpha & 1-\alpha & 0 \end{array}$$

Rehearsal

Memory maintenance. Rehearsal serves as a memory maintenance function that refreshes the encoded representations that would otherwise be lost from primary memory. Restrictions on the rehearsal process ensure that only a limited number of objects can be retained. We assume the following: All encoded objects are rehearsed in sequence and form cycles in which each object is rehearsed for a required *t* seconds. The rationale for the time requirement is that any rehearsal shorter than some critical duration is ineffective in refreshing memory for the object's trajectory. A further restriction derives from the limited span of primary memory: memory is presumed to decay unless an object is rehearsed at least once every μ seconds.

Required rehearsal time. In addition to the μ restriction, the working capacity of the rehearsal buffer clearly depends on *t*, the time required for one rehearsal of an object. Two competing assumptions are considered:

Assumption R1: The required rehearsal time t for an object is fixed. That is, $t = \theta$ always.

Assumption R2: The required rehearsal time t for an object depends on the amount of its rehearsal during prior updates. In particular, t is a constant (θ) until at least τ seconds of prior rehearsal time have been accumulated, after which t decreases exponentially to ε with decay constant ν.

According to assumption R2, as objects are rehearsed over several updates they each require less and less rehearsal time. The effect is to make additional room available in the rehearsal buffer and effectively increase its capacity. This assumption departs from other interpretations of buffer capacity (cf. Norman, 1968, Atkinson & Shiffrin, 1968).

Response

After the last update, the observer's task is to indicate the next position for each object. It is assumed that the contents of the rehearsal buffer are output sequentially and that the response for each object conforms to its last encoded trajectory. A chance response is made to all unencoded objects. The 3-valued scoring system is applied to all objects, but note that each encoded object presumably receives a score (1 or 2) that corresponds to its current learning state.

Summary of the Models

Four alternative models of performance in the multi-object tracking task are obtained.

The models share identical assumptions for the probabilistic learning process (parameters α and β), for the constraint on rehearsal maintenance (parameter μ), and for the response process.

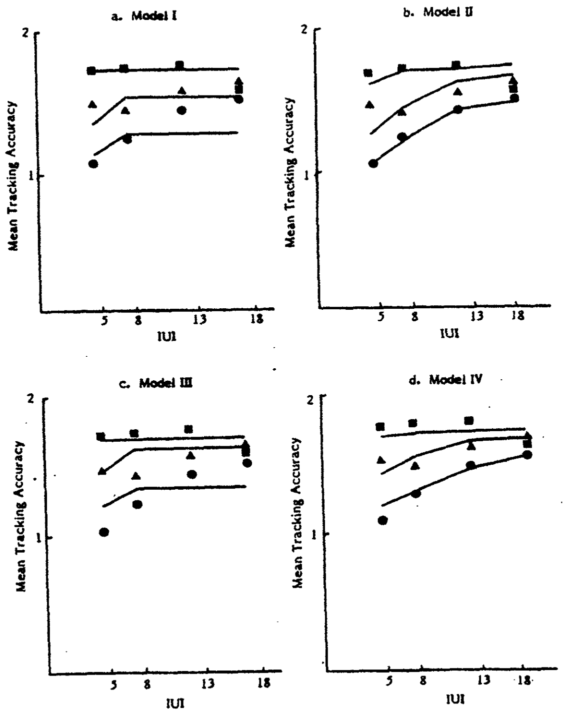
The first distinction among the models concerns the encoding of new objects. Models I and II (assumption E1) hold that enough new objects are encoded at each update to keep the rehearsal buffer filled. In contrast, Models III and IV (assumption E2) invoke a modified Poisson sampling process (with parameter 2) to govern the encoding of new objects.

The second distinction is the required rehearsal time for objects in the rehearsal buffer. Assumption R1 in Models I and III holds that the required rehearsal time is constant (parameter θ). Models of this type, when combined with the memory maintenance assumption, imply a conventional fixed-size rehearsal buffer. On the other hand, assumption R2 in Models II and IV postulates that required rehearsal time for an object varies as a function of its prior rehearsal (parameters θ, τ, ε, and ν). The effect is to create a variable-sized buffer as the required rehearsal time decreases over the sequence of updates.

Assessment of the models

Predictions of the four models are derived in Greitzer, Kelly, & Hershman (1982). Parameter estimates from least-squares fits to the observed data are shown in Table 1. Models II and IV clearly provided better fits; this is also evident in Fig. 4, which plots the observed data and the predictions for the four models.

Thus, the evidence strongly favors the rehearsal assumption R2 in Models II and IV: viz., the time required to rehearse a given trajectory decreases as the prior rehearsal time increases. We reject Models I and III, which assume fixed rehearsal times and imply rehearsal buffers with fixed capacity. They fail to predict properly the observed improvement in performance as IUI increases.



Note. Solid lines are predictions. Plotted points are for five-object displays (■), seven-object displays (▲), and nine-object displays (●).

Fig. 4. Observed data and predictions of the four models.

One cannot make a clear choice between the alternative encoding assumptions (E1 vs. E2) that distinguish Models II and IV. Model IV requires the additional Poisson parameter (λ) to govern the encoding of new objects, but its predictions were only trivially better than the deterministic "fill-the-buffer" assumption of Model II. The fits for these two preferred models are generally satisfactory, [Fig. 4 (b) and (d)], and the values of the estimated parameters (Table 1) are reasonable.

Our interpretation of human information processing in the tracking task is then as follows:

(a) If there is room in the rehearsal buffer, the observer encodes about two objects on each update. In general, this is

TABLE 1 Parameter Estimates and Prediction Errors for the Four Models

	Encoding	Learning		Rehearsal					
Model	λ	α	β	θ	μ	τ	ε	ν	Error
I	-	.58	.078	1.0	5.5	-	-	-	.089
II	-	.072	.233	1.4	4.5	2.80	.40	.50	.058
III	1.25	.083	.25	0.8	5.5	-	-	-	.084
IV	1.83	.20	.20	1.3	4.44	2.28	.40	.65	.054

sufficient to fill or nearly fill the rehearsal buffer.

(b) An object's first encoding is most likely (probability = $1-\alpha \simeq .8-.9$) to produce marginal tracking accuracy (i.e., absolute error ≃ 15-45°).

(c) There is a moderate chance (probability = $\beta \simeq .2$) on each subsequent update of improving its accuracy.

(d) An object's first rehearsal requires ≃1.4 seconds (the parameter θ).

(e) The rehearsal maintenance constraint demands that each trajectory be rehearsed at least every μ seconds, with $\mu \simeq 4.5$.

(f) After two rehearsals of an object, sufficient rehearsal time is accumulated ($\tau \simeq 2.3-2.8$ seconds) to produce an exponential decrease in its required rehearsal times on succeeding updates. Rehearsal time then decreases to an asymptote = $\varepsilon \simeq 0.4$ seconds with rate = $\nu \simeq .5-.6$.

The analysis above implies that with a sufficiently large number of updates, an unaided observer could maintain as many as 9-12 trajectories if the interval between updates is at least 5 seconds. If hardware/software aids were provided, the operator's capacity could, of course, be increased. Likewise, any demands of auxiliary tasks would result in lower performance limits due to disruption of processing.

SUMMARY AND CONCLUSIONS

1. The unaided observer can keep track of and later report the movements of about seven objects. This constraint results from perceptual and memory limitations.

2. Tracking performance improves as the time between display updates is increased up to about 13 seconds. This processing time is apparently required in order to establish and maintain suitable representations in memory.

3. Multi-object tracking performance is accounted for by a mathematical model that expresses human memory limitations in terms of encoding, learning, and rehearsal processes. Evidence supports the unconventional notion of a variable rehearsal capacity in short-term memory.

4. Analysis of human memory and information-processing limitations should be applied to complex operational tasks in order to support system designers with quantitative estimates of operator performance.

REFERENCES

Atkinson, R. C., and Shiffrin, R. M. (1968). Human memory: A proposed system and its control processes. In K. W. Spence and J. T. Spence (Eds.), The psychology of learning and motivation, Vol. 2, New York: Academic Press, 89-105.

Greitzer, F. L., Kelly, R. T., and Hershman, R. L. (June 1982). Human memory limitations in multi-object tracking. (NPRDC Tech. Rep. 82-48). San Diego: Navy Personnel Research and Development Center.

Miller, G. A. (1956). The magical number seven plus or minus two: Some limits on our capacity for processing information. Psychological Review, 63, 81-97.

Neisser, U. (1967). Cognitive psychology. New York: Appleton-Century-Crofts.

Norman, D. A. (1968). Toward a theory of memory and attention. Psychological Review, 75, 522-536.

Peterson, L. R., and Peterson, M. J. (1959). Short-term retention of individual verbal items. Journal of Experimental Psychology, 58, 193-198.

Sperling, G. (1960). The information available in brief visual presentations. Psychological Monographs, 74 (Whole no. 11).

Sperling, G. (1963). A model for visual memory tasks. Human Factors, 5, 19-31.

DETECTING A CHANGE IN TARGET LOCATION: A COMPARISON OF HUMAN AND OPTIMAL PERFORMANCE

R. L. Hershman and F. L. Greitzer

Navy Personnel Research and Development Center, San Diego, CA 92152

Abstract. Given noisy data, consider the surveillance problem of inferring the location of a target where only two states of nature are of practical concern: The target is either (1) at an arbitrary origin or (2) it is R units removed from the origin, where all directions are assumed equally likely. Standard Bayesian analysis yields an optimal expected-cost minimizing rule. For this rule, the observed data are completely summarized in a simple statistic, the distance from the origin to the center of mass of the data. Receiver operating characteristic curves were derived for this problem, and data were obtained for human decision makers in computer-driven tasks using both fixed and sequential sampling procedures. Performance with fixed sample-sizes was virtually optimal. In the sequential task, decision makers correctly adopted dynamic decision bounds for the distance statistic, but took more samples than specified by the optimal procedure.

INTRODUCTION

In our San Diego laboratory, we were looking for a task to use in the study of time-sharing skills--those in which a human operator must do two things at once. We chose for one of the tasks a "simple" idealized decision problem--deciding whether or not a target has changed location from an earlier fix.

It was easy enough to tell whether observers were doing better or worse by the proportion of their correct decisions--but we lacked an absolute standard of performance to describe the true level of their skills. Some analysis was clearly in order.

FIXED SAMPLE-SIZE PROBLEM

Consider the following inference problem for a human observer: A target is either at a given location (say, the origin) or it has moved R known units since last observed to the circumference of a circle with radius R. We assume that the target is stationary at the time the data are available; and that all directions of movement are equally likely.

As shown in Fig. 1, there are two states of nature S_0 and S_R. In the state S_0 the target is at the origin, its original location. In state S_R, the target has moved R units with direction of movement given by the angle θ. θ is a uniform random variable on the interval $[0, 2\pi]$. The problem is to decide whether the target has moved or not: i.e., whether S_0 or S_R is true.

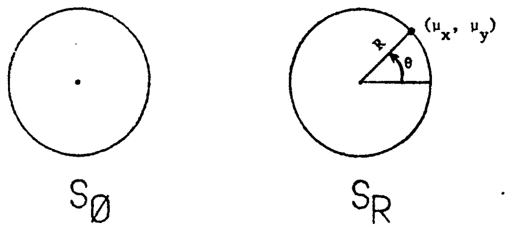


Fig. 1. Two states of nature.

We let a sample vector $\tilde{z}$ of N observations be available at decision time. Each of these has spatial coordinates (x_i, y_i). We assume also that the data generator has the circular normal density[1] given either S_0 or S_R. That is, $\tilde{z}$ is a vector of N independent and identically distributed circular normal variates, each having mean (μ_x, μ_y) and variance σ^2. If S_0 is true, the target hasn't moved, and the mean is (0,0); if S_R is true, the target has moved and the mean satisfies $\mu_x^2 + \mu_y^2 = R^2 > 0$.

The task for the decision maker is illustrated in Fig. 2. At the left is a sample of data from S_0. At the right, a sample from S_R. The decision maker sees one such display or the other and must decide whether S_0 or S_R is true.

---

[1] $f(x,y) = (2\pi\sigma_x\sigma_y)^{-1} \exp -\frac{1}{2}\left\{ \frac{(x-\mu_x)^2}{\sigma_x} + \frac{(y-\mu_y)^2}{\sigma_y} \right\}$

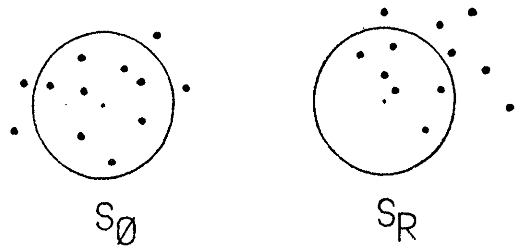


Fig. 2. Two typical displays

We want the decision maker to minimize the expected cost of decisions. The a priori probabilities of S_0 and S_R are p_0 and p_R. Costs are incurred as shown in the following cost matrix:

		S_0	S_R
Decision:	"O"	0	c_0
	"R"	c_R	0

Optimal Decision Rule

Standard Bayesian analysis leads to this expected-cost minimizing rule:

Decide "R" iff $f_R(\tilde{z})/f_0(z) > c_R p_0 / c_0 p_R$,

where $f_j(\tilde{z})$ is the probability density function of $\tilde{z}$ given state j. In other words, the observer should decide "R" (that the target has moved) if and only if the ratio of the probability densities exceeds a constant given by a simple function of the priors and costs.

The circular normal assumption gives directly

$$f_0(\tilde{z}) = \exp\left\{(\Sigma x_i^2 + \Sigma y_i^2)/2\sigma^2\right\} / (2\pi\sigma^2)^N. \tag{1}$$

For $f_R(\tilde{z})$, the target has moved to a new location $(R\cos\theta, R\sin\theta)$, where θ is a random variable. When we integrate over θ we obtain

$$f_R(\tilde{z}) = (2\pi\sigma^2)^{-N}\, I_0[(NR/\sigma^2)\sqrt{\bar{X}^2 + \bar{Y}^2}]\, \exp[(-2\theta^2)^{-1}(NR^2 + \Sigma x_i^2 + \Sigma y_i^2)], \tag{2}$$

where $\bar{X} = \Sigma x_i/N$, $\bar{Y} = \Sigma y_i/N$, and $I_0(.)$ is the modified Bessel function of the first kind and zeroth order. (Psychologists seldom get to see, let alone derive, a Bessel function, so this was an analytical orgy and a great surprise to us!)

The square-root term in Eq. (2) is clearly a distance. In fact, it is the distance from the origin to the center of mass (or centroid) of the data. If we define this distance as d, then the ratio of interest becomes

$$\frac{f_R(\tilde{z})}{f_0(\tilde{z})} = I_0(dNR/\sigma^2)\, \exp(-NR^2/2\sigma^2) = Q(d). \tag{3}$$

The only variable in this ratio is the statistic d, so we have called this ratio Q(d). The optimum rule then reduces to a test of whether $Q(d) \gtrless c_R p_0/c_0 p_R = K$. That is, the decision maker need only compute the simple statistic d, the distance to the centroid of the data, and decide whether the target has moved or not based on whether or not Q(d) exceeds K. An equivalent test is:

Decide "R" iff $d > d^*$,

where d^* is such that $Q(d^*) = K$. We took $K = 1$ (i.e., symmetric costs and priors) and computed d^* for selected values of N/σ^2 and R. The result is shown in Fig. 3. It can be seen that the critical distance d^* is a decreasing function of N/σ^2 (as we should expect) and approaches R/2 in the limit for K = 1.

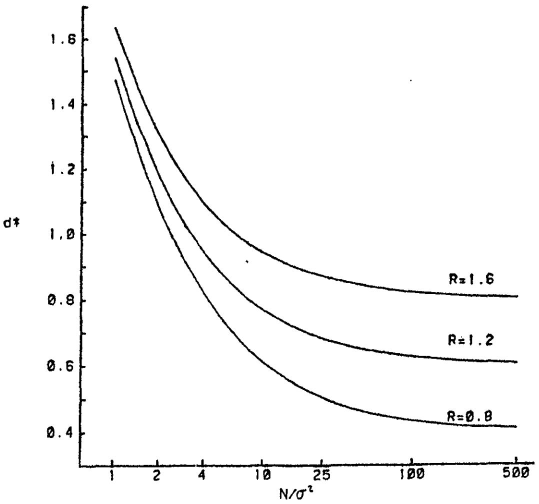


Fig. 3. Critical distance d^* as a function of R and N/σ^2.

Receiver Operating Characteristics

We next regard the movement of the target as a "signal" to be detected--and obtain the probability of detection and the probability of a false alarm. These are, respectively, the probabilities that the distance d to the centroid of the data exceeds d^* given either S_R or S_0.

Given S_0, the quantity Nd^2/σ^2 turns out to be a chi-square random variable with 2 df. And, the same quantity, Nd^2/σ^2, is a noncentral chi-square random variable with 2 df and noncentrality parameter R^2.

Figure 4 exhibits the receiver operating characteristic, which plots in the unit square the detection probability versus the false alarm probability. The curves shown are for sample sizes N = 1,3,5,7,9 with R=1.2 units and $\sigma^2=1$. For a given sample size, a decision maker who uses the distance statistic d is constrained to the given curve. The points marked "o" designate optimal performance for the case K=1, symmetric costs and priors.

AS-IF Detector

Recall that our optimal decision maker quite properly considers all directions θ of the target's possible movement. We now consider a quite different beast that forms the maximum likelihood estimate $\hat{\theta}$ arctan $(\bar{Y}/\bar{X})$ and then tests the origin against the simple alternative that the target has moved R units at the angle $\hat{\theta}$. This decision maker behaves "as if" $\hat{\theta}$ were the only possible θ. We call him an AS-IF detector.

Analyzing this detector for K=1 yields the decision rule,

Decide "R" iff d > R/2.

Referring again to Fig. 4, the digits 1, 3, 5, 7, 9 are in fact the operating points for the AS-IF detector. This detector is always on the same curve as the optimal decision maker because they both use the same test statistic d. But the AS-IF detector is more disposed to assert that the target has moved. He will always have higher detection and false alarm probabilities than is optimal to minimize expected cost.

Relation to Signal Processing

Alas, after some searching we discovered that the solution to our spatial task of detecting a change in target location is not an original one. The solution may be "new" but it is identical to that for a

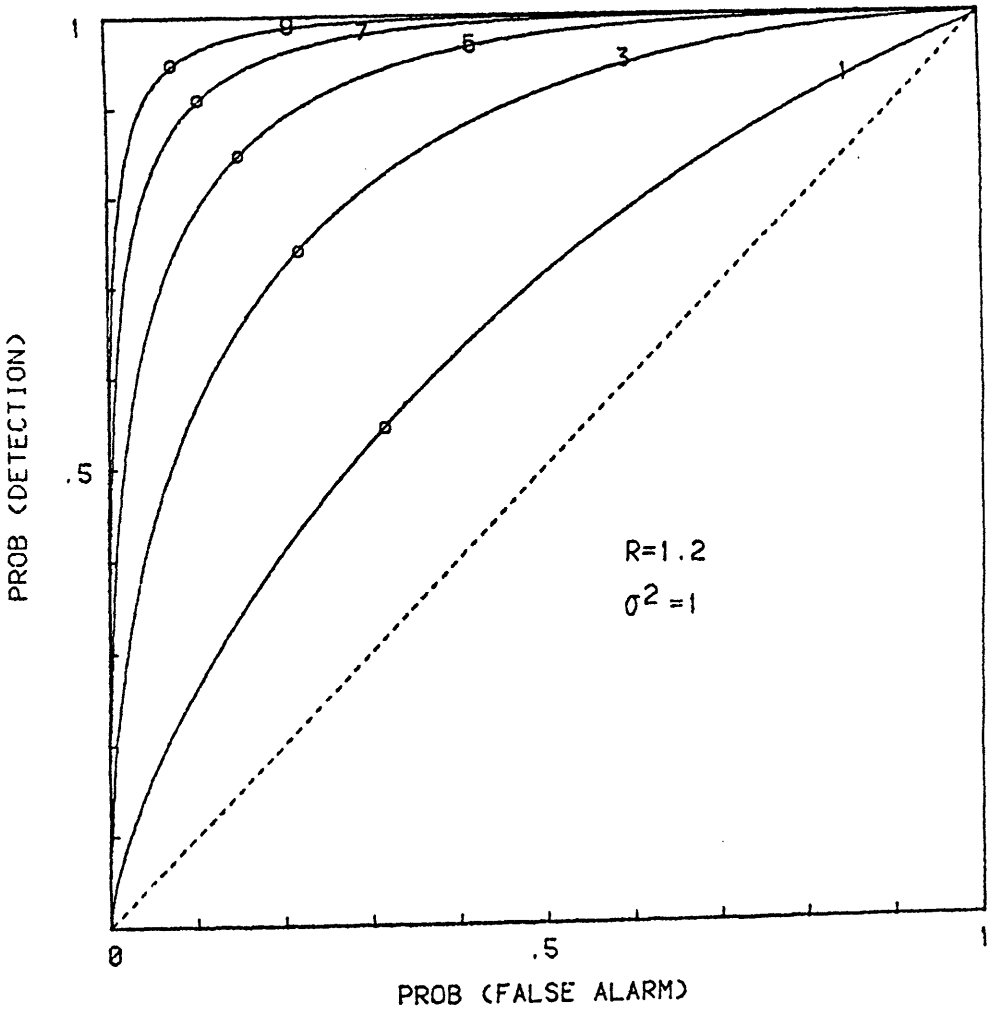


Fig. 4. ROC curves for R = 1.2, σ^2 = 1, and sample sizes N = 1, 3, 5, 7, and 9.

well-known problem in acoustic and radar signal processing--that of detecting in Gaussian noise a sinusoid signal known exactly except for phase. There is a strict isomorphism between these two problems: They are indeed the same problem and have the same solution.[2] The unknown phase of the sinusoid "is" the unknown θ in our problem. The amplitude of the envelope output by the narrow-band filter is precisely the distance statistic that arises in our target movement detection task. The Rayleigh-Rice probability density functions that arise in the signal-processing domain have as their counterparts our chi-square and noncentral chi-square probability density functions.

Human Performance

Now we present some human performance data for this task. We assigned equal prior probabilities (so that there is a 50-50 chance that the target has moved). The observer earns 10 points for each correct decision and loses 10 points for each incorrect one. Each observer was tested for approximately 600 plays at each of the sample sizes N = 1,3,5,7, and 9. A microcomputer drove a graphics terminal that always displayed the circle of radius R and plotted the N samples as points in the plane. Feedback--right or wrong--and the correct location of the target were given after each play.

Figure 5 shows the performance of three observers (denoted as X, G, and H) for the

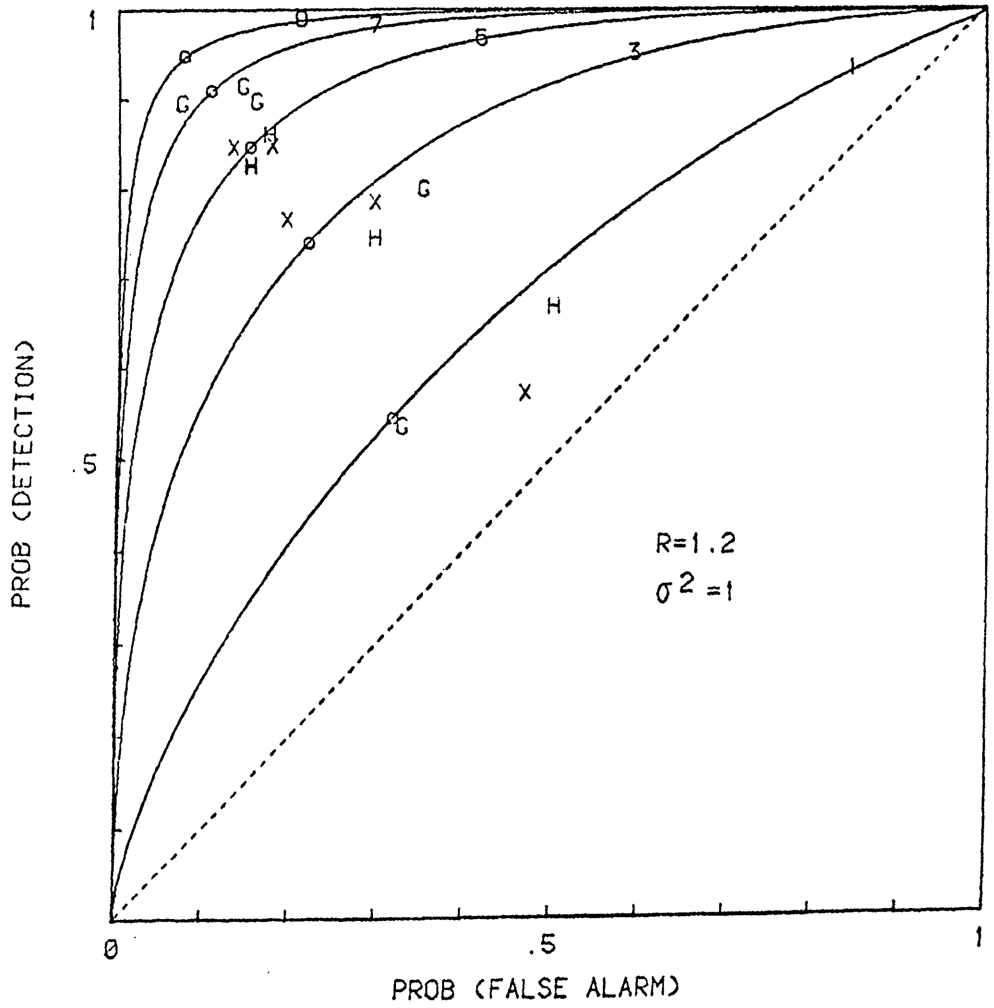


Fig. 5. Performance of three observers (X, G, H) under five sample-size conditions.

---

[2] One of the attendees at the C^3 Workshop tells us that this issue was addressed some four decades ago by Rice (1944, 1945).

last 200-300 plays under the five sample-size conditions. Note that in this figure we have not attempted to distinguish among the sample sizes; however, the data generally move toward the upper left corner as N increases.

We should now like to emphasize two characteristics of human performance in this task. First, the observers are not strictly optimal. Their performance only approximates the ideal. And although they do better as the sample size increases, it is clear that they fail to extract all the information available in the larger sample-size conditions.

Second, it is clear that the observers are not behaving like the AS-IF detector. They are using something like the distance statistic d and adjusting their criterion as the sample size increases.

Figure 6 shows this more clearly. Here the critical distance is shown as a function of N. The open circles show how d* properly decreases for the optimal detector. We estimated the d* criteria for each of the three observers by taking that distance d for which they were equally likely to respond "R" or "O". These are plotted as the three dashed curves in the figure. Note that they are "close" to the ideal d*, and we clearly must reject the AS-IF detector as a possible model for our observers.

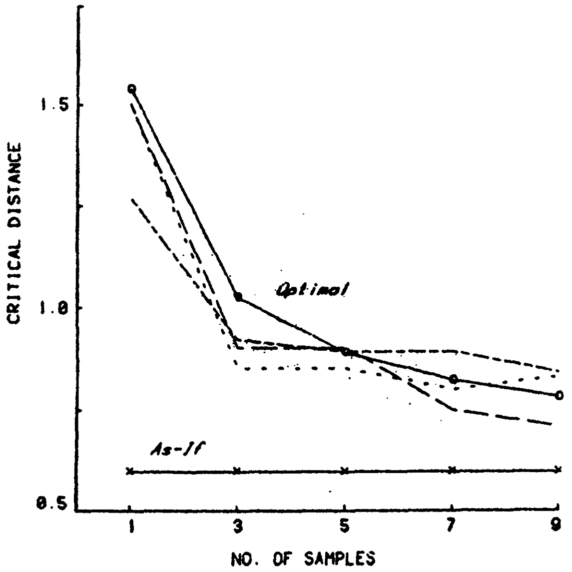


Fig. 6. Estimated d* criteria for three observers (dashed curves) as function of sample size. Optimal (o——o) and "As-If" (x——x) performance are plotted for comparison.

SEQUENTIAL DECISION PROBLEM

Up to now, we have treated the sample size N as fixed. We next consider the more interesting sequential decision problem in which the decision maker, after each observation (sample), must decide either "O" (the target is at its original locus); "R" (the target has moved); or elect to buy another sample at some cost. Correct terminal decisions are worth, say, 10 points. Incorrect terminal decisions cost 10 points. Let the price of a sample be 1 point. We again use equal priors and we want the decision maker to play the sequential game so as to maximize his expected payoff.

Now, the optimal decision maker will adopt a decision rule of the form shown in Fig. 7. For an observed distance d to the centroid that exceeds the upper curve, the decision maker decides that the target has changed location. If d is beneath the lower curve, the decision maker decides the target has not moved. For values of d between the bounds, the observer should defer and buy another sample. The bounds are, of course, dynamic and vary as a function of the number of samples already in hand.

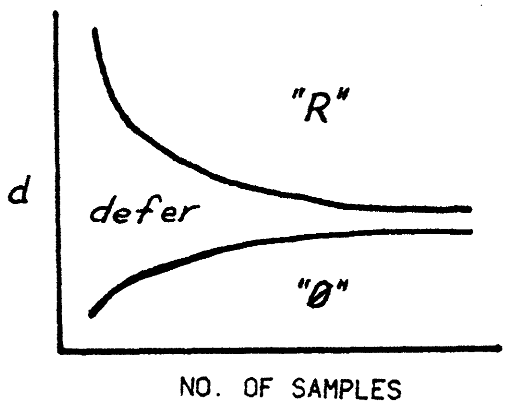


Fig. 7. Optimal decision bounds as a function of sample size in the sequential decision problem.

We used the Wald (1947) approximations for the decision boundaries that are usually given in terms of the desired error probabilities, α and β. We found the "quasi-optimal" bounds by running Monte Carlo simulations based on the Wald boundaries, and then searching the Wald space for the maximum expected payoff. Fig. 8 shows these optimal bounds (the solid lines) for the first six samples of the sequential game in which R = 1.2 and $\sigma^2 = 1$. Notice incidentally that for this game the optimal decision maker never decides that the target has not moved with less than three samples in hand. However, a single sample with a large distance from the origin (about 2.75 units) can trigger the decision that the target has changed location.

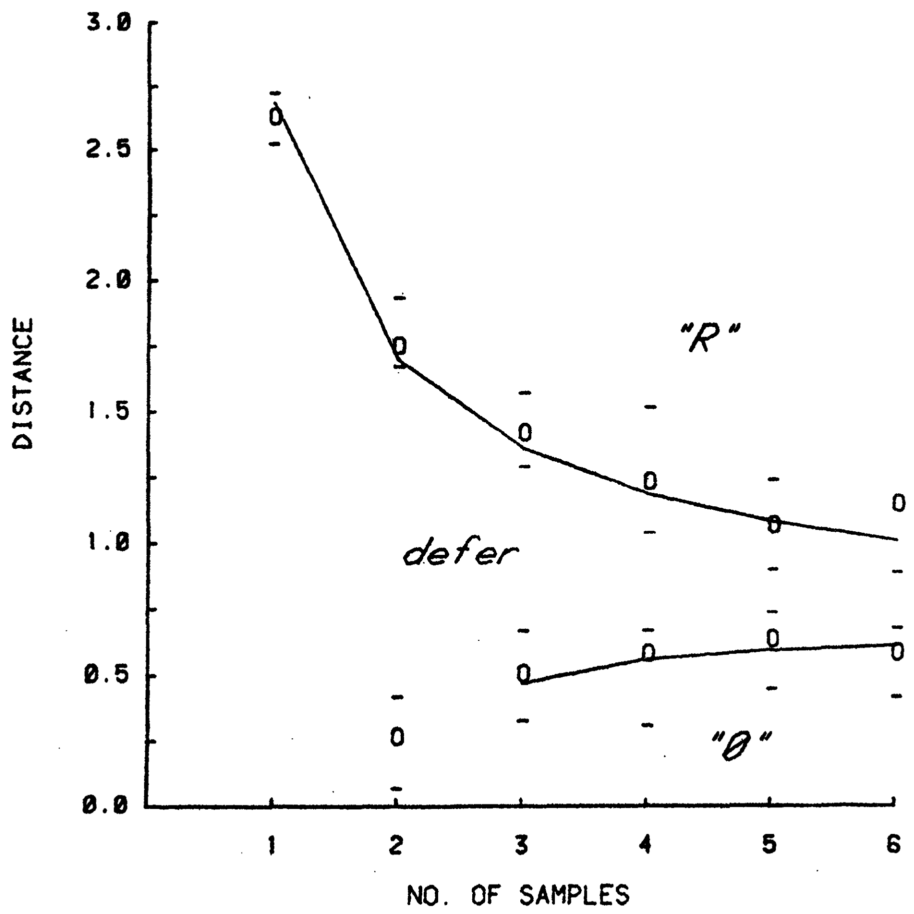


Fig. 8. Optimal decision bounds (solid curves) and estimated bounds for a human observer (O's) for the sequential game with R = 1.2, $\sigma^2 = 1$.

Also plotted in Fig. 8 are the data of a well-practiced observer for 500 plays of this sequential game. The upper O's are the observer's indifference points between the terminal decision "R" and the decision to "buy another sample." The lower O's are the indifference points between the decision "0" and "buy another sample." The dashes above and below each point indicate the extent of variability in the observer's bounds. For this observer there is surprisingly good conformity with the optimal bounds. Data from other observers also clearly follow the form of the optimal prescription.

CONCLUSIONS

As psychologists, we are particularly interested in the limits of human information processing abilities and in the sources of their nonoptimality when they arise. In real systems these are all too often ignored--and result in expensive hardware fixes, software modifications, or just plain degraded systems performance.

The preliminary data we have given here, however, speak quite well for the human as a spatial processor and decision maker in a task that is by no means straightforward.

And lo and behold--one need never have seen a Bessel function!

REFERENCES

Rice, S. O. (1944). Mathematical analysis of random noise. Bell System Technical Journal, 23, 282-332.
Rice, S. O. (1945). Mathematical analysis of random noise. Bell System Technical Journal, 24, 46-156.
Wald, A. (1947). Sequential Analysis, New York: Wiley.

MEASURING THE HUMAN FACTORS IMPACT OF COMMAND AND CONTROL DECISION AIDS

Wayne Zachary
Decision and Organizational Science Team, Analytics, Inc.
Willow Grove, Pennsylvania

Julie Hopson
Human Engineer Division, Naval Air Development Center,
Warminster, Pennsylvania

Abstract. Decision aiding in C^2 requires a new type of cognitive Human Factors Engineering which seeks to blend the mental requirements/capabilities of the human operator with the characteristics of computational decision support subsystems. Methodologies for solving four such cognitive human factors problems are presented, using examples from the domain of Air Antisubmarine Warfare. Particular attention is paid to difficulties in measuring the impact of candidate aids on the operator and overall C^2 system.

Recent years have seen a dramatic increase in technological sophistication of Command and Control (C^2) operations. New sensor and communication technologies have substantially increased the volume and rate of flow of information within C^2 systems. The result of this trend has been a growing burden on the human operators of C^2 systems, as they are forced to make an ever greater number of decisions, each with less room for error and less time available than ever before. One solution to this technology-induced dilemma has been to apply the same technology that created it towards eliminating it. Subsystems to augment the decision-making performance of C^2 personnel, often referred to as "decision aids," have been advocated for some time (Shrenk, 1969) and are being implemented in C^2 environments with increasing frequency.

Unfortunately, however sound the fundamental idea of computational decision support for C^2 is, its realization to date has been less than a panacea (Andriole and Hoppel, 1982). Substantial problems have emerged in "fitting" the aid to the needs and capabilities of both the human operator and the C^2 system in which the aid is embedded. This has in turn given impetus to the creation of a new sort of Human Factors Engineering, which strives to mesh not just the physical capabilities of the human with the machine environment (as does more traditional Human Factors) but to integrate the cognitive characteristics of the human component into the design/evaluation process as well.

This paper summarizes several years of research in this new domain of cognitive human factors, as applied to the problem of designing effective decision aids for command and control environments. The emphasis is on providing answers to four primary questions:

1. What is the level of"grain" at which decision making should be aided?
2. Given some level of grain, what are the most important decisions to aid?
3. Are these most important decisions amenable to improvement by some (or any) candidate decision aid?
4. What is the impact of a decision aid or suite of decision aids on operator cognitive and manual workload?

Rather than exploring in the abstract, these issues were examined in the context of a specific operational C^2 environment -- Naval Air Anti-Submarine Warfare (ASW) -- for three reasons. First, it is both strategically and tactically important, and is thus itself a primary application domain for decision aiding. Second, it is one of the most technologically advanced C^2 environments, and exhibits many problems of operator information-processing and decision-making overload. And third, ASW systems are self-contained and easily bounded, thus permitting them to be studied in a straight-forward manner.

SELECTING THE UNIT OF COGNITIVE ACTIVITY TO AID

"Decision making" is a cognitive activity and therefore non-observable in the strict sense. It is also, in some measure, an analytic construct -- a category used to describe behavior rather than a category arising directly from behavioral phenomena The conclusion of these two observations is that in most real-world or "natural" contexts (such as C^2), it is difficult to bound precisely where each decision involved begins and ends. To complicate things further, any given decision can usually be broken into a number of constituent decisions at some lower level of abstraction or "grain", so that decision making appears to be an infinitely recursive activity. In building a decision aid for a command and control decision maker, it is important to realize that the designer must settle on a definition (or at least a boundary) of the decision(s) to be aided. Defining the decision in too broad or abstract a manner can leave the human "up in the air," with too much detail to fill in before a meaningful set of actions is generated; defining the decision in a too-restrictive or fine-grained level can over-structure the decision process, and add to rather than detract from information processing workload. Moreover, the "difficult" aspects of decision-making are not always the same; in some cases it is the high-level aspects that are most difficult while in other cases it is the low-level detailed aspects. Thus, selecting the level of "grain" of decision making to be aided is the first important question in decision aid design. It is in a real sense this first question that is the most important question as well, for if the wrong level of grain is established, the resulting aid will probably not be useful, no matter how well it is designed and implemented.

The initial investigations into the decision-making activities of ASW C^2 environment showed that the precise roles which decision aids should play in Naval Air ASW were unclear. There already existed a large number of software aids in the aircrafts' computers for the processing of specific data, and there were numerous tactical aiding programs available for use on operators' hand-held programmable calculators. There were, in fact, so many "micro-aids" of various kinds that their use and coordination posed a substantial problem in itself. Thus, a need for higher-level aids was suggested.

A detailed analysis of the various decisions made by the Tactical Coordinator or TACCO (the senior C^2 crewmember in the Air ASW environment, and the focus of this research) showed that C^2 decision making in this arena was primarily hierarchical in structure.

A generic mission structure analysis decisions throughout the Air ASW mission but toward different ends, depending on the particular objective or goal event of the current mission phase. Sonobuoy (acoustic sensor) placement decisions, for example, are required in all phases of the mission. However, the TACCO uses one set of criteria in deciding on sonobuoy placements for that segment of the mission where the goal is merely gaining contact with a hostile submarine, and another set of criteria in deciding on sonobuoy placement for the segment where the goal is to determine the submarine's precise location so that an attack may be launched.

The differing goal events of the different mission phases were found to give rise to complex decision-making contexts which constrain the way in which the TACCO's primary functions were carried out. These contexts were termed <u>decision-making situations</u> and assigned to the highest level of a structural hierarchy of decision-making. Underlying the decision situations is a shared set of <u>decision functions</u> which are combined <u>in differing ways in each decision situation</u>. Decision functions are assigned to the next highest level of the hierarchy. The lowest level considered in the hierarchy was that of a decision task, a well-defined choice requiring minimal cognitive resources. An example of such a minimal decision task is deciding which computer program to enable in order to initiate a given data-processing action.

The hierarchy of grain-levels in decision-making was used to form a classification of decision aids, based on the level of grain in decision making on which the aid focused. Given the difficulties observed in coordinating decision functions to achieve the principal goal events, it was felt that the decision situation was the primary unit to which decision aiding should be applied in ASW C^2. A detailed discussion of research supporting this conclusion is provided in Zachary (1980a). Six major decision situations and associated "goal events" were defined for Air ASW C^2 as follow:

1. <u>On-Station Search</u> -- where the goal is <u>obtaining a sensor</u> contact with a hostile submarine,

2. <u>Contact Classification/Verification</u> -- <u>where the goal is identifying as precisely</u> as possible the source of a sensor contact,

3. <u>Localization</u> -- where the goal is <u>reducing the uncertainty</u> in the location, course, speed, and depth of the target to the point where a target track may be maintained,

4. <u>Surveillance Tracking</u> -- where the <u>goal is a continuous maintenance</u> of a target track,

5. <u>Attack Planning</u> -- where the goal is <u>placement of an optimal</u> attack on the hostile submarine, and

6. <u>Lost Contact Reacquisition</u> -- where <u>the goal is re-establishment</u> of sensor contact with a lost target.

While it proved possible to develop methods for quantitatively answering the remaining three questions addressed in this reearch, it seemed necesary to deal with the question of grain-level in a more intuitive manner. It appears that before any more precise method can be developed for determining the "right" decision to aid in a given C^2 environment, a new kind of "cognitive calculus" is required. This cognitive calculus is needed to guide the mapping of the various hierarchical levels of decision-making in a given context onto the cognitive resources of the humans involved. This would permit direct determination of which parts of the decision-making environment are hard (and require aiding) and which parts are easy.

Having defined a level of abstraction (the decision situation) of decision-to-be-aided at that level of abstraction, attention was next turned to the complementary issues of estimating the priority and aidability of decision situations. Before development and implementation of a decision aid can be fully warranted, it is necessary to demonstrate that the decision being aided is tactically important (i.e., high priority) in the mission where it arises and that the decision is capable of being aided (i.e., is "aidable") by some candidate decision aid for it.

PRIORITIZING DECISIONS FOR AID DEVELOPMENT

The need to prioritize decisions for further consideration arises from two sources. First, whatever level of grain is chosen to partition the set of decisions made in a given C^2 environment affects the design of a decision aid since not every decision at that level will have an equal impact on the achievement of various mission objectives. Some will be more operationally important, others less. The second is that given the reality of limited financial, human, and computational resources, it is clear that a decision aid can not be provided for every "deserving" decision. Therefore, some method must be established for determining the operational importance of each decision and for identifying from this operational importance the relative priority among all the decisions for decision aid development.

A preliminary examination of this prioritization problem (see Zachary [1980a: Section 6]) pointed out the need to treat priority as a multidimensional construct and to incorporate the knowledge, experience, and judgments of domain experts -- experienced operational personnel -- into the prioritization procedure. To this end, a method called Priority Mapping was developed and applied to the six Naval Air ASW decision situations identified previously. Using the psychometric techniques of multidimensional scaling and unfolding analysis, this method translates nonquantitative judgments elicited from experienced operational personnel about the similarity among and ranked importance of specific decisions into numerical priority scores for those decisions.

Priority Mapping has three important characteristics. First, it uses the intuition, experience, and knowledge of experienced personnel as the basis for determining situational priority for decision aiding. Second, it uses psychometrically reliable data in the form of nonnumerical judgments about decision similarity and simple rank orderings. And third, it produces precise quantitative importance values or priority scores for each decision considered.

In Priority Mapping, judgmental data on the perceived similarities among a set of decisions are collected using a pychometric task known as unconstrained sorting (see Miller [1969]). The resulting data are then preprocessed with a computer program entitled METRIC that calculates a measure of the pairwise dissimilarity of the set of decisions for which data was collected. This measure is the Burton (1975) F* measure. Multidimensional scaling is then applied to this dissimilarity measure to uncover the principles or dimensions that interrelate the decisions considered. In MDS, the decisions are represented in a multidimensional space -- the precise number of dimensions in the space must be determined as part of the "solution" -- with each dimensional axis representing a fundamental feature or principle which interrelates the decisions.

Multidimensional Unfolding Analysis (see Bennett and Hays [1960]), is then applied to the data on the ranked importance of the decisions and the MDS solution to determine the mathematical form of the implicit priority functions used by the domain experts to rank the decisions. Unfolding analysis works by seeking a "reference point" in the multidimensional space and a distance metric (formula for computing interpoint distances) such that the order of the distances of the decisions in the space from the reference point replicates the rank orderings (by importance) given in the raw data. When such a reference point and metric are found, the distances of the decisions from the reference point give the decisions' numerical priority scores. The overall prioritization methodology, as applied to Naval Air ASW, is summarized in Figure 1.

From an empirical perspective, it was found that decision functions were a better unit than situations on which to

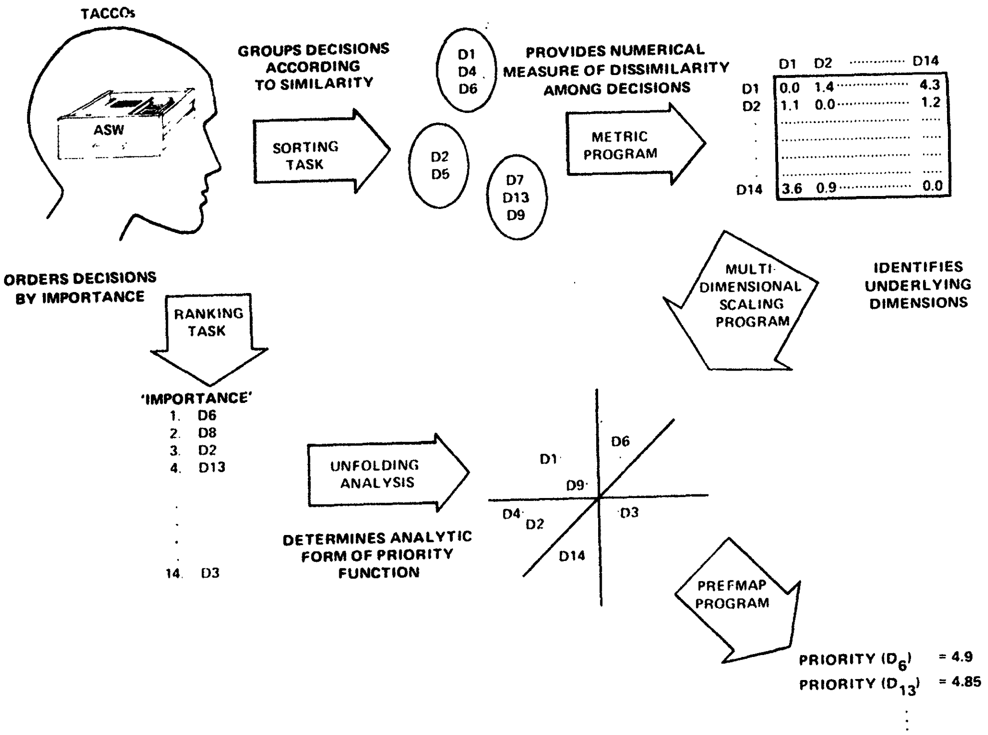


Figure 1 **Priority Mapping Procedure for Prioritizing Naval Air ASW Decisions**

focus the Priority Mapping data collection. This is because decision functions are more concrete constructs than the more abstract decision situations, and as a result subjects appear more able to provide meaningful judgments (i.e., psychometric data) about them. Thus, when the unit of interest is the higher level decision situation, it is necessary to map out the relationship between the situations and their constituent functions so that priority scores assigned to functions can subsequently be combined to create priority scores for the decision situations.

In the application of Priority Mapping to the Air ASW domain, a sample of 14 decision functions was selected. The functions in the sample were selected so as to include at least some functions from each decision situation, and as many functions that contributed to multiple situations as possible. Data for the Priority Mapping procedure were collected from a total of 52 highly experienced P-3C TACCOs at NAS Moffett Field and NAS Jacksonville, and 31 highly experienced S-3A TACCOs at NAS San Diego.[1] The full Priority Mapping method was then applied to this data to generate priority scores for the six decision situations listed above.

The MDS analysis of the basic unconstrained sorting data produced a three dimensional solution. The three dimensions were interpreted as follows:

- Dimension One: Input Uncertainty -- the amount of uncertainty typically present in the information available as input to the decision function.
- Dimension Two: Information Processing Load -- the amount of (mental) information processing typically required by the decision function.
- Dimension Three: Complexity of Alternative Structure -- the number of and interrelationships among the alternatives typically available to the decision maker in the decision function.

A more detailed discussion of the MDS analysis can be found in Zachary (1981b) and Bennett, Goodson, and Zachary (1982).

The Unfolding Analysis performed with the three dimension MDS solution and the ranking data on perceived importance resulted in a relatively complex quadratic solution (technically, a generalized

---

[1] The P-3C and S-3A are the two primary type of Naval Air ASW platforms.

distance model solution; see Carroll and Chang [1972]). The formula generated by this solution was used to calculate priority scores for the 14 decision functions in the sample. Priority scores were generated for the decision situations by summing the scores of the decision functions in each decision situation and then normalizing to account for the fact that a different number of functions were included in the sample for each situation. Two separate prioritizations, shown in Table 1, were created (from the sets of raw ranking data), one with regard to surveillance missions, and another with regard to attack missions, because the subjects could not perform the basic psychometric tasks without explicit reference to one of these two specific missions. The full result of the Priority Mapping analysis are given in Zachary (1980b); a replication is discussed in Cagle (1980).

Table 1 ASW Decision Situation Prioritizations in Two Types of ASW Missions

RANK IN MISSION WITH ATTACK OBJECTIVE	DECISION SITUATION	DECISION SITUATION	RANK IN MISSION SURVEILLANCE OBJECTIVE
1	CONTACT CLASSIFICATION/ VERIFICATION	CONTACT CLASSIFICATION/ VERIFICATION	1
2	ATTACK PLANNING	SURVEILLANCE TRACKING	2
3	LOCALIZATION	LOST CONTACT REACQUISITION	3
4	SURVEILLANCE TRACKING	LOCALIZATION	4
5	LOST CONTACT REACQUISITION	ON-STATION SEARCH	5
6	ON-STATION SEARCH	ATTACK PLANNING	6

NECESSARY AND SUFFICIENT CONDITIONS FOR DECISION AID DEVELOPMENT

Before the process of decision aid design can ever begin, it is necessary to provide answers to the two questions considered above. A level of grain in decision-making must be established, and the various decisions at that grain level must be identified and prioritized. Priority Mapping provides a way of uncovering the relative priority of the decisions under consideration and, on the surface, it would seem that the development of decision aids should begin directly for those situations showing the highest operational priority. But although high priority is necessary for aid development, it is not by itself sufficient. For construction of a decision aid to be fully justified, it must be determined that human performance in that decision is actually amenable to augmentation by means of some candidate decision aid. In short, it has to be proven that the decision is "aidable." The demonstration of aidability can thus be thought of as a sufficiency condition which complements the necessity condition of priority.

The aidability of a decision can not be considered without reference to some candidate decision aid, but as shown below, the candidate aid can be represented as only a functional concept, i.e. a specification of a set of aiding functions to be provided to the operator. There are two ways in which human performance in a given decision can be augmented by some candidate aid. The first way, which can be termed direct augmentation, is via an increase in the quality of decision making, as evidenced by gains in mission achievement. The implicit premise underlying decision aiding is that the quality of decisions made during a mission directly affect the outcome of the mission and that consequently any improvement in decision quality will yield a concomitant improvement in the level of mission achievement.

A sceond way in which a candidate aid can augment performance is by reducing the overall workload of the human decision maker. In virtually all C^2 environments the human operator is performing a variety of functions more or less concurrently, and thus must make multiple decisions and interact with multiple hardware/software subsystems simultaneously. Although clear data on human performance in simultaneous complex tasks are lacking, it is intuitively clear that in multitask envornments the reduction of workload (either cognitive, manual, or both) on one task can result in increased performance on others because of the time and attention resources this frees for them. If a decision aid can simplify operator's procedures in some of the decisions in the C^2 environment, it will provide him with more manual and cognitive resources for others and improve his performance on these others, even though they are not explicitly addressed by the decision aid. Thus, by reduction of a workload a candidate aid may augment decision performance indirectly.

Separate methods are required to answer the separate questions of whether the decision itself is (directly) aidable and whether there is a favorable impact on workload. These methods are presented in the following sections.

ASSESSING DECISION AIDABILITY

Potential increases in mission achievement are estimated in a three-step procedure. First, the tactical contingencies which affect the quality of decision making (e.g., the accuracy of intelligence, mechanical/electronics system failures, unanticipated enemy actions) are identified for the decision situation and C^2 context addressed by some candidate aid for that situation. Second, these contingencies are factorially combined, in the context of a typical mission scenario, to produce a scenario tree. This is a scenario which can evolve in different ways according to the various combinations of contingent events that may occur. Third, a mathmatical or computer model of mission achievement is used to calculate the "best" or optimal decision for each combination of contingencies in the scenario tree. These optimal decisions are assumed to represent the decision making in the "best of all possible worlds" -- that is, using perfect rational strategies, without any stress, and with all time and computational limitations removed. The optimal decisions then are compared to the decision that would be made in the same circumstances using current unaided (baseline) procedures. The comparison is made simply by taking the ratio of optimal to baseline performance for each unique scenario evolution in the scenario tree.[2] Each ratio thus represents the "room for improvement" in unaided decision-making, and indicates the maximum possible increase in mission achievement that could be realized from a decision aid for that given scenario evolution. The aided/unaided ratio for each particular set of contingencies (i.e., scenario evolution) is "weighted" according to the probability that a given operational scenario would possess characteristics that represent those contingencies. The weighted ratios are then summed across all scenarios. This calculation produces an expected maximal increase in mission achievement.

The practical use of this quantity is in determining whether an aiding concept for a given decision has sufficient potential to justify further development, or in determining which among several candidate aids for a given situation holds the most promise. It is also possible that the decision under examination is relatively insensitive to any candidate aid, and will show no potential for improvement in any case. When this occurs no aid development is warranted, no matter how important the decision is.

ASSESSING THE WORKLOAD IMPACTS OF CANDIDATE AIDS

A second aspect of the "aidability" question concerns operator workload. If there is a hidden pitfall in decision aiding, it is in the area of operator workload. As indicated earlier in this paper, a prime motivation for introducing C^2 decision aids is the current excessive workload levels of C^2 system operators. Yet in one sense a decision aid is only one more device/subsystem with which the operator must contend. Thus there is a very real possibility that without *a priori* attention to overall workload, a decision aid might actually add to rather than alleviate the workload problems. Because of this, it is essential to factor the impact on operator workload into the decision aid design process. Moreover, workload should be considered both in the local sense of workload associated with the specific decision(s) being aided and the global sense of the other activities in which the operator is simultaneously engaged.

Methodologically, it is desirable for workload impacts to be assessed in a manner structurally similar to that in which mission achievement impacts are assessed, so that some degree of comparability is maintained. In particular, the same breakdown of key mission contingencies and range of resulting scenarios should be used in both cases so that workload and mission achievement assessment are made with regard to the same standard. To accomplish this, a three step methodology was developed for assesing changes in operator workload associated with introduction of some candidate decision aid; this methodology is analogous to the three-step methodology for assessing changes in mision achievement process.

First, current (i.e., unaided) procedures undertaken by the C^2 operator (in this example, the TACCO) in making the decision(s) to be aided are identified, along with all exogenous but concurrent decisions and actions. These decisions and actions are next described and formally represented in a task-analytic

---

[2] In order to be able to compare aided and unaided decision quality in this manner, it is necessary to have a ratio-scale measure which relates the characteristics of both aided and unaided decisions to overall mission achievement. Such a measure of mission achievement will normally have to be constructed for each application of this method. An example of the measure construction process is given in the appendix.

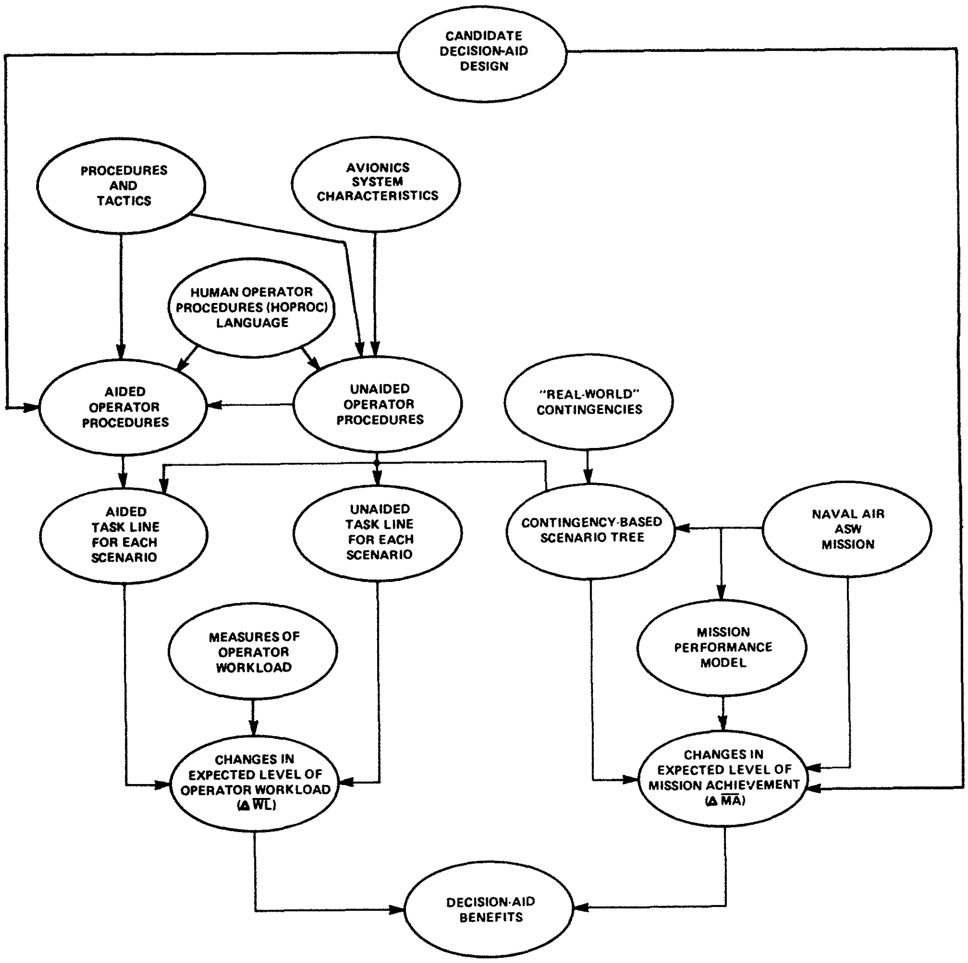


Figure 2. Estimating Benefits of Candidate Decision Aids

language called HOPROC.[3] A corresponding identification, description, and formalization are then undertaken for the procedures the TACCO would use if the candidate aid were available. Second, using the same scenario tree as employed previously, a timeline of operator procedures is developed for both the aided and baseline conditions in each unique scenario evolution. Third, subjective measures for estimating all key aspects of workload (both mental and physical) are independently applied to each timeline and combined to produce a single workload estimate.

The resulting aided and baseline workload estimates are then compared for each scenario evolution and combined across the scenario tree in the same way as mission-achievement measures. Unlike the mission-achievement assessment which yields a maximum level of change (increase), this workload assessment method yields a measure of the average level of change (decrease) in operator of the candidate aid. The entire aidability assessment methodology, showing both the decision-quality and the workload impact steps, is pictured in Figure 2.

It should be noted that the measurement of workload has long been a central concern in human factors research, and a variety of approaches to workload assessment have been developed (see Weirwille and Williges [1979]). The growing concern with cognitive processes in man-machine interactions has broadened this long debate to include the measurement of mental workload as separate issue (see Moray [1982]). In the workload asessment method shown in Figure 2, workload is measured on 14 individual (but not necessarily independent) scales, falling into four primary

---

[3] HOPROC is a English-like formal language developed by Analytics and the Naval Air Development Center to model complex task sequences performed by human operators in airborne platforms (see Lane, Strieb, Glenn, and Wherry [1980]). It was designed to become a standard formalism for man-machine interactions, and was used here for that reason.

categories: cognitive workload, psychomotor workload, motor workload, and interactional workload. All but two of the 14 measures are five-point Likert scales, whose value are assigned by a panel of raters consisting of both human factors and operational domain experts, using explicit rating criteria.

A statistically derived formula for combining these 14 measures was independently generated using empirical data on perceived workload and a general linear statistical model. More details on the workload measures and measure-combination rule can be found in Zachary (1980b) and Bennett, Goodson, and Zachary (1982). Unlike the mission-achievement measures of overall effectivenes, which are problem-unique and constructed for each decision considered, the overall workload measure combination function is generally applicable to all decision situations, at least in theory. In practice however, it is difficult to construct a combination function which exhibits all the desired properties especially the interactions among, and non-linear effects of the various constituent workload measures. The overall workload measure is therefore likely to be an evolving construct for some time. This is discussed further in the conclusion.

Aidability Analysis for Two Decision Situations

Based upon the results of the Priority Mapping analysis, decision aid design concepts were generated for two decision situations, one of high priority --Attack Planning -- and one of moderate priority -- On-Station Search. The full benefit-assessment methodology (Figure 2) was then applied to these two situation/candidate-aid pairs. The candidate aid for the On-Station Search situation (the Sonobuoy Pattern Planning Decision Aid) was found induce a moderate decrease in operator workload around 18%. This decision situation encompasses a portion of the mission where TACCO workload is currently relatively low. Although the candidate decision aid simplifies certain cognitive tasks now performed by the TACCO and eliminated additional other functions, it also empowers the TACCO to make some C^2 decisions which he is presently unable to make. Thus, it both aids to and diminishes workload, albeit in different ways. Still, the overall picture is one of a reduction of workload.

Mission achievement results were more striking, and indicated that decision-making in the On-Station Search decision situation is highly aidable. Within the context of initial sensor pattern selection and deployment, a potential increase of 87% in mission achievement was calculated. This clearly indicated that by placing the choice of this initial search pattern in the hands of the TACCO assisted by a decision aid, mission achievement in this decision situation could be dramatically improved.

The aidability assessments for a candidate decision aid for the Attack Planning decision situation also produced striking results. A substantial potential increase in decision quality was suggested, as evidenced by a calculated room for improvement of more than 200% in aided Attack Planning mission achievement. The improvement in mission achievement was realized by increases in the final probability of kill, and reduction of the time required for final prosecution, both passive and active.
The workload assessment for the candidate aid also indicated that a substantial increase in operator workload -- approaching 50% --is possible with the candidate aid. This result is especially important, since the previous analysis (Zachary [1980b]) indicated that the Attack Planning decision situation carries the heaviest TACCO workload in the Air ASW C^2 environment.

CONCLUSION

There is every indication that the current trends toward increasing speed and complexity of C^2 systems will continue and even accelerate in the future. It is therefore imperative that a firm grasp of the human issues involved in C^2 automation be gained as soon as possible. This paper has summarized several years of research in that direction, and has presented some quantitative methods developed during that period for measuring the human factors impact of decision aiding in command and control systems. In many regards this paper should be viewed as a progress report, for although many issues have been successfully resolved, many more remain and further research is currently underway to build upon the results presented above.

One important current concern is replication. While all the methodologies described here were designed to be of general applicability, only Priority Mapping has had any additional application (Cagle, 1980). Efforts are currently under way to replicate the aidability and workload assessment analyses in new C^2 domains to obtain further indications of their generality.

Another, related, issue is validation/verification. The aidability and workload assessment methods provide estimates of the impact of a decision aid on a C^2 environment, but the use of these estimates in the design process depends on knowing how accurate they are, and whether they contain any systematic biases. To this end, experimental studies are now underway to measure empirically the increases in decision quality and decreases in workload that result from the two candidate decision aids used in the initial aidability/workload assessments.

These experimental results will be carefully compared to the analytically derived estimates, and initial "benchmark" figures will be calculated on the accuracy of the aidability/workload assessment methods.

Another area of current research also concerns the workload assessment method. Since the function used to combine the fourteen individual measures of workload is intended to be of general utility, substantial work remains to be done to increase its generality. Specifically, efforts are underway to develop combination schemes that reflect some of the complex interactions among measures, and some of the non-linearities in their contributions to overall workload.

If there is any conclusion to be drawn from this research, it is simply that decision aiding may represent a significant turning point in the design of man-machine systems. In the past, the design of systems (including C^2 systems) has been primarily driven by hardware capabilities and opportunism. If it was possible to provide the human with a piece of information then the system was designed to do so -- information was good and more information was better. The human has thus been used in the residual category in every system, that is, doing what the machines can't. The operator fills in the gaps, collapses the information, and makes the decisions. It is now clear that such a strategy is no longer optimal, or perhaps even viable. From this point foreward, C^2 systems must be designed so that the people who <u>must</u> operate them <u>can</u> operate them, and <u>this</u> means that <u>design</u> must now proceed from the perspective of human limitations, not machine capabilities.

APPENDIX

Mission Achievement Measure for the Attack Planning Decision Situation

Every application of the methodology in Figure 2 requires construction of a new measure to relate decision quality to mission achievement. Each such overall mission achievement measure must combine the various indicators of decision quality produced by the specific mission achievement model being used. The measure must also have the necessary mathematical properties and exhibit the intuitive properties of mision achievement in the decision situation being considered. This appendix provides an example of the process by which a mission achievement measure is constructed, using the case of the Attack Planning decision situation from Naval Air ASW. Additional details can be found in Section 5 of Zachary (1981).

The mission achievement model constructed to examine the Attack Planning provides four basic outcomes or indicators of attack planning decision quality. The first (and by far the most important) is the probability of killing the submarine, denoted P_k. The second is the time, denoted T, expended between the start of attack planning and the placing of an attack on the submarine. The third is the number of passive sonobuoys utilized in the attack planning process, denoted S. The fourth is the number of minutes spent in active prosecution of the submarine, denoted A. These last three measures allow consideration of some of the more subtle aspects of mission achievement.

It is obviously important for the decision aid to increase the expected value of P_k. Beyond this, however, it is also desirable for the aid to help the TACCO place an attack more quickly (reduce T), utilize fewer resources (reduce S), and minimize the duration of active prosecution in which the submarine is alerted and undertakes evasive action (reduce A).

The assesment of the room for improvement in Attack Planning decision quality required the construction of mission achievement function or rule which combined these four indicators into a single measure of Attack Planning success. The first step was determining the specific properties which the rule must exhibit. From a mathematical perspective, its only required property was that it provide a true ratio measurement scale for mission achivement, so that the ratios of aided to unaided mission achievement can be legitimately calculated.

From an intuitive perspective however, four separate properties desired in the combination rule. The first was that the overall measurement be dominated by P_k. No combination of other factors should be able to compensate for the failure to kill the submarine, just as no combination of other factors should be able to diminish the desirability of killing it. The second required property was that the effects of T and S should be exhibited primarily at the margin. That is, for a given P_k the effect of the total time (T) to attack and the number of sonobuoys (S) used should become important only as they near their limiting values. Thus, if R is the remaining on-station time when Attack Planning begins, then T should become an important factor only when it nears R. Similarly, if I is the total passive sonobuoy inventory when Attack Planning begins, then S should become an important factor only as it nears I. A should have a similar effect, but since there is no maximum possible active time, its effect should merely become more pronounced as A increases in value.

The third property is that, other things being equal, the value of T should be more important than the value of S. This is because there are other sensors which could be used to continue the mission after all sonobuoys are gone, but once the on-station time has expired there is no way to compensate for it. The fourth property is that the combination rule should allow some measurable mission achievement even when P_k is 0 (i.e., when the torpedo completely misses the submarine or when an attack is not placed).

The combination rule which possesses all these properties is given by:

$$MA = \sqrt{3/(3+A)} \cdot ((\ln 100P_k)+1) \cdot (1-(T/(R+1))^2) \cdot (1-(S/(I+1))^4)$$

The first term on the right allows A to "discount" the the remainder of the expression in a way which is minor when A is near 0 but which increases as A gets large. The constant of 3 was empirically selected as that value for which an active period of more than nine minutes has the effect of reducing the value of the overall expression by 50 percent. This time of nine minutes was felt to be the expected median active prosecution period.

The second term clearly dominates the entire formula, as desired, and causes the overall MA to increase monotonically with P_k, while providing added emphasis to gains in the 0-.6 range. This reflects a desire to increase P_k into the region where there is a sizable likelihood that the submarine will actually be destroyed. The third and fourth terms are also "discounting" terms, based on T and S. As T and S each become closer to their physical limits (R and I respectively), the value of the term where they appear has an increasingly pronounced effect. The exponents limit the effect of these terms to the cases when the limits are very nearly reached. The difference in exponents reflects giving more weight to T than to S.

REFERENCES

Andriole, S. and G. Hopple (1982). They're only human: decision makers in command and control. *Signal*, 36, 45-48.

Bennett, J., J. Goodson, and W. Zachary, (1982). *Multidimensional Scaling and Unfolding Analyses of TACCO Workload in P-3C and S-3A ASW Missions*. Analytics Technical Report 1400.24A, Analytics: Willow Grove, PA

Bennett, J.F. and W.L. Hays (1960). Multidimensional unfolding; determining the dimensionality of ranked preference data. *Psychometrika*, 25, 27-43.

Burton, M. (1975). Distance measures for unconstrained sorting data. *Multivariate Behavioral Research*, 10, 409-24.

Cagle, C.M. (1980). *An Application of Multidimensional Scaling to the Prioritization of Decision Aids in the S-3A*. Unpublished Masters Thesis, Naval Postgraduate School: Monterey, CA.

Carroll, J.D. and J.J. Chang (1972). Individual differences and multidimensional scaling. In R. Sheppard, K. Romney, and S. Nerlove, (Eds.), *Multidimensional Scaling Theory and Applications. Vol I: Theory*.

Lane, N., M.I. Streib, F. Glenn, and R.J. Wherry, Jr. (1981). The human operator simulator--an overview. In J. Moraal and K.F. Kraiss, (Eds.) *Manned Systems Design: Method, Equipment, and Applications*, New York: Plenum. pp. 121-52

Miller, G. (1969). A psychological method to investigate verbal concepts. In *Journal of Mathematical Psychology*, G, 169-91.

Moray, N. (1982). Subjective mental workload. *Human Factors*, 24(1), 25-40.

Schrenk, L.P. (1969). Aiding the decision-maker: a decision process model. *Ergonomics*, 12 (4), 443-57.

Wierwille, W.W. and R.C. Williges (1978). *Survey and Analysis of Operator Workload Assessment Techniques*, Systematics Technical Report S-78-101. Systematics: Blacksburg, VA.

Zachary, W. (1980a). *Decision Aids for Naval Air ASW*. Analytics Technical Report 1366A. Analytics: Willow Grove, PA.

Zachary, W. (1980b). *Application of Multidimensional Scaling to Decision Situation Prioritization and Decision Aid Design*. Analytic Technical Report 1366B. Analytics: Willow Grove, PA.

Zachary, W. (1981). *Cost Benefit Assessments of Candidate Deciion Aids for Naval Air ASW*. Analytics Technical Report 1366C. Analytics: Willow Grove, PA.

PRESCRIPTIVE ORGANIZATION THEORY IN THE CONTEXT OF SUBMARINE COMBAT SYSTEMS

Dr. Rex V. Brown

Decision Science Consortium, Inc., 7700 Leesburg Pike, Suite 421, Falls Church, Virginia 22043

Abstract. Prescriptive organization theory is a new discipline which we believe has major long term promise for the design of Navy combat systems. A significant problem in the development of improved combat control and other decision aiding systems, is that the role of organizational factors is imperfectly understood and imperfectly allowed for in system design. For example:

- organizational resistances can impair the effectiveness and acceptability of computerized and other aids;
- the organizational side-effects of these aids may be overlooked; and
- the capacity for aids to actually solve organizational problems may not be fully exploited.

Organizational theory has lagged behind other analytic technologies relevant to system design, such as data management and engineering. The current development of new Navy-sponsored design concepts (such as SUBACS) to enhance submarine combat control systems presents us with a major opportunity both to develop the general state-of-the-art of prescriptive organization theory and to enhance the effectiveness of the combat systems themselves.

DSC is actively working on a number of applied submarine/combat control problems, which both expose its staff to the organizational problems involved and give them direct access, without incremental cost or disruption, to fleet experience and expertise. The researchers involved in these problems have been sponsored by ONR and NSF to synthesize the current state-of-the-art of organizational prescription, and to generate hypotheses which can be generalized to decision-aiding systems in general.

Our research approach develops three interrelated issues:

- Analytic: the appropriate form for prescriptive theory (through building blocks of descriptive theory, to be synthesized into ad hoc prescriptions).
- Empirical: promising content for such theory (incorporating established, emerging and new theories of organizational process structure, use of experts, centralization, motivational field, etc.).
- Normative: the validation of general theory or specific prescriptions (e.g., using empirical pre-testing approaches developed by our staff (Brown and Watson, 1977) and using multiattribute, multiconstituency decision theory).

The key research activities involve:

- observation in depth of organizational situations (e.g., exercises);
- informal canvassing of the practice, experience and opinion of Navy personnel;
- secondary research on documented case studies and other papers;
- analytic development of concepts and techniques.

INTRODUCTION

This paper is to be considered in the context of an ongoing DSC program of effort to enhance tactical command decisions on board attack submarines, which itself was an outgrowth of ONR's recent Operational Decision Aids program. Some background on this effort will be helpful and may also be of interest in its own right.

Our initial focus was almost entirely on developing algorithms that would process available data and judgment as logically as possible, specifically to help an attack submarine commander in approaching an enemy submarine with intent to destroy. Using decision analysis techniques, data from the fire control system was combined with judgments of uncertainty and value to infer conclusions about probable target range, appropriate time-of-fire, etc. (Cohen and Brown, 1980). For example, multiple, conflicting target range estimates from existing TMA solutions were pooled, using recent technical developments in decision analysis, such as the reconciliation of incoherent judgments (Brown and Lindley, 1982). When validated on exercise data, the results were quite encouraging. For example, the target range estimates showed an improvement of some 20% over those currently used in fire control solutions. So far, so good. However, it rapidly became clear, when we canvassed the user community of submarine skippers and such on the value of our aids, that a critical sticking point in the way of their effective applications was man-machine interface. We found a good deal of resistance to the intimidating and confusing black boxes which already dominated the combat center. The prospect of "more of the same," however sophisticated, inspired little enthusiasm. This led us to study the human operability of combat control systems (Cohen and others, 1982a) and then to develop personalized displays which would smooth the interface between man and model and to exploit the logical capabilities of the latter more effectively (Cohen and others, 1982b). The computerized displays enable the ship's commander to put up the information he wants in the order and form that he wants it. For example, he may sketch out a trial maneuver he wishes to evaluate on a screen and interrogate the underlying decision model on the probability of a "first shoot kill" if he fires at different points along the path.

It is still too early for us to know if the CO will find our display devices as useful as we hope in putting the logic of our aids at his effective disposal. However, it has already become clear that, even if they should be, tactical decisions may still be seriously impaired for reasons which have to do with the institutional setting within which the CO operates. For example, he may be motivated to wait "too long" to fire his torpedo, because, in post-exercise "hot wash-up" evaluations, he is rewarded most by having an accurate fire control solution (which he can achieve by getting in dangerously close) and will not be penalized (as he would in wartime) by a successful enemy counterattack. International comparisons based on exercises and wartime experience, suggest that the combat effectiveness of British, Australian and German Navy units typically exceeds that of comparable U.S. units due to better organizational and cultural climates. Accordingly, and in a response to similar indications from other parts of our decision process consulting practice, we have begun to devote significant effort to the organizational aspects of decision aiding, which is the prime focus of this paper. However, unlike the logical and psychological aspects of decision aiding, organizational prescription finds very few theoretical foundations in the literature to build on. We have, therefore, had to back off from the applied problem to develop some prescriptive organizational theory[1], whose elements we will now discuss.

SCOPE OF PRESCRIPTIVE ORGANIZATIONAL THEORY

We are concerned with prescribing two types of organizational action:

- external - action the organization takes on the outside world (such as to fire torpedo);
- internal - action within the organization (such as to install a decision aid).

Organizations are typically set up in order to take external actions. Internal actions are typically taken to effect or enhance external actions, and are the only ones which can be directly controlled by a system (organization) designer.

When prescribing action--whether external or internal--we need to take into account what constituency is being served. Is it: an individual, such as the President; the people that make up the organization in question; or some outside social group, such as society at large? In the military context, it is clearly some formulation of (or surrogate for) the latter.

Three further elements need to go into prescribing an organizational action: information specific to the problem at hand; a logic for drawing a conclusion; and an appropriate body of substantive material on which the logic operates. The substantive material may be organizational (say, relating to group dynamics) or not (say, weapons technology or acoustic science). It may be some established scientific doctrine, or some ad hoc judgment tailored to the situation.

Any prescriptive theory is essentially conditional. It depends on features of the situation, which in the case of external organizational action are so variable as to defy any but the most local generalization. For

---

[1] Supported by the Technology Assessment and Risk Analysis Division of the National Science Foundation and the Psychological Sciences Division of ONR.

example, submarine tactical doctrine is bound to be largely situation (e.g., threat) specific; unlike principles of engineering design or even interpersonal relations (as witnessed by Ovid's enduring advice on the art of loving!) Although classical prescriptive military theorists such as Clausewitz produced promising general precepts, the most we can hope for here of any universality is a technique for deriving external prescription; perhaps augmented by a compendium of useful substantive materials to draw on as the building blocks of prescription (e.g., findings of social psychology or physics). Personalized decision analysis (Bayesian decision theory), if adapted for distinctive features of organizations (notably the elicitation of utility and uncertainty) would seem perfectly appropriate here.

As far as internal prescription is concerned, we are more ambitious. Machiavelli, after all, was able to base his place in history on what is essentially a body of internal organizational precepts! More recently, Lorsch, Lawrence and others have proposed interesting prescriptions for the design of industrial organizations. In the Appendix a taxonomic framework is put forward for evaluating internal actions which characterizes factors as design, performance, mediating or setting variables.

We propose to limit ourselves largely to those internal actions (not necessarily organizational) aimed at enhancing the quality of external action (for example, having to do with effective command and control or decision-aiding procedures). The major problem to be addressed here is: how do you get organizations to take optimal external action? This has to do with overcoming the institutional and other impediments to rational action. As March and Shapira (1982) have pointed out, organizations are commonly observed to behave "foolishly," i.e., clearly non-optimally from virtually anyone's point of view. Allison (1971) has persuasively documented how far the bureaucratic processes of public organization fall short of what one would expect from a "rational unitary actor," in his analysis of the Cuban Missile Crisis. This suggests that there is enormous room for improvement in the quality of external actions, and that it is promising to explore how internal action might be used to realize that improvement.

External action is typically the result of a complex distributed process involving numerous individuals acting in concert or in conflict, in parallel or in sequence. Firing a torpedo, for example, involves two-way communications between sonar operator, fire control operators and coordinator, executive officer and commander. Any one of them may behave dysfunctionally for reasons that may be susceptible to internal action. A MATE operator, for example, may delay reporting valuable but uncertain information about target location to the CO, for fear of criticism that his "solution is not ready." The appropriate internal action (by system designers) might be to take the initiative out of the operator's hands by requiring a continuous updated display of uncertain target location; or to reward him for timely reports without penalizing uncertainty.

The technology of manipulating the distributed decision process giving rise to external action through internal actions to bring the former into line with rational prescription has been largely overlooked by the research community. It is possible that the issues are sufficiently well circumscribed that general prescriptive principles can indeed be sought. One such principle might be that the reward structure for individuals in the process should be manipulated to coincide with the value structure to be optimized for the organization. (In business, this might be achieved by making shareholders of employees.) For the time being, however, we will concentrate on a much narrower setting, submarine combat control, and consider later what findings can be generalized.

Our focus is on the internal actions which constitute the design of command and control systems used by the commanding officer of an attack submarine. They might include:

- the installation of command decision aids with the results (available to higher echelons) either optional or required;
- assignment of tasks among individuals, say by centralization or by conditioning command structure on engagement development (such as "battle stations");
- physical layout and organization of equipment (for example the use of a large screen display);
- comparison of radical with adaptive design changes;
- the manipulation of reward structures.

How alternative actions might be related to external (operational) decision quality is illustrated in Fig. 1. The arrows correspond to causal linkages, whose evaluation requires assessment of largely organization factors. This assessment may be extracted from existing descriptive organizational theory (to the limited extent that it has currently been developed) or it may be the result of direct application of whatever judgment is available. For example (taking the first arrow in the figure), the assessor may feel intuitively that a CO is most likely to use any aids provided in a multi-threat scenario; or he may draw on theory which says that decision makers under stress revert to earlier learned procedures (which would imply the opposite conclusion).

Proceeding to the next arrow down, if the aid is used, one may conclude, say, that it will enhance tactical decision quality in two ways: by allowing the CO to better meet his own objectives and by forcing him to align his own objectives with those of the country he serves. Note that the first conclusion (unlike the second) would be based on decision analysis or exercise data, rather than descriptive organization theory.

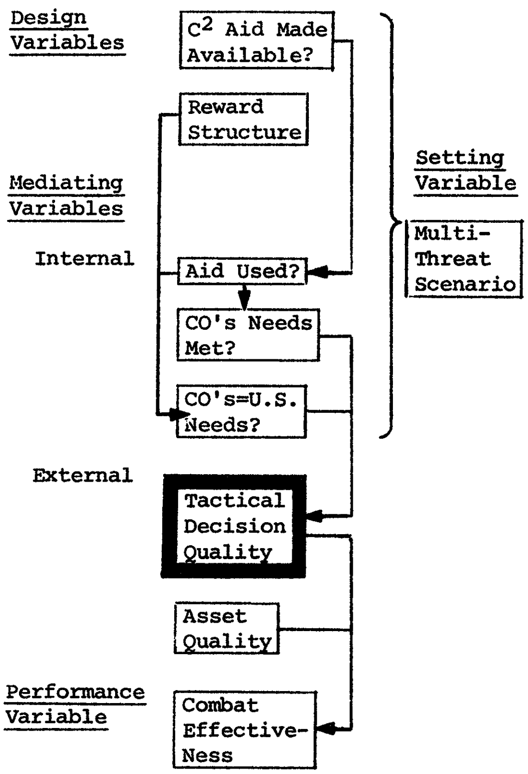


Fig. 1. Design impact on tactical decision quality.

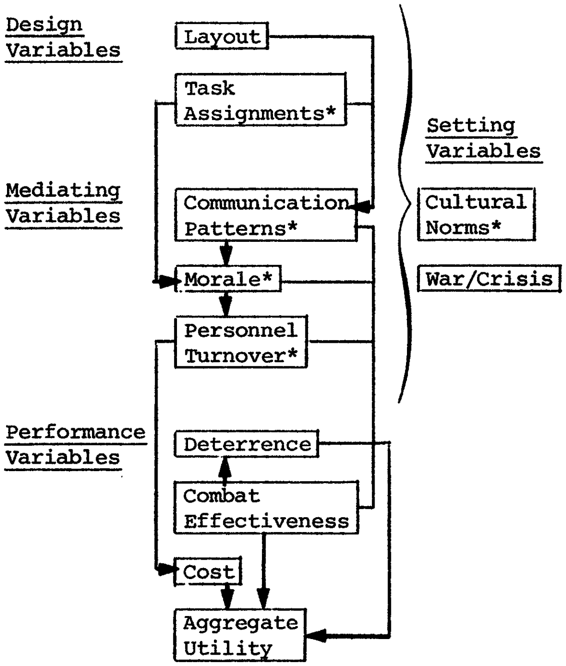
Linkages of this type can be derived in principle for any internal actions; and from these the implications for external action quality can be assessed. Although an explicit probabilistic model could be constructed for such an analysis, we believe it makes most sense to handle the issues informally and qualitatively. The effort used in quantifying the linkages is almost certainly used more constructively in getting right the qualitative nature of what will surely be a complex and subtle model.

APPENDIX: ORGANIZATIONAL FACTORS IN SYSTEM DESIGN[2]

There is virtually no established body of precept in organizational behavior (the way there is, for example, in individual human factors). Certainly there is none at a general level and there are only very tentative contributions directed specifically at combat system design (Meister, 1976). However, study of the largely descriptive organizational literature, supplemented by our own reflection and observation, suggests some tentative hypotheses and promising directions of inquiry for developing a prescriptive organizational theory.

An Analytic Framework

Generally four types of variables need to be considered when evaluating system designs--design options, performance, mediation, and setting--each of which may have significant organizational elements. Fig. 2 gives some hypothetical examples and linkages in a combat system context.



*Organizational factors

Fig. 2. Some linkages between organizational and other factors in system design.

Organization can itself be part of the design, as a set of controllable options. Organizational options may include: manning requirements; who reports to whom; task and duty assignments (as in "battle stations"); rewards and penalties; command and control procedures and doctrine; and channels of communication. These factors may be elements of the system being designed (such as SUBACS); or they may figure in a larger system design (such as the Submarine Navy), which may interact with the smaller system.

System performance can be expressed in a hierarchy of attributes, each of which contributes to a higher level, with possibly a combined measure of overall utility at the top. One level below will be major "bottom-line" defense objectives, such as: combat effectiveness; influence over the enemy (e.g., by deterrence and deflection); general political and military power; system cost (initially and to maintain and to operate). These tend

---

[2] Adapted from Cohen and others (1982a).

not to be distinctively organizational, except to the extent that organizational "health" is held to be desirable in and of itself. The next level down will include organizational performance measures, such as personnel retention, which are viewed as valuable, but only as they enhance combat effectiveness or reduce cost. Operability measures generally would be in this category.

Mediating attributes, i.e., those whose realization will tend to influence the more basic attributes, are still further down in the hierarchy and include many organizational effects produced by design. For example, physical layout may affect informal communication and other organization patterns, which in turn affect combat effectiveness, directly or through retention. (What we consider "mediating," as opposed to "performance," variables depends on whether it is clear that "more is better"--in which case it seems reasonable to call it "performance.")

Uncontrollable setting factors of organization may have important impacts on design, including: the permeating influence of western, military or submarine culture; the general determinants of promotion in the submarine Navy (who you have to impress and how); the immutabilities of Navy command structure (e.g., the problems of having a junior rank give direction to a senior rank), and the scenario within which the system is operating (e.g., peace or war).

Our interest is in the way organization factors in all four categories interact with each other and with non-organizational factors, as illustrated in Fig. 2. Understanding the linkages contributes to the evaluation and technology of system design. For example: physical layout impacts the organizational flow of power and communication; reward structures may impact on individual stress; morale may affect organizational effectiveness and the capacity to withstand catastrophe (e.g., if key officers are out of action in an engagement).

Organizational Implications of System Design

Centralization. A tentative conclusion cited by Meister (1976, p. 263) is that a centralized team structure, with a leader in a commanding role, works less effectively when the leader is under stress. This would seem to suggest that as submarine warfare becomes more stressed, for example through multiple threats and targets, the most appropriate organization may shift away from the highly centralized one previously found effective on submarines. Alternatively, it may suggest that the organization should change in response to engagement developments, becoming looser as the environment becomes more complex and stressful. This could be a form of organization discrimination within "battle stations."

Decentralization, and delegation of authority more generally, interact with system design, especially the adoption of performance aids which incorporate judgment external to the operator. Examples include: preprogrammed maneuvering aids for submarines; automatic collision avoidance mechanisms on airplanes; and general purpose decision aids that incorporate higher-level value judgments.

Such aids automatically tend to bring about centralization, in that command influences are felt lower down in the organization through parameters of the aids being, in effect, agents of higher authority. In addition, the aids encourage the introduction of more centralized organizational designs, since these then become more feasible and effective. For example, if the Pueblo had had an aid installed which prompted its response to a threat of the type it in fact faced, the commander's hands would have been somewhat more tied by the judgment of his superiors (especially, of course, if the aid produced a firm prescription, rather simply than a guide). In addition, the very availability of the aid makes it feasible to change the command structure deliberately in a more centralized direction; for example, by reducing the discretion allowed the lower level commander and/or increasing the specificity of standard operating procedures.

Centralization affects where in the organization operability needs are most critical. Centralization tends to make the task of high level personnel more exacting and, therefore, makes greater demands on operability at that level. On the other hand, it may relieve demands for skill at lower levels of command and so reduce the need for operability there. Whether this is an argument in favor of centralization depends on whether, say, senior officers or junior technicians are more difficult to recruit, train, and retain.

Adaptive versus radical system designs. One of the design dimensions of man-machine systems is the extent to which a new system is a radical departure from the status quo. Lindblom and other political scientists have argued that, largely for organizational reasons, adaptive solutions of "creeping optimization" are generally to be preferred to more radical solutions (Lindblom, 1959). The soundness of this general precept, and its dependence, if any, on properties of the task at hand, could have an important impact on the design of military systems. For example, it may only hold in a world where the environment of interest is available to test and provide feedback on adaptive changes so that these can be reversed or adapted as appropriate.

This may not be true in an environment (typified by military conflict) where the design must be finalized without benefit of any real world test (e.g., hot war); and where it would be slow and costly to test successive adaptations in a surrogate environment (e.g., through military exercises).

Motivational field. An important question to be addressed explicitly in evaluating any design proposal (whether itself an organizational design or not) is how it impacts, or is impacted by, the motivations of players in the system being designed. For example, a fire control system which documents exactly how the fire control coordinator does his job may increase his perceived exposure to criticism after the fact and thereby impair the acceptability of the system (even though it may actually improve his performance). This is a factor to be taken into account when considering the operability of a system.

Conversely, systems may be designed to overcome dysfunctional motivational pressures. To pick up again on the Pueblo example, suppose the commander attaches more importance to protecting his crew than to avoiding an international incident. If his superiors differed, they might design a system (say, through standard operating procedures) which relieved him of responsibility for making choices where that trade-off was relevant.

The Design of the Organization

Command structures. Researchers at the Harvard Business School have developed guidelines on how organizational structures should be adapted to the task at hand (Lawrence and Lorsch, 1967). For example, they argue that research tasks call for looser, more diffuse lines of command than the conduct of day-to-day operations in business enterprises. These are static settings in the sense that tasks are fairly stable, whereas combat units, like submarines, have a critical dynamic aspect not found in the business situations where most prescriptive research has been done. The environment a fighting system faces may change dramatically at short notice and with it the type of organization which is appropriate. This idea is recognized at least implicitly by the practice of moving into "battle stations" when combat is engaged. This involves primarily a re-assignment of responsibilities. Finer discrimination may be called for, for example, in different phases of an engagement, but the principles that should underlie it have yet to be established. For example, the approach phase of a submarine engagement may have different requirements from the torpedo firing phase.

The assumption that the appropriate practice will evolve over time of its own accord may be appropriate where the nature of warfare is stable. However, when the perceived nature of warfare is changing rapidly, because of development in own technology, perceived threat, etc., this assumption may be dangerous. For example, when the complexity of the environment increases beyond that encountered in the past (say, because of multiple targets and threats in a submarine situation), stress on the players in the system may increase to the point where a switch to automatic procedures is called for, which was not previously envisioned.

Mode of technical support. As the technical sophistication of military (as of other) systems increases, the attendant increasing need for operability raises the issue of whether technical experts should be made available to the users of the technology. A survey conducted by DSC staff (Brown, 1970) in a business context suggests that the use made of technical expertise will depend critically on the organizational mode in which it is offered. In particular, a user will be reluctant to use a technical service if the expert is organizationally distant; for example, if he reports to a potential rival in the power structure. The survey suggested that the typical user will accept a lower level of technical support in order to avoid this dimension of threat. For example, he will prefer the marginal technical superiority of a non-threatening close colleague than the higher expertise of an organization-wide specialist (such as a centralized expert on a submarine).

Other Issues

Adaptation of well-established behavioral concepts. Political scientists such as Graham Allison of Harvard have developed models of how organizational decision units operate (Allison, 1971). In particular, he has proposed a "bureaucratic process" model which explains behavior in organizations, often highly dysfunctional, much better than the more obvious "unitary rational actor" model. Even seemingly highly centralized decisionmaking units like a submarine may show some of these properties, which in turn may influence appropriate designs.

Sociologists specializing in the behavior of small groups have noted what appear to be universal properties of relationships in small groups, which could be adapted to system design. For example, they have noted predictable linkages between sentiment, authority, and interaction, and a tendency to urge congruence between different indicators of status (Homans, 1950).

Social psychologists studying the nature of power in organizations have noted that power and other rewards accruing to individuals in an organization are a function of organizational structure (Tannenbaum, 1968). For example, there are certain organizational structures which permit everyone in an organization to have more power, or at least for the total amount of power to be increased. An important issue to address (in view of practical difficulties of retaining skilled military personnel such as submarine officers) is how such rewards trade-off against others which may be more difficult to enhance (such as income and family contact) and how they can be engineered into the system, if viewed as valuable.

Comparison of organizations and individuals. One of the reasons why there is less in the way of research findings with actual or potential value to military system design than in individual psychology should be born in mind when attempting to generate such findings. Humans are immensely more uniform than organizations. Generalizations about humans in general are likely to give very good insight into any particular human. The diversity of organizations is such that any generalization about organizations in general must usually be hedged with a multitude of qualifiers before it applies usefully to any particular organization. Difficulties of research methodology aside, there may not be much in the way of universal generality to look for! While this may be too extreme a position to take, it may nevertheless be true that the appropriate unit of investigation may be, not organizations in general, but narrow classes of organizations, or even specific organizations. (For example, the conditioning properties of the organization may need to be so finely defined that the only place that one can be found is on a submarine!) In any case, this phenomenon would certainly argue for cautious inductive extrapolation beyond particular cases.

Furthermore, as Hogarth has noted (Hogarth, 1981), the universe of organizations is not limited to those that currently exist. New organizations can readily evolve or be designed in a way that, to all intents and purposes, humans cannot. It is entirely possible that with the emergence of new technologies such as automated data processing, new organizational forms may need to be developed that do not have any close analogy in history. In such cases the application of deductive principles in the design of new organizational structures may not be helpful (by contrast, say, to the development of engineering systems).

On the other hand, experimentation with organizational designs may be more tractable than with engineering systems which may call for massive irreversible investment and delay before a prototype can be tested. For example, alternative organizational structures could be tried out on submarine exercises at much less cost than testing a proposed new fire control hardware/software system, much less a new submarine hull design.

REFERENCES

Allison, G.T. (1971). Essence of Decision: Explaining the Cuban Missile Crisis. Little, Brown and Company.

Brown, R.V. (1970). Do managers find decision theory useful? Harvard Business Review, May-June, 78-79.

Brown, R.V. and D.V. Lindley (1982). Improving judgment by reconciling incoherence. Theory and Decision, 14, 113-132.

Brown, R.V. and S.R. Watson (1977). Pretesting innovation: methodology for testing the design of management systems. Theory and Decision, 8, 315-336.

Cohen, M.S. and R.V. Brown (1980). Decision Support for Attack Submarine Commanders. Decision Science Consortium, Inc., Falls Church, VA.

Cohen, M.S., R.V. Brown, D.A. Seaver, J.W. Ulvila and W.G. Stillwell (1982a). Operability in Attack Submarine Combat Systems: An Exploratory Review (Technical Report 82-2). Decision Science Consortium, Inc., Falls Church, VA.

Cohen, M.S., R.C. Bromage, J.O. Chinnis, Jr., J.W. Payne, and J.W. Ulvila (1982b). A Personalized and Prescriptive Attack Planning Decision Aid (Technical Report 82-4). Decision Science Consortium, Inc., Falls Church, VA.

Hogarth, R.M. (1981). Decision making in organizations and the organization of decision making. Proceedings of the 8th Research Conference on Subjective Probability, Utility, and Decision Making, Budapest, August.

Homans, G.C. (1950). The Human Group. Harcourt, Brace, and World, New York.

Lawrence, P.R. and J.W. Lorsch (1967). Organization and Environment. Irwin, Homewood, IL.

Lindblom, C.E. (1959). The science of muddling through. Public Administration Review, 19.

March, J.G. and Z. Shapira (in press). Behavioral decision theory and organizational decision theory. In Ungson, B. and Braunstein, D. (Eds.), New Directions in Decision Theory. Kent Publishing Co.

Meister, D. (1976). Behavioral Foundations of System Development. John Wiley & Sons, London.

Tannenbaum, A. (1968). Control in Organizations. McGraw-Hill.

ADDENDUM

Exercise validation data for the target ranging technique referred to is contained in:

Cohen, M.S. Quantitative validity of objective and subjective inputs for a target range pooling decision aid. To appear in the Proceedings of the 49th MORS Symposium, 1982.

DECISION PROCESSING IN A DISSEMINATION SYSTEM FOR MILITARY MESSAGES

Judith Froscher, Anne Werkheiser, and Joan Bachenko

Navy Center for Applied Research in Artificial Intelligence
and
Computer Science and Systems Branch
Naval Research Laboratory
Washington, D.C. 20375

Abstract. At the Naval Research Laboratory, we are using computational linguistics and artificial intelligence to develop a computer-based dissemination system for military messages. The disseminator uses message content, a body of factual knowledge, and information about message recipients to determine a distribution list and an urgency rating for Navy operational reports. Message content is extracted by the message analyzer which uses linguistic techniques to interpret the English narrative in messages. This paper presents a description of the dissemination environment and its constraints and discusses how this description was represented as a knowledge base.

INTRODUCTION

The level of technology exhibited by today's C3 systems is uneven. Message transmission, for example, is supported by a very high technology, but methods for exploiting message content remain rudimentary and inadequate. If C3 systems are to be capable of dealing with the content as well as the form of message transmissions, they must be provided with qualities we usually think of as human: knowledge of the world, the capacity for language, and an ability to draw inferences.

This paper describes an experiment in progress that applies concepts in linguistics and artificial intelligence to the automated dissemination of Navy messages. Our goal is to build a message dissemination system that makes fine distinctions among message recipients and is responsive to changing world conditions. Decisions about dissemination will be based on the information content of a message, a body of factual knowledge about the world, and the special interests of message recipients. To accomplish this, the system will

- analyze an incoming message and extract its meaning;
- use a knowledge base to represent the dissemination environment.

Since decisions take place in a changing world, the system must have the following features:

- be easily modified to reflect changes within Navy organizations;
- allow users to interactively manipulate the knowledge base in order to accommodate unforeseen changes in world events.

These are very broad expectations. In order to achieve early success we have limited the problem to the dissemination of operational reports, which are formatted messages containing a mixture of English and pro forma data. Our immediate goal is to build a system, Charlotte, that disseminates reports of shipboard equipment failure; such messages are called Casualty Reports, or CASREPs.

Charlotte is a member of a family of application systems that perform such tasks as dissemination, database update, and question answering for CASREPs, UNITREPs, RAINFORMs, and other operational reports. Each member of the family uses a message analyzer to extract message content. The analyzer comprises a text grammar that partitions a message into information fields and an English grammar that interprets the narrative in an information field. The purpose of the analyzer is to extract those aspects of message content that are required for a given application.

This paper focuses on the design of the Charlotte system. We begin with a brief discussion of message analysis. Then we present an initial description of the dissemination environment and discuss how this description was implemented as a knowledge base. We conclude with an overview of our future plans and a summary of our results.

MESSAGE ANALYSIS

Within the Charlotte system, the main tasks of the message analyzer are (1) to extract CASREPs from an unrestricted stream of messages, (2) to transform each CASREP into a set of information fields (e.g., equipment identification, problem description, ship location, and so on), (3) to derive semantic structures (or meanings) for the English expressions that occur in a given information field, and (4) to map these structures onto representations that can be used by the application. The output of the analyzer is a message representation that summarizes the information content needed for dissemination. A detailed design of the analyzer is given in (Froscher, Hayes, Bachenko, 1982).

To generate a message representation, the analyzer depends in part on the format conventions that govern the content and form of a CASREP. For example, the CASREP format requires, among other things, a section in which technical assistance may be requested. However, the format does not specify the way in which this request must be made. It can be a pro forma expression, e.g., "NONE", or a narrative expression, e.g., "SHIPS FORCES WILL INSTALL". The following examples are taken from the "assistance request" section of actual CASREPs.

(1) a. NONE
    b. N/A

(2) a. REQUEST NAVSEA ASSIST
    b. CONTRACTOR ASSISTANCE REQUIRED
    c. REQUEST NAVSEA COORDINATE DELIVERY OF REPLACEMENT
    d. SHIPS FORCES WILL INSTALL

We have determined that message distribution is affected by whether or not the message asks for assistance. Thus for Charlotte to perform dissemination, the message representation must provide a "yes" or "no" answer to the question, "Is assistance needed?". That is, the expressions that occur in the assistance section, e. g. those in (1a-b) and (2a-d), must be reduced by the analyzer to a simple "yes" or "no".

When the message contains pro forma items such as (1a) and (1b), deriving an answer is straightforward because there exist direct and simple links between surface form and meaning. The English expressions in (2a-d), however, are neither pro forma nor stereotyped items, but free narrative expressions in which the connections between form and meaning are obscured. To derive an answer from these expressions therefore requires linguistic techniques that can uncover the meanings encoded in words and word order patterns. In the Charlotte system, a grammar of English derives semantic structures that are neutral with respect to application requirements. A body of application-dependent rules maps these structures onto representations that can be used for dissemination, e.g., the "Yes" or "No" statement for "assistance request".

DISSEMINATION ENVIRONMENT

The real world context of message dissemination is complex and hard to describe. There is no single expert or group of experts on CASREP dissemination, nor is any single organization completely responsible for CASREPs. Current dissemination practice obscures the picture further, since it is based on deliberate overdissemination. This often means that there is no well-defined relationship between the CASREPs received by an organization and the organization's real interests. We know, however, that this environment is subject to certain constraints. For example:

- CASREPs are factual reports whose purpose is to provide explicit information about the item that failed and the effect of a failure on materiel readiness.
- Navy organizations have well-defined mission areas and responsibilities. With respect to CASREP processing, organizations perform such activities as technical analysis, readiness assessment, and repair assistance.

These constraints provide a general framework for describing the dissemination domain. They allow generalizations about the environment that can be embodied in a formal description.

The information on which our description is based is the result of considerable field work. Interviews with CASREP recipients at system and type commands revealed that CASREPs are used for a variety of purposes by people with differing interests. They are disseminated broadly within a command; each recipient selects any aspect of the CASREP that he can use.

Although the data about dissemination are diffuse and anecdotal, they can nonetheless be structured according to a simple principle. Navy organizations that receive CASREPs can be distinguished with respect to two primary interest areas: ships and equipment. For example, in the domain of CASREP dissemination, COMNAVSURFLANT is a "ship organization", and COMNAVELEXSYSCOM is an "equipment organization". Within a ship organization, "equipment" may form an optional and secondary interest area; e.g., the Hull, Mechanical, and Electrical Branch (N772) of COMNAVSURFLANT is an "equipment" suborganization. Similarly within an "equipment" organization such as COMNAVSEASYSCOM, "ship" may form an interest area; e.g., Surface Combatant Ships (NAVSEA-93) is a "ship" suborganization. This branching pattern is recursive because it occurs at all levels of an organization. It is the basis for our description of the dissemination environment and, thus, for the design of the knowledge base.

IMPLEMENTATION OF THE DISSEMINATOR

Our initial description of the dissemination environment was implemented as a knowledge base, using the Knowledge Engineering System (KES) developed by Reggia (1981). We selected KES for this work because it provided a means of building a quick prototype, it is well supported through prior testing and evaluation, and it has a friendly user interface. The interface is an important feature because it allows us to enter the data from the CASREP manually. The manual interface will soon be replaced by the message analyzer, which is now being implemented.

A central requirement for the dissemination subsystem is that it provide a list of recipients for each message. That is, the disseminator must produce many decisions for a message rather than a single decision. This distinguishes our application from others, such as, medical diagnoses where the goal is to provide the minimum number of decisions that account for the symptoms. Another requirement is that the system be flexible, i.e., the disseminator should be responsive to changes in Navy organizations and in existing world conditions.

The production rule system, KES.PS, allowed us to satisfy both requirements in the initial implementation. An alternative system, KES.HT ("Hypothesis Testing"), has more descriptive power but was rejected since it is geared to a diagnostic application and as such, does not support the "many decisions" requirement.

The knowledge base comprises two sections. The attribute section lists variables along with their possible values. For example, "addressee" is a variable whose value can be COMNAVSEASYSCOM, COMNAVELEXSYSCOM, or COMNAVSURFLANT. "Assistance request" is a variable whose value is "Yes" or "No". "High interest item" is a variable whose value can be a ship name, a ship location, an equipment item or "none". Descriptions of message recipients are given in terms of attribute values.

The rules section contains if-then statements that use current attribute values to make inferences about distribution and urgency. For example, if the ship organization COMNAVSURFLANT is an addressee on the message, and if "assistance request" has the value "Yes", then the message will be sent to the equipment suborganization N42, Maintnance Support Office.

In the current knowledge base, attributes are distinguished by whether they are input attributes or derived attributes. Input attributes are those whose values are determined by message content or set by the user at runtime. "Assistance request" is an input attribute whose value is provided by the message; the value of "high interest item" is specified by the user when a particular ship, ship location, or equipment item demands special attention. Derived attributes are inferred from the values of input attributes. For example, "ship class" with a value of "Spruance" is inferred from the "ship name" with a value of "Moosebrugger".

That the value of an input attribute can be given by the user provides one facility for making the system flexible. For example, given the attribute "high interest item", the user can assign it the value "South Atlantic", so that the system contains a rule essentially like that in (3):

(3) if ship location = "South Atlantic", &
        high interest item = "South Atlantic"
    then urgency = "high".

The system allows such interactions in order to accommodate changes in world events. Usually rules such as (3) will be active for only a short period of time.

Changes in the dissemination environment require the knowledge base to be changed and reparsed. For example, the rule in (4) states a fact about the dissemination environment:

(4) if ship type = "frigate"
    then distribution list = "NAVSEA-93"

A change in the cognizance of NAVSEA-93 should be reflected in the rule, i.e., the rule could be deleted, the attribute values could change (e.g., "submarine" replaces "frigate"), or new attributes could be added (e.g., if ship class = "knox", & ship type = "frigate" then distribution list = "NAVSEA-93"). Making these alterations is a complex task because each affects the knowledge base. Since the current knowledge base is small, such changes can be made quickly and easily. As it becomes more complete, however, this flexibility will become more difficult to support. Ease of change will, therefore, be a central requirement for future implementations.

FUTURE PLANS

Future work on the disseminator will focus on two related areas: evaluating alternative implementations and constructing a knowledge base that is both more comprehensive (more organizations, more ships, more equipment) and more detailed (finer distinctions among recipients). In such an expanded system, software engineering considerations will play an important role.

Our approach to making the knowledge base easy to change will use the software engineering techniques of information hiding modules (Parnas, 1972) and a uses hierarchy to facilitate expansion, contraction and subsetting (Parnas, 1979). The current knowledge base contains rules for disseminating within several commands, but an expanded version would be subsetted, with each command being given only the subset of the knowledge base that it needs.

Our goal is to produce a demonstration system by the summer of 1983. The dissemination system will include an expanded knowledge base and a pilot message analyzer.

CONCLUSIONS

This paper has presented an approach that applies methods of linguistic analysis and artificial intelligence to the automated dissemination of CASREPs. We have described the dissemination system, Charlotte, as a member of a family of systems that use application-independent techniques for extracting message content. Our discussion has focussed on a model of the dissemination environment and its representation as a knowledge base. Specifically, we have

- provided a means for organizing the dissemination data;
- identified requirements for the knowledge base representation of a dynamic environment; and
- described the role of linguistic techniques in a dissemination system.

ACKNOWLEDGMENT

We are indebted to Constance Heitmeyer for her valuable comments on earlier drafts of this paper.

REFERENCES

Froscher, J., Hayes, K., and Bachenko, J. (1982). A Prototype Message Analyzer for the CHARLOTTE System. NRL Technical Memorandum 7590-229:JF:KH:JB. Naval Research Laboratory, Washington, D.C.

Parnas, D. (1972). On the Criteria to be Used in Decomposing Systems into Modules. Comm. ACM, 15, pp. 1053-1058.

Parnas, D. (1979). Designing Software for Ease of Extension and Contraction. IEEE Trans. Software Engineering, SE-5, pp. 128-137.

Reggia, J. (1981). Knowledge-Based Decision Support Systems: Development through KMS. University of Maryland Department of Computer Science Technical Report TR-1121.

BATTLE — An Expert Decision Aid for Fire Support Command and Control

J. R. Slagle, R. R. Cantone, and E. J. Halpern

Navy Center for Applied Research in Artificial Intelligence
Naval Research Laboratory — Code 7510, Washington, D.C. 20375

Abstract. We have implemented an expert consultant system (called BATTLE) to aid in the assignment of a particular set of weapons to a set of targets, given a battlefield situation. BATTLE illustrates the application of several artificial intelligence techniques to a decision aid for the Marine Corps. A pruned tree traversal, though not guaranteed to discover the optimal solution, will attack the problem of assigning weapons to targets from the viewpoint of finding a globally optimal assignment plan. This should be an improvement on present algorithms that optimize weapon to target assignments only in a local sense. Inference networks allow BATTLE to consider many relevant environmental conditions while making its decisions. The use of dynamic inference networks allows military experts to update the program and keep it consistent with new developments in military doctrine without actually altering the computer code. These design considerations should be practical and very useful in some future version of the Marine Integrated Fire and Air Support System (MIFASS), the current decision aid under production for the Marine Corps.

Introduction

Decision aid technology is one of the more rapidly expanding uses of computers. One such aid, the Marine Integrated Fire and Air Support System (MIFASS), will "provide for the establishment of fire and air support centers to plan, integrate, direct, and coordinate the fires of supporting arms."[5, pp. 2-19] An algorithm for use in the MIFASS system to direct fire power was developed at Oklahoma State University (1975-1977). The Oklahoma State University algorithm attempts to discover good fire support allocation schemes through an iterative technique, always optimizing the weapon being selected for the next most crucial target.[1,2] Selection of the next target to be assigned is determined by a heuristic formula.

An inherent limitation of the Oklahoma State University algorithm is that potential targets are considered separately, rather than simultaneously. An optimal solution would consider the problem as a whole, rather than as a set of individual, smaller problems. It is probably not possible to find an optimal allocation scheme in a reasonable amount of time. The time required using a tree search without pruning would be proportional to the number of targets plus one, taken to the power of the number of weapons. Generation of a complete tree of possible weapon to target assignments and exploration of that tree for the optimal assignment would require an unreasonable amount of computing time for a typical battlefield scenario. Rather we wish to investigate the possibility that applying tree traversal with pruning to the assignment tree might provide a reasonable solution to the assignment problem. Even with pruning, however, this will probably require more computing time than the iterative technique developed by Oklahoma State University, so their technique is probably more suitable for very large scale battlefield situations. But for smaller scale situations, our technique will provide more accurate decisions.

An important part of our system, which could also be incorporated in MIFASS, is a dynamic set of of environmental factors which might influence the effectiveness of a weapon against its target. In addition to the simple restrictions included in Norden's implementation, such as requirements for ammunition, fire time, and restricted fire zones, there may be many factors which cannot be known *a priori*. Troop morale, for example, might be the decisive factor in a military engagement. There is no present mechanism for the inclusion of such considerations in the programs presently being developed for MIFASS.

The proposed system, which we call BATTLE, has two major components: the computation networks and the assignment tree. The computation networks, described in the next two sections, are used to evaluate the effectiveness of each weapon against each target. The assignment tree, described in the fourth section, generates weapon assignment plans based on the effectiveness values.

The Computation Networks

We have designed and implemented computation networks which include inferencing modeled after the method of subjective Bayesian updating found in the PROSPECTOR system.[3] This is a very flexible method which may be easily adapted by military experts unfamiliar with computer programming. A computation network may be created to deal with any sort of external condition which the military experts believe might the performance of weapons. Such an ability would be an invaluable asset to the MIFASS system; it would in effect allow military experts to modify the considerations used in weapon to target assignments, and keep the system updated with the latest doctrine on what is most pertinent to weapon allocation decisions.

A *computation network* consists of a set of *facts* (or *propositions*) and the *rules* (or *links*) which direct their associations. The facts are related to one another by rules that allow information about facts to yield information about other facts. Computation networks are a generalization of the *inference networks* used in PROSPECTOR, differing in that the information which propagates through a computation network is not restricted to probabilities of truth and falsity. The computation network may involve simple tree structures or more complicated graph structures, a decision to be determined by the expert.

Facts are described by the expert in a propositional form (e.g., **friendly-is-a-good-match-against-target**). Such a fact only has meaning in a specific context, say in comparing friendly unit **friendly2** against target unit **target7**. We say that the *context type* of this fact is *both* as a shorthand for saying that it only makes sense (i.e., has a truth value) in the context of both a specific friendly unit and a specific target unit. Some other facts need just a specific friendly unit (or target unit) to take on meaning (for example, **friendly-has-ample-ammunition** or **target-position-is-accurately-known**). The context type of such a fact is *weapon* (or *target*). Still other facts are independent of any friendly or target unit (for example, **it-is-raining**) and have the context type *global*. The expert must specify the context type of each fact. The context type may be viewed as a means of specifying the set of variables upon which a fact depends.

The system supplies the expert with a variety of link types for associating facts. These are the standard logical connectives **and**, **or**, and **not** and one called **evidence**. Each link type has an associated *updating function* to propagate information through the network. The updating function provided with **and** links assigns the product of the probabilities of the antecedents to the consequent. The updating function for **or** links assigns the product of the complement probabilities of the antecedent to the complement probability of the consequent. The updating function for **not** assigns the complement of the antecedent's probability to the probability of the consequent.

The **evidence** link type specifies updating with the subjective Bayesian method described by Duda, Hart, and Nilsson.[3] This type of link provides a more symmetrical mechanism for the propagation of probabilities than either **and** or **or** links. The **and** link type will tend to keep the consequent probability low. It is very easy for a single antecedent of an **and** link to reduce the consequent probability. Similarly, a single antecedent of an **or** link may greatly increase the consequent probability, but will have very little ability to lower the consequent probability. The **evidence** link type provides a more stable situation. Probabilities tend to remain about the *a priori* level, and are influenced equally in both directions. If one antecedent has a high probability and another antecedent has a low probability, their influences will tend to cancel.

The updating function for **evidence** links proceeds in two steps. First, each antecedent E_i yields a consequent probability $CP_i(P(E_i))$ by linear interpolation between the expert-provided values

$$\begin{aligned} CP_i(0) &= P(H\,|\,\overline{E_i}), \\ CP_i(P_p(E_i)) &= P_p(H), \text{ and} \\ CP_i(1) &= P(H\,|\,E_i), \end{aligned}$$

where $P_p(E_i)$ and $P_p(H)$ are the *a priori* probabilities of the antecedent E_i and the consequent H. The probability $P(H)$ of the consequent is then derived by solving

$$\frac{P(H)}{1-P(H)} = \frac{P_p(H)}{1-P_p(H)} \prod_i \left[\frac{1-P_p(H)}{P_p(H)} \frac{CP_i(P(E_i))}{1-CP_i(P(E_i))} \right].$$

The system-supplied link types are not expected to suffice for all applications. The expert may define a new link type by writing LISP functions for propagating information between facts.

Using the Network

The user of the network, as opposed to the expert creator, has available two modes of inputting data. The *volunteering* mechanism allows the user who is familiar with the nodes in the network to enter a particular piece of information directly. The user of this mechanism specifies the fact to be volunteered and its value. The *questioning* mechanism produces a system-directed dialogue with the user. The system chooses facts based on the best combination of being relatively easy to provide and relatively influential to the top node, and requests their values from the user. Before describing this mechanism, we will define some properties and values assigned to facts in the network.

Given any fact (e.g., **friendly-has-good-morale**) in any applicable context (e.g., **friendly7**), the BATTLE system will keep track of whether the fact has been asked or answered in that context. A fact is said to have been *asked* in a context when the user has indicated that he does not know its value in that context. A fact is said to have been *answered* in a context when the user has supplied its value in that context. In addition, the expert creator of the network may mark certain facts as *unaskable*; this property is applies to the fact in all contexts. The questioning mechanism will not request the value of a fact in a context where any of these three properties apply. In the case of an asked or unaskable fact the system will consider questioning the user about the fact's antecedents in order to calculate its value. This is not done for answered facts.

The expert assigns two values to each fact. One is a *prior* (or *a priori*) probability of its truth or falsity. The other, *self-merit*, is an approximation of the variability of the proposition with respect to the cost of expanding it. A large self-merit value is assigned to askable facts whose values are deemed likely to change substantially, or to unaskable propositions with few antecedents. Unaskable propositions generally have higher self-merits than their askable counterparts. The most important principle in assigning self-merits is that they should all be given values that are correct relative to one another. To a first approximation, a changeable fact should be given a greater self-merit than a more stable proposition.

In the implementation of BATTLE, when the user opts to use the questioning mechanism, the *merit* system is used. Given a top consequent, the system finds its antecedent having the highest merit value (i.e., the most *meritorious*), wherever it might be located in the network. This fact is considered by the system to have the best combination of being easy to supply and influential to the top consequent. Questions are asked in order of decreasing merit. If the merit of the most meritorious antecedent is less than a cutoff value specified by the user, the system will stop questioning the user.

A complete description and derivation of merit is in [6]. Here we will just present its basic definition and describe its use in BATTLE. For the purposes of this discussion, assume that we have a computation network with a top proposition G and subpropositions G_i, for $i=1, \ldots,$ degree(G). Each subproposition G_i may itself have subpropositions designated G_{ij}, for $j=1, \ldots,$ degree(G_i). Additional subscripts indicate further levels down the computation network. The merit of an untried proposition $G_{ij \ldots st}$ is defined by the partial derivative

$$\left| \frac{\partial P(G)}{\partial C(G_{ij \ldots st})} \right|$$

where $\partial P(G)$ is the change in the probability of the top proposition G, and $\partial C(G_{ij \ldots st})$ is the cost of expanding the untried proposition $G_{ij \ldots st}$. Absolute value is used because increases and decreases in probability are equally important; the amount of change is the determining factor.

Note that this definition of merit describes in precise mathematical terms those qualities we desire most for the next proposition on the inference network which is to be expanded. A high merit states that a proposition will exert much influence on the top proposition with little cost. Low merits indicate that expansion of a proposition will have little effect on probabilities at the top level or that the expansion will be accomplished only at a high cost.

The merit has been expressed as a derivative relating the probability $P(G)$ of the top node to the cost $C(G_{ij \ldots st})$ of expanding an untried proposition somewhere else on the proposition tree. The chain rule may be used to express the derivative as

$$\left| \frac{\partial P(G)}{\partial C(G_{ij\ldots st})} \right| = \left| \frac{\partial P(G)}{\partial P(G_i)} \frac{\partial P(G_i)}{\partial P(G_{ij})} \cdots \frac{\partial P(G_{ij\ldots s})}{\partial P(G_{ij\ldots st})} \frac{\partial P(G_{ij\ldots st})}{\partial C(G_{ij..st})} \right| .$$

All terms save the last in this expansion have a similar form. These terms, called ***link merits***, are computed by the system using a ***merit function*** for each link type. The last factor, called the ***self-merit*** of $G_{ij\ldots st}$, represents the ability to change $P(G_{ij\ldots st})$ per unit cost applied in expansion of $G_{ij\ldots st}$. The self-merit is given as an expert opinion.

To select a question with high merit, the system starts at the top consequent selected by the user, examining all antecedents of that proposition. A merit value is calculated for each antecedent, and the antecedent with the merit value of greatest absolute value is determined to be the most meritorious. If that antecedent is askable and has not been previously asked, then it is asked. If the user answers the question, his response is propagated through the network to the top consequent, and the fact just answered by the user is marked answered in the current context. If the user decides to skip the question, the fact is marked asked in the current context. That fact is then expanded, and new merit values are calculated for all the antecedents of the fact. In either case, the system then repeats the process of finding the most meritorious question of those remaining.

Merit values allow the system to question the user about the battlefield situation in an intelligent manner. This algorithm may result in some skipping around the inference network, but always asks the most important questions first, and could save the user considerable time over a classical depth-first traversal of the inference network.

The Assignment Tree

In the BATTLE system, there is a set of inference networks and the top of each represents the effectiveness of one type of friendly weapon (e.g., 105mm artillery) against one type of target (e.g., an oil depot or a 122mm artillery unit). Targets may be composites of more than one type, e.g. an oil depot that is is in the same camp as a 155mm artillery unit. To calculate the effectiveness of one friendly unit against the composite target, we use a weighted average of the effectiveness of the friendly against each target component. To calculate the effectiveness of massing several friendly units against a single target, we take the complement of the product of the complements of the individual effectiveness numbers, corresponding to independent effectivenesses of the weapons.

Once an effectiveness value has been determined for each target that has been assigned in the planned assignment scheme, the total value for the assignment may be calculated. The effectiveness value for each target is first multiplied by the target value to produce the normalized value of the amount of destruction expected on that target. Finally, the normalized values for each of the targets are added together to produce a total effectiveness value for the entire assignment.

Presently, an assignment tree is used to explore all the possible assignment plans for the battlefield situation. Each level on the tree corresponds to the assignment of a different weapon, so that the number of levels on the tree is equal to the number of weapons in the situation. The degree of the tree is one greater than the total number of targets, with one branch for each target, corresponding to assigning the weapon to that target, and one other branch, corresponding to not assigning the weapon. At the bottom of the assignment tree all the possible assignments of friendlies to targets will be enumerated. Massing of friendlies on targets follows very naturally in this scheme, corresponding to taking the same branch down from a node for more than one weapon.

It should be apparent that the number of leaves of an assignment tree for w weapons and t targets is w^{t+1}. This is too large to traverse for realistic fire support situations, so we prune the tree during its traversal. For this purpose the fighting capacity values specified for each of the weapons is used. The fighting capacity is supposed to represent the maximum possible value to be gained by using a friendly. Once k complete plans have been found (where k is the number of plans the user has requested), if the value of a partial assignment plan plus the combined fighting capacities of the weapons to be used in completing the plan is less than the value of the kth best plan, then no attempt is made to complete the partial plan. This results in a substantial savings of time during the traversal. The weapons and targets are each ordered in decreasing value to maximize the amount of pruning.

The various assignment plans are rated according to this computed total value and the k best plans are displayed. The presentation of multiple plans allows the user to override the system's ranking of the options.

Future Improvements

We will be looking into using genetic algorithms[4] in place of an explicit assignment tree. Such algorithms may speed up the generation of assignment plans and are less domain-dependent. Also, we hope to extend the nature of facts from their current propositional form to permit predicates applied to variables. This would decrease the size of the inference network representation.

References

[1] Case, Kenneth E., and Thibault, Henry C., "A Heuristic Allocation Algorithm for Conventional Weapons for the Marine Integrated Fire and Air Support System", School of Industrial Engineering and Management, Oklahoma State University, Stillwater, Oklahoma, July 1976.

[2] Case, Kenneth E., and Thibault, Henry C., "A Heuristic Allocation Algorithm With Extensions for Conventional Weapons for the Marine Integrated Fire and Air Support System", School of Industrial Engineering and Management, Oklahoma State University, Stillwater, Oklahoma, September 1977.

[3] Duda, Richard O., Hart, Peter E., and Nilsson, Nils J., "Subjective Bayesian Methods for Rule-based Inference Systems", National Computer Conference, 1976; reprinted in Webber, Bonnie L. and Nilsson, Nils J., eds., *Readings in Artificial Intelligence*, Tioga Publishing Company, Palo Alto California, 1981.

[4] Holland, J.H., "Adaptation in Natural and Artificial Systems," University of Michigan Press, 1975.

[5] "Marine Tactical Command and Control Systems (MTACCS), Master Plan", Headquarters, U.S. Marine Corps, Washington, D.C., March 1981.

[6] Slagle, J. R. and Halpern, E. H., "An Intelligent Control Strategy for Expert Consultant Systems", NRL Memorandum Report 4789, Naval Research Laboratory, Washington, D.C., April 1982.

A ROUTE PLANNING AID FOR THE TACTICAL AIR FORCE C^3I SYSTEM

R. A. Riemenschneider and A. Joseph Rockmore

Systems Control Technology, Inc.
1801 Page Mill Road, Palo Alto, CA 94304

Abstract. Planning routes through heavily defended enemy territory is a complex task that requires the planner to integrate a variety of considerations into a high quality route. The Route Planning Aid is an interactive, user-friendly computer aid that assists the planner in determining his route. It combines three elements, each contributing significant factors to the final route. These elements are: optimization techniques that find minimum lethality routes based on threat exposure (including terrain masking) and evaluate numerically user-chosen routes; the user himself finding routes that have good properties vis-a-vis navigation; and a knowledge-based system to provide route explication. This paper concentrates on the knowledge-based system, describing both what it does and how it works.

INTRODUCTION

The Tactical Air Force (TAF) is responsible for the exercise of air power in a tactical environment. In the fulfilment of this responsibility, major effort is devoted to the command, control, communications, and intelligence (C^3I) functions associated with the TAF. C^3I processes tend to be complex and time consuming. As the number and variety of threats increase and C^3I systems become more sophisticated in the future, C^3I processes are bound to become even more complex, severely taxing already overburdened commanders.

Recognizing this problem, Rome Air Development Center has initiated major programs to design and build computer-based decision aids that will assist the TAF commander and his staff in their decision-making processes. This paper describes a part of this effort, focussing on one of the decision aids under construction, the Route Planning Aid (RPA). This aid assists the pilot or mission planner in selecting routes through heavily defended enemy territory on an interdiction mission.

The remainder of this paper describes the RPA with emphasis on the knowledge-based system portion of the aid. The next section describes route planning, and introduces some of the subtleties involved in the decision making process. Then the function of the aid is described, followed by details of the operation of the knowledge-based system. The paper concludes with the status of and plans for the RPA.

ROUTE PLANNING

A primary objective of the mission planner is to find the "best" route to and from a prespecified target. The best route is the route that maximizes the probability of accomplishing the goals of the mission. Successful completion of the mission requires penetration of defenses on ingress, striking the target, and evasion of defenses on egress. Therefore, the planner must consider both performance limitations of the aircraft (such as maximum range and navigation requirements) and threat exposure (to air interceptors, surface-to-air missile systems, anti-aircraft artillery, and even small arms fire) in the course of devising a mission plan.

This work was sponsored by Rome Air Development Center, Decision Aids Section, under contract F30602-81-C-0263 (Mr. J. Jacen, technical monitor). This support is gratefully acknowledged.

Currently, the planner proceeds roughly as follows. First, the area immediately surrounding the target is examined, using whatever maps and photographs of the area are available. An "attack axis" is chosen based upon target defenses and delivery tactics. The axis begins at the IP (initial point), a navigation update point that must be readily and precisely identifiable, and ends at the target. Choice of the IP is crucial because even a small navigation error at this stage will prevent accurate delivery of the weapon load. Next, the ingress route -- from the FEBA (Forward Edge of the Battle Area) to the IP -- is determined, working backward leg-by-leg. Then, the egress route -- from the target to the FEBA -- is determined, working forward this time. Finally, the FEBA approach and return to the recovery base from the FEBA are set. Each piece of the route is determined by an iterative process. For instance, several attack axes may be considered, one tentatively chosen, but later rejected due to difficulty of approach.

THE ROUTE PLANNING AID

The Route Planning Aid allows a number of improvements to be made in this process. Currently, only rough estimates of threat exposure are obtainable. Given the location of a threat, it is very difficult to determine whether an aircraft at low altitude is actually in line-of-sight at some point along the route from a topographical map alone. Yet terrain masking from threats is a vitally important factor in the planning process. RPA provides accurate terrain masking information. But the main virtue of the RPA is that it keeps all of the relevant information "in mind" and supplies it to the planner when appropriate. Although mission planners generally know as many or more planning heuristics as the RPA, it is only too easy to fail to consider all of them in the course of constructing of a complex plan.

The RPA contains two processing components and an executive for convenient user interface. One component consists of sophisticated algorithms that calculate the danger posed to an aircraft throughout a "mission space" specified by the user. Latitudes, longitudes, and altitudes that demarcate the boundaries are input by the user; threat locations and types, supplied by Intelligence, and digital terrain data for the specified space are then retrieved from storage. The algorithms also calculate minimum exposure routes between points in the mission space, using full dynamic programming techniques, and evaluate user-input routes with respect to various performance measures (most importantly, with respect to threat exposure).

The other component is a knowledge-based production rule system (KBS) capable of evaluating routes in light of heuristics supplied by route planning experts. This KBS is discussed in greater detail below.

The RPA is based on the conviction that each component -- algorithms, heuristic production rules, and user -- should concentrate on what it can do most effectively: the algorithms perform numerically-intensive calculations (of terrain masking effects, for example); the KBS infers route qualities and organizes information; and the user is freed so that his time can be devoted to tasks that are difficult or impossible to automate, such as evaluation of the quality of features to be used for navigation updates. Because the RPA aids, rather than replaces the planner, the user of the system is responsible for all final decisions regarding the route.

RPA OPERATION

The structure of the RPA is shown in Fig. 1. The Executive serves as an interface between the optimization and evaluation algorithms and the KBS, as well as a user interface. The purpose of the current effort is to demonstrate the feasibility of this general approach, so the present RPA is simpler than an operational system would be. In particular, the KBS would require many more production rules, and a much more comprehensive database. However, the present system is designed so that these extensions are straightforward: the basic structure would remain intact.

The operation of the KBS is illustrated in Fig. 2. Conceptually, the KBS consists of an event list, a collection of knowledge sources (sets of production rules), and a pair of databases. A processing cycle begins with a comparison of the event at the top of the event list to the precondition of each knowledge source (KS) in turn.

Preconditions P_i have the form: if EVENT is of type T, then activate KS_i. In case the KS is activated, the left-hand-sides of its production rules, the *conditions* C_i on the databases, are matched against the databases. The right-hand-sides are *actions* A_i to be taken if the condition is satisfied. An action consists of adding events to the event list and/or modifying the databases. This continues until the current event has been matched against every KS precondition. The current event is then removed from the event list and a new one is selected.

The KBS produces an explanation of a path chosen by either the user or by the optimization algorithms, as follows. First, files produced by the algorithms are transfered by the Executive to the KBS. These files contain the values of the performance measures calculated by the algorithms and the mission scenario. The path is given as a list of *critical points* along the path identified by the algorithms. Roughly speaking, a critical point is any point where "something happens" -- the heading changes, an important fixed feature is passed over, or the exposure status of some threat changes. Performance measures evaluated include the status of all threats at each critical point and the danger accumulated in passing from each critical point to the next. The scenario description is a list of threat locations and types, and the identity of the target.

From this information, a model of the route is constructed. Its structure is intended to be similar to that of the planner's own cognitive model of the route. This model consists of a network of *units*. A unit consists of a name, a list of attributes, and values for those attributes. The lowest level units represent the critical points, while higher levels represent route segments. A segment is a piece of the path, bounded by a pair of critical points, that has special significance. Segment creation is partly data-driven, partly model-driven, and partly based on a priori considerations.

This model serves as a basis for both explanation and criticism. Important functions of the explanation include convincing the user that the performance measures calculated by the algorithms deserve his confidence by providing a rationale in terms he is familiar with, and giving the user greater confidence in his chosen tactics. Criticism of the route can be based upon local or global features. The explanation and criticism are both rule-driven, and only created on demand.

The operation of the RPA accounts for the way routes are currently planned. The interaction with the system is menu-driven. A typical session would proceed as follows:

a) The user initializes the system by providing data concerning the mission, such as target identity, required time on target, weather conditions, and so on.

b) The user gets a threat briefing: a summary of the current state of the scenario via display, with additional information as requested by menu-driven database retrieval.

c) The user then focusses on the target area and picks an attack axis, entering it via joystick or latitude and longitude coordinates. Threat exposure in the target area is indicated by displayed threat contours and the location of the minimum lethality route. The user tentatively chooses an attack axis by picking the location of the IP, based on threat exposure and features that provide high quality IP's, and compares the chosen route with the minimum. If the threat exposure differs too greatly, one or more alternate attack axes might be considered, until the user is satisfied with his choice.

d) The user then reverts to considering the entire mission space to determine an ingress route. First, the minimal exposure route through the IP to the target is displayed (Fig. 3). Then, just as above, an ingress route is tentatively chosen, entered, and evaluated: an explanation of the route is produced (Fig. 4), particularly threatening legs may be examined in greater detail (Fig. 5), and a critique may be requested (Fig. 6). Based on the evaluation, alternate routes may be considered.

e) Finally, the egress route is determined in the same manner.

Of course, the user may become dissatisfied with early decisions at later stages of the process. In such cases, looping back and changing the decision is supported.

The present KBS contains about 50 rules organized into 15 knowledge sources. For instance, one very simple rule in the present system (contained in a KS activated by a heading change in the proximity of some threat) is:

If: the next heading change after POINT (the point where the current heading change occurs) is in the proximity of THREAT (the threat in proximity of POINT),
and: at no critical point between POINT and the next heading change is there any exposure to THREAT,
then: modify POINT to show that a possible reason for the heading change is avoidance of THREAT.

The static database characterizes about 12 threat types and 30 subsystems and associated systems, organized as a network of units.

CONCLUSIONS

The Route Planning Aid is still under development. It has powerful capabilities that could aid the pilot or mission planner in his route determination, but it also has some shortcomings. Currently, it is capable of determining minimum lethality routes through complicated scenarios, it is very friendly to the user, and it provides limited but significant explanations and criticisms. The primary shortcoming is that the explanations currently generated, which would be of use to an inexperienced user, tend to be rather obvious to an experienced user. This is especially true when the routes are chosen carefully, with attention to the manifold goals of the planner. In such cases the criticism of the route is often trivial; because there is nothing "wrong" with the route, there is no basis for criticism. For routes that are poorly chosen, the KBS guides the user in finding the portions of the route responsible for the low quality. However, there is still much to do in integrating the knowledge of different experts, thereby significantly improving the performance of the system. There are also a number of technical improvements being studied (efficiency of code, reduction of storage, additional graphic displays, and so on).

RPA has the potential of providing significant assistance to mission planners. It gives planners capabilities that they currently do not have, it takes advantage of the processing powers of the user and several technologies, and it is easy and efficient for a planner to use. The Route Planning Aid indicates the potential improvements in operation of the Tactical Air Force via knowledge-based decision aids.

Acknowledgements: We would like to acknowledge the invaluable contributions to this work by Scottie Brooks, Tom Wikman, and Jack Murphy of SCT, and by Capt. Dave Jewell of the 390 Tactical Fighter Squadron, Mountain Home AFB, ID.

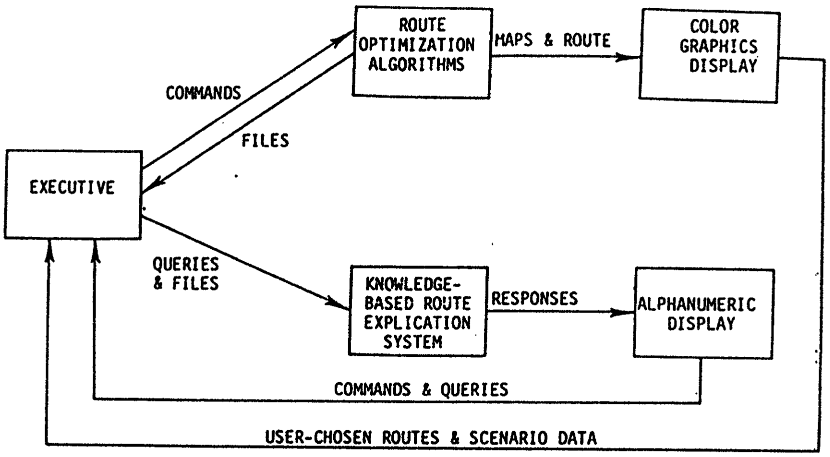


Figure 1. RPA Structure

THE DEMPSTER-SHAFER THEORY APPLIED TO TACTICAL DATA FUSION IN AN INFERENCE SYSTEM

R. A. Dillard

Naval Ocean Systems Center, San Diego, CA

Abstract. The Dempster-Shafer theory is applicable to tactical decision problems which can be formulated in terms of sets of exhaustive and mutually exclusive propositions. Dempster's combining procedure, a generalization of Bayesian inference, can be used to combine probability mass assignments supplied by independent bodies of evidence. This paper describes the use of Dempster's combining method and Shafer's representation framework in rule-based inference systems. It is shown that many kinds of data fusion problems can be represented in a way such that the constraints are met. Although computational problems remain to be solved, the theory should provide a versatile and consistent way of combining confidences for a large class of inferencing problems.

INTRODUCTION

In most applications of inference systems, estimates are computed of the degree of truth of the conclusions reached by the system. The medical system MYCIN, for example, employs formulas based on Bayes' Theorem and fuzzy set theory to produce a measure of belief that the patient has disease i in the light of certain pieces of evidence (Shortliffe, 1976). The mineral exploration system PROSPECTOR uses a similar scheme to assign probabilities concerning the composition of an ore deposit (Duda, Hart and Nilsson, 1976). A somewhat different approach involving the Shafer representation (Shafer, 1976) and Dempster's rule of combination (Dempster, 1967) is used in some recently introduced inference methods (Garvey, Lowrance and Fishler, 1981; Friedman, 1981).

In a tactical Navy application, the inference rules typically involve the association and identification of radar, ESM, and sonar contacts. When the problem is to accept or reject the hypothesis that a contact is a particular type of ship, many of the methods of propagating confidences are applicable. STAMMER2, a rule-based system developed at NOSC, uses incremental deduction formulas similar to those of MYCIN to decide whether or not a contact is a merchant, and also to conjecture about several other platform types (McCall and others, 1979; Bechtel, Morris and Kibler, 1981). Although these formulas are adequate for simple scenarios, they will not permit consistency to be maintained when deciding among more than a very few platform types. The best method of combining independent evidence in a consistent manner appears to be Dempster's approach, which is a generalization of Bayesian inference. This paper describes the Dempster and Shafer methods and discusses experimental implementations of the methods in STAMMER and in ROSIE (Fain and others, 1981), a Rule-Oriented System for Implementing Expertise.

THE DEMPSTER-SHAFER THEORY

A scheme for combining evidence which includes uncertainty or ignorance was devised by Dempster (1967) and later formulated within a flexible representation framework by Shafer (1976). In Shafer's representation, a "frame of discernment" on a domain is a set of propositions about the exclusive and exhaustive possibilities in the domain. The evidential interval [s (Ai), p (Ai)], a subinterval of the unit interval, may be used to represent the likelihood of Ai, the ith proposition (Garvey, Lowrance and Fischler, 1981). Here, s (Ai) represents the "support" for Ai and p (Ai) the "plausibility." The plausibility is the complement of the support for ~Ai and represents the degree to which one does not doubt Ai. (The symbol "~" is the Boolean NOT.) The "uncertainty" of Ai is u (Ai) = p (Ai) - s (Ai).

The representation involves the assignment of "probability masses" by knowledge sources. The mass allocated by knowledge source j to Ai is denoted mj (Ai). The evidential interval representing evidence about Ai contributed by the jth source is then [sj (Ai), pj (Ai)], where sj (Ai) = mj (Ai) and pj (Ai) = 1 - sj (~Ai). The probability masses contributed by various knowledge sources can be integrated by Dempster's rule to produce m (Ai), a combined probability mass of Ai.

Dempster's rule of combination requires that the knowledge sources be independent. (The same physical source, however, may contribute several pieces of sufficiently independent evidence; e.g., a radar can give measures of cross section, speed and location.) The combining operation is commutative and associative. The masses contributed by the various distinct knowledge sources can be combined in any order and in any combination of pairs, triples, etc.

Special Case

Dempster's rule is simple to implement for the special case where the various knowledge sources assign probability masses only to the propositions Ai and to uncertainty. Figure 1 shows the component masses for two knowledge sources. The mass assigned to θ represents mass assigned to uncertainty; it is assumed to be distributed in some unknown manner among the n propositions. Specifically, we define

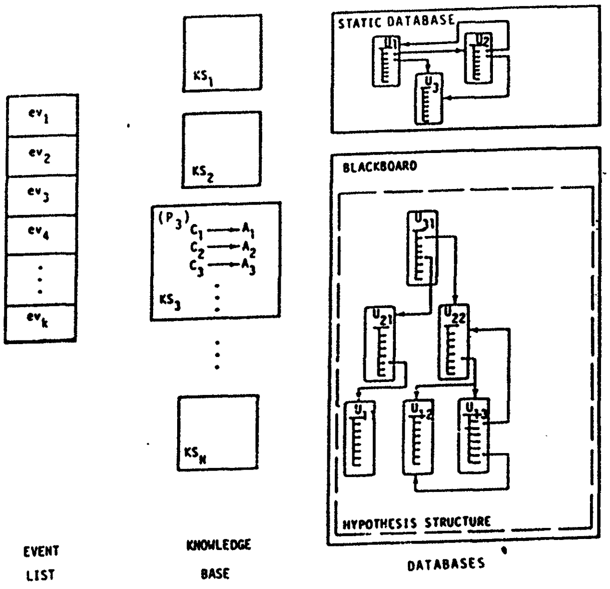


Figure 2. Knowledge-Based System

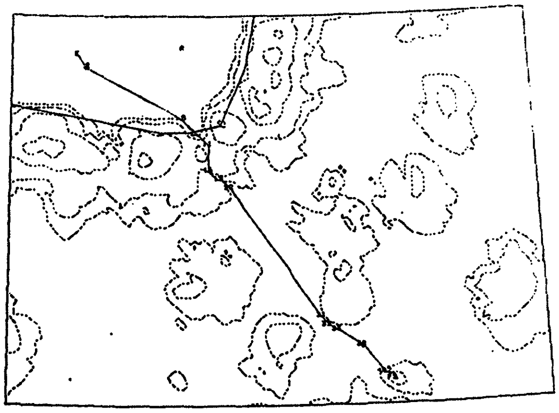


Figure 3. Route Display (normally in color)

```
                ROUTE EXPLANATION

               Total Lethality: 91

From     To       Time
Critical Critical (At Leg           Relative  Leg
Point    Point     End)   Duration Lethality Features

  1        5       21:20    8:41      16       LOW RISK
                                               LONG.LEG
  5        12      17:07    4:13      98       EXPOSED TO   1 SA-6
                                               EXPOSED TO   1 SA-2
                                               THREAT28
  12       20      12:59    4:08      78       EXPOSED TO   1 SA-2
  20       22      8:46     4:13      5        LOW RISK
  22       25      4:44     4:02      4        LOW RISK
  25       28      2:57     1:47      0        LOW RISK
                                               SHORT LEG
  28       33      :00      2:57      70       IP.POINT
                                               EXPOSED TO   1 SA-2
                                               EXPOSED TO   1
                                 SMALL BORE WEAPONS CONCENTRATION
                                               SHORT LEG
```

Figure 4. Route Explanation

```
             LEG FROM CRITICAL POINT 5 TO 12

               Duration: 4:13
               Time (at leg end): 17:07


               Heading on leg: 133.34
               Latitude at start of leg. 55.15
               Longitude at start of leg: 23.48
               Latitude at end of leg: 54.81
               Longitude at end of leg: 24.12


               THREAT EXPOSURE SUMMARY

               Threat #24  (of type SA-4)
               Threat #34  (of type SA-6)
               Threat #5   (of type SA-2)
               Threat #25  (of type SA-4)
               Threat #4   (of type SA-2)

Pt  Time Duration Lethality Tht  Status         Exposure
5   8:41    70         61    24  MASKED
                             34  EXPOSED           19
                             5   OUT.OF.RING
                             25  OUT.OF.RING
                             4   MASKED
6   9 51    18         78    24  MASKED
                             34  EXPOSED           43
                             5   IN.PROXIMITY
                             25  IN.PROXIMITY
                             4   MASKED
7   10 09   4          55    24  MASKED
                             34  EXPOSED           42
                             5   IN.PROXIMITY
                             25  MASKED
                             4   MASKED
8   10.13   48         89    24  MASKED
                             34  EXPOSED           42
                             5   MASKED
                             25  MASKED
                             4   MASKED
9   11:01   109        94    24  MASKED
                             34  EXPOSED           45
                             5   MASKED
                             25  MASKED
                             4   EXPOSED           57
10  12 50   0          9     24  MASKED
                             34  MASKED
                             5   MASKED
                             25  MASKED
                             4   EXPOSED           57
11  12 50   4          24    24  MASKED
                             34  IN.PROXIMITY
                             5   MASKED
                             25  MASKED
                             4   EXPOSED           57
```

Figure 5. Leg Details

```
                ROUTE CRITIQUE

              Total Lethality  91

From     To       Time
Critical Critical (At Leg           Relative  Leg
Point    Point     End)   Duration Lethality Features


  1        5       21:20    8.41      16       LOW RISK  -
                                               MASKED FROM
                                               THREAT34
                                               THREAT4
                                               THREAT24
  5        12      17:07    4:13      98
  12       20      12:59    4:08      78
  20       22      8:46     4:13      5        LOW RISK  -
                                               MASKED FROM
                                               THREAT28
                                               THREAT27
                                               THREAT8
                                               THREAT7
  22       25      4:44     4:02      4        LOW RISK  -
                                               MASKED FROM
                                               THREAT7
                                               THREAT8
                                               THREAT28
                                               THREAT27
  25       28      2:57     1:47      0        SHORT LEG  -
                                               HEAD FOR TARGET
                                               LOW RISK  -
                                               MASKED FROM
                                               THREAT7
                                               THREAT8
  28       33      :00      2:57      70
```

Figure 6. Route Critique

$\theta = A1vA2v \ldots vAn$

where "v" is the Boolean OR. (This definition differs from the usual one in which θ is the set of propositions $\{ A1, \ldots, An \}$.) The abscissa is a unit line segment partitioned into segments whose lengths are equal to ml (A1), ml (A2), . . . , ml (An), and ml (θ), respectively. Similarly, the ordinate represents the masses assigned by KS2. Each "rectangle" supports the proposition which is the conjunction of the propositions on the abscissa and ordinate. The crosshatched area represents mass associated with conjunctions of exclusive propositions.

For m knowledge sources, the combined probability mass for proposition Ai is

$$\begin{aligned} m(Ai) &= \{ \prod_{1 \leqslant j \leqslant m} [mj(Ai) + mj(\theta)] - \prod_{1 \leqslant j \leqslant m} mj(\theta) \} / C \\ &= F(Ai) \; / \; C \end{aligned} \tag{1}$$

where F(Ai) represents the expression in braces and

$$C = \prod_{1 \leqslant j \leqslant m} mj(\theta) + \sum_{1 \leqslant i \leqslant n} F(Ai). \tag{2}$$

The resulting uncertainty is

$$m(\theta) = \prod_{1 \leqslant j \leqslant m} mj(\theta) / C = 1 - \sum_{1 \leqslant i \leqslant n} m(Ai). \tag{3}$$

The resulting plausibility of Ai is

$$p(Ai) = 1 - \sum_{k \neq i} m(Ak) = m(Ai) + m(\theta). \tag{4}$$

For m = 2, the normalizing factor C represents the non-crosshatched area of Fig. 1; it restores the total probability mass to one. Note that when the knowledge sources contribute mass only to the propositions Ai and to θ, the uncertainty of every Ai is u(Ai) = p(Ai) - s(Ai) = m(θ).

General Case

While an analyst frequently will be able to express his degree of belief in individual propositions, more typically he will conclude from evidence "It's probably not a merchant" or "It's one of the ship types with a submarine-support function." The resulting mass assignments do not permit use of the special-case equations.

What we will call "general propositions" correspond to all possible subsets of the set $\{ A1, A2, \ldots, An \}$ of exclusive and exhaustive propositions. There will be $2^n - 1$ such general propositions. The proposition $\sim (A1vA2v \ldots vAn) = \sim \theta$ is not included, since it is in conflict with the assumption that the set $\{ A1, \ldots, An \}$ is exhaustive. A contribution to ~ Ai is treated as uncertainty mass spread in an unknown manner over the other n- 1 propositions; e.g., m(~A1) = m(A2vA2v . . . vAn).

The combined probability mass of the general proposition B is

$$m(B) = \sum_{B1 \& \ldots \& Bm = B} \; \prod_{1 \leqslant j \leqslant m} mj(Bj) \; / \; C \tag{5}$$

where Bj varies over the general propositions to which KSj assigns mass and

$$C = \sum_{B1 \& \ldots \& Bm \neq \sim \theta} \; \prod_{1 \leqslant j \leqslant m} mj(Bj). \tag{6}$$

The ampersand denotes the Boolean AND. Letting F(B) denote the numerator of Eq. 5, i.e., F(B) = C*m(B), we can write

$$C = \sum F(B) \tag{7}$$

where the sum is over all valid general propositions. The resulting support for a general proposition B is

$$s(B) = \sum_{B' \& B = B'} m(B') \tag{8}$$

Equation 8 is needed in the calculation of the plausibility of an original proposition Ai, using p(Ai) = 1- s(~Ai). Note that s(Ai) = m(Ai), as originally defined, and s(θ) = 1. Shafer (1976) and Barnett (1981) discuss the exact relationship between support functions and mass functions (in terms of "belief functions" and "basic probability assignments").

Dempster's combining operations reduce to standard Bayesian operations when uj(Ai) = 0 for every proposition Ai and every knowledge source j whose evidence is combined. The advantage of Dempster's method is that ignorance and uncertainty may be consistently modeled. There is no need, for example, to arbitrarily assign initial probabilities to each proposition before evidence is gathered. When evidence is available, any uncertainty involved in the measurement or its interpretation may be adequately represented.

DEMPSTER'S RULE APPLIED TO PLATFORM TYPING

A Frame of Discernment

In the application of platform identification, a "coarse" frame of discernment might be the propositions "platform x is a combatant" and "platform x is not a combatant," while the most refined frame would be the very large set of propositions of the form "platform x is <name of a specific platform>," a frame too difficult to completely formulate. In our examples, we will let a frame of discernment be the set of exclusive and exhaustive propositions { Ai: Platform x is type i }, where some types of lessor importance are lumped together to keep n relatively small.

A1:	carrier
A2:	cruiser
A3:	destroyer
A4:	frigate
A5:	amphibious
A6:	submarine (surfaced or periscope/snorkel/ antenna)
A7:	small fighting ship (e.g., Corvette)
A8:	fast attack/patrol craft
A9:	patrol craft
A10:	intelligence collector (e.g., AGI)
A11:	survey/research (navy operated)
A12:	fleet auxiliary – medium & large
A13:	fleet auxiliary – small
A14:	small boats (navy and commercial)
A15:	merchant

A16: fishing
A17: other commercial & private
A18: debris

We are assuming that there is one and only one platform under consideration; e.g., the radar blip is not just radar noise or clutter (although sometimes we will allow it to be the return from debris) and is not the return from two platforms close together. We will consider only surface and subsurface platforms and will disregard the possibility that a contact which appears to be a surface platform might be an aircraft. In some cases an "observed characteristic" of a platform will result from successive observations of a platform, and we will assume that the successive observations are indeed of the same platform.

An Application of the Special Case of Dempster's Rule

Initial probability masses based on the number of each type of platform, the region, and the political/military situation can be estimated. Arbitrarily, we will assume these masses to be

$$\{ ml(Ai) \} = \{ .003\ .013\ .04\ .03\ .04\ .02\ .04\ .06\ .02\ .015\ .012\ .09\ .03\ .01\ .13\ .07\ .02\ .007 \}$$

We consider first the simple case of a radar contact with an initial detection range and a single measurement of speed. To convert these measurements into probability masses, we employ a method similar to that used with emitter parameter distributions by Garvey, Lowrance and Fishler (1981). For each measurement and each ship type, a strip corresponding to the measurement ± the average measurement error is overlaid on the appropriate distribution curve (Fig. 2 illustrates this for range) and the overlapped area is computed. (Fig. 2 would be valid for only one antenna height, frequency and environment state, and assumes a low signal strength; a family of curves would be required.) These areas are normalized to sum to the complement of the probability mass of the uncertainty. In practice, the process could be automated by initially storing the mass vectors for incremented values of range and speed. Assuming an uncertainty mass of .3 for range and .4 for speed to represent our lack of confidence in the measurement and the distribution functions, we have the following values of mass for range and speed.

$$\{ m2(Ai) \} = \{ 0\ 0\ 0\ 0\ 0\ .037\ .045\ .136\ .136\ .027\ 0\ 0\ .04\ .055\ .023\ .077\ .064\ .055 \}$$

and

$$\{ m3(Ai) \} = \{ .103\ .103\ .103\ .029\ .008\ .009\ .029\ .074\ .009\ .008\ .008\ .008\ .008\ .011\ .034\ .011\ .045\ 0 \}.$$

Using Eqs. (1), (2) and (3) with m = 3, we have the combined masses

$$\{ m(Ai) \} = \{ .037\ .042\ .055\ .023\ .019\ .030\ .052\ .140\ .079\ .022\ .008\ .039\ .038\ .035\ .083\ .076\ .059\ .029 \}$$

and $m(\theta) = .138$.

Implementation in a Rule-Based System

A number of the "merchant rules" implemented in STAMMER2 (McCall and others, 1979) are of the following type.

- If the contact's speed < 9, it is somewhat unlikely to be a merchant.
- If the contact's initial detection range < 16 and the radar signal is weak, it is somewhat unlikely to be a merchant.

When a decision is to be made concerning a number of types, these should be replaced by more general rules which assign mass in a manner similar to that described for initial detection range. Many other rules from STAMMER2 can be used in their original simplicity; for example, the following.

1. If the contact changes course, it is probably not a merchant.
2. If the contact is not in a merchant lane, it is probably not a merchant.
3. If the contact's initial detection range < 8 and its radar signal is strong, it is somewhat likely to be a submarine which just surfaced.

Assume that the contact satisfies the conditions of the three rules above and that the contributions of mass are

From rule 1: $ml(\sim A15) = .3$, $ml(\theta) = .7$
From rule 2: $m2(\sim A15) = .25$, $m2(\theta) = .75$
From rule 3: $m3(A6) = .4$, $m3(\theta) = .6$.

Using Eqs. 5 through 8 results in the evidential inverval [.4, 1] for A6, [0, .315] for A15, and [0, .6] for the remaining propositions. (Computations are simpler if we combine the resultant of m1 and m2 with m3 rather than simultaneously combine the three.) Alternatively, we can derive these same results using a scheme devised by Barnett (1981) for efficiently implementing Dempster's rule. When each knowledge source contributes mass only to θ and to either Ai or ~Ai for only one value of i, i.e., when each confirms or denies just one of the n exclusive and exhaustive propositions, then Barnett's scheme may be used to reduce computation-time from exponential to linear.

TACTICAL PROBLEM FORMULATION

Examples of Mutually Exclusive Propositions

We conjecture that sets of nonexclusive propositions currently handled by tactical analysts can be reformulated into a set or sets of exclusive propositions. The ultimate default is to form a set of two propositions (A and ~A) out of each proposition in the initial set. The following are examples of how tactical problems may be expressed as sets of mutually exclusive propositions.

What is it? "The contact is type i." "The submarine is class i."

Who is it? "The contact is name i."

Whose is it? "The contact's country of origin/registration is nation i."

Which one did it? "Radar contact i emitted the intercepted signal." "Surface contact i launched the helicopter." (For both, an additional proposition is: None of the contacts did it.)

Which one is it? "Current contact i is earlier contact X." "Earlier contact i is current contact X." "Current contact i is platform X." (For all three, an additional proposition is: None of these contacts is it. Also, possible duplicates must be eliminated; i.e., contact j must be disallowed if it could be another detection of contact i).

Which partitioning is it? "Partitioning i of contacts is the correct partition." (Each partitioning is a collection of disjoint sets – each set representing a candidate track – whose union is the set of all contacts.)

Where is it? "The submarine is in sector i."

Is it or isn't it? (I.e., true or false?) "Contact X is the same platform as contact Y." "The contact is a merchant." "The contact is hostile." "The contact is a combatant." "The platform is preparing to attack." "The hostile submarine is in innocent passage."

We see from these sets of propositions that some questions about a contact are not independent of other questions. For example, associating a new contact with an earlier sighted platform whose type is known contributes to the confidence that the new contact is that type. Conversely, observing that the new contact has structural or behavior attributes consistent with the type of the earlier sighted platform increases the confidence in associating the two. In general, the computations of masses for two sets of propositions (in this example, the set for type and the set for contact association) frequently will share some of the same evidence, in which case the resulting mass assignments for one set must not be used in the computation of the other. The lesson is that the computation processes for two sets of propositions should share any evidence pertinent to both but should not then use each other's output.

A Contact Association Problem

We assume that detection data from the various sensors has already been preprocessed and correlated to the degree feasible with a multi-sensor correlator-tracker scheme. The contact association problem of primary interest here, however, is that where detections of contact X have ceased for a while, and in this situation an alternative to beginning with algorithmic correlation is to use simple inference rules which eliminate obviously impossible pairings of contacts, e.g., those requiring impossible speeds or in conflict because of a difference in observed type or class.

As an example of the "which one is it?" problem, suppose that exclusive propositions can be formulated about which current contact is the earlier contact X and that the type of contact X is known. Assume that the impossible pairs have been eliminated and "An" is the proposition that none of the remaining current contacts is contact X. One way of partitioning knowledge into knowledge sources which are sufficiently independent to combine using Dempster's rule is the following.

KS1:	Completeness of surveillance coverage
KS2:	Non-geolocation attributes of contact i (structural and miscellaneous attributes)
KS3:	Behavior attributes of contact i (speed and course, travelling in a merchant lane, in a storm, etc.)
KS4:	Relative positions (average speed and course from contact X position to contact i position; patrols that could have crossed path)

KS1 represents knowledge about the degree of surveillance coverage in the region and provides probability masses of An and θ . The knowledge sources KS2, KS3 and KS4 each represent the interpretation of combined pieces of raw data. Appropriate algorithms and combining methods are needed to generate the mass assignments m2, m3 and m4. (In a rule-based system, rules would control the assignment process.) The assignment of mass to θ by each should derive from the uncertainty of the correctness of the data and its interpretation.

Unless it is certain that one of the contacts is contact X, knowledge sources KS2, KS3 and KS4 should contribute mass to An. The mass mj(An) generally should be large if that knowledge source indicates that none of the contacts match contact X very well, judged by its evidence. The probability masses mj(Ai) for i = 1, . . ., n–1 are derived by normalizing the computed measures of match (based on KSj evidence) to sum to 1– mj(An) - mj(θ).

EXPERIMENTAL RESULTS AND CONCLUSIONS

Initial experiments in STAMMER and in ROSIE employed a set of rules which contribute confidences concerning platform type. Two of the rules assign masses according to the scheme outlined for initial detection range and speed. Algorithms for computing Dempster's rule were implemented for (1) the special case where mass is assigned only to the propositions Ai and to θ , (2) the combining of masses mj(~Ai) and mj(θ) for a single value of i, and (3) the combining of the results of (1) and (2). The combining was accomplished with procedure rulesets and function rulesets in ROSIE and with oracle-like functions in Interlisp code in STAMMER. Oracles are computation functions used in STAMMER rule conditions in much the same way as relational assertions. In this combining application, however, the oracles are called as actions.

In both systems, the mass assignments resulting from the firing of rules were stored in such a way that they could be selectively "recalled" if later information warrants. In the ROSIE version, the mass combining results are presented as "pro-con" pairs rather than as evidential intervals; i.e., the complement of the plausibility is presented rather than the plausibility. This form of presentation seems to convey more immediately a meaningful measure of evidence against a proposition.

A more complete discussion of these experiments can be found in Dillard (1982). Although computational limitations presently constrain us in the assignment of probability masses, we find the results encouraging. Future plans include experimenting with other kinds of tactical hypotheses, designing computational schemes for less constrained cases of Dempster's rule, and designing mechanisms for explaining the assignment and computing of confidences.

REFERENCES

Barnett, J.A. (1981). Computational methods for a mathematical theory of evidence. *Proc. IJCAI 7,* Vol. II, 868-875.

Bechtel, R., Morris, P. and D. Kibler. (1980). Incremental deduction in a real-time environment. *Proc. CSCSI-80.*

Dempster, A.P. (1967). Upper and lower probabilities induced by a multivalued mapping. *Annals of Math. Stat., 38,* 325-339.

Dillard, R.A. (1982). Computing probability masses in rule-based systems. NOSC Technical Document in Preparation.

Duda, R.O., Hart, P.E. and N.J. Nilsson (1976). Subjective Bayesian methods for rule-based inference systems. Technical Report 124, SRI, Menlo Park, Calif.

Fain, J., Gorlin, D., Hayes-Roth, F., Rosenschein, S., Sowizral, H. and D. Waterman (1981). The ROSIE language reference manual. Report N-1647-ARPA, The Rand Corporation, Santa Monica, Calif.

Friedman, L. (1981). Extended plausible inference. *Proc. IJCAI 7,* Vol. I, 487-495.

Garvey, T.D., Lowrance, J.D. and M.A. Fishler (1981). An inference technique for integrating knowledge from disparate sources. *Proc. IJCAI 7,* Vol. I, 319-325.

McCall, D.C., Morris, P.H., Kibler, D.F. and R.J. Bechtel (1979). STAMMER2: A production system for tactical situation assessment. Technical Document 298, Vols. 1 and 2, Naval Ocean Systems Center, San Diego, Calif.

Shafer, G. (1976). *A Mathematical Theory of Evidence.* Princeton University Press, Princeton, New Jersey.

Shortliffe, E.H. (1976). *Computer-Based Medical Consultations: MYCIN.* American Elsevier, New York.

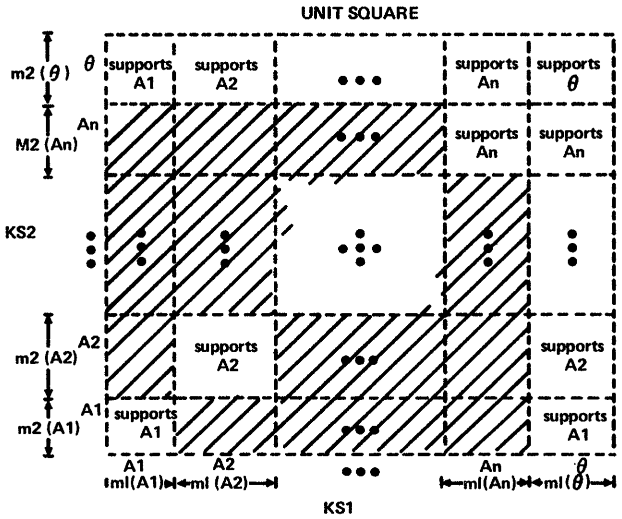


Fig. 1. Graphical representation for two knowledge sources, KS1 and KS2, and for n exhaustive and mutually exclusive propositions.

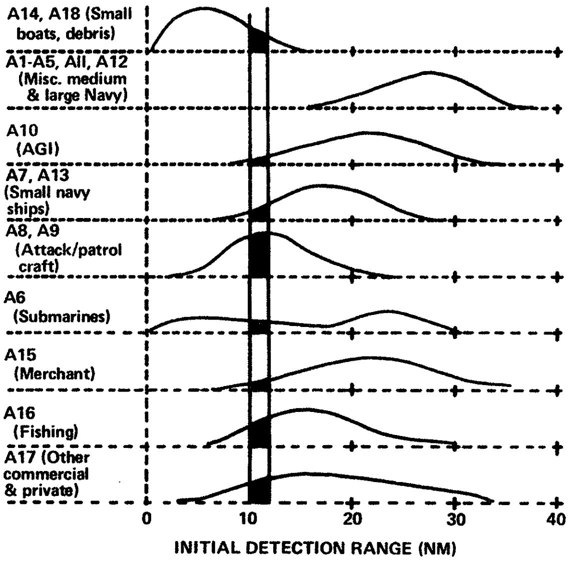


Fig. 2. Distribution functions for initial detection range.

RECENT ADVANCES IN DISTRIBUTED DETECTION*

Dr. G. S. Lauer
Dr. D. Teneketzis
Dr. D. A. Castanon
Dr. N. R. Sandell, Jr.

ALPHATECH, Inc., 3 New England Executive Park, Burlington, Massachusetts 01803

1. INTRODUCTION

This paper is a summary of the recent progress at ALPHATECH in the detection of signals and events using a distributed detection system. The basic distributed detection problem is illustrated in Figure 1. Several detectors take noisy measurements of the "world;" these measurements, may or may not include the presence of a signal. The purpose of the distributed detection system is to coordinate the processing done at each detector to optimize an overall system criterion. The results which we present in this paper study several variations of these problems. Sections 2 and 3 refer to problems where each detector obtains only a single measurement and processes this measurement into a decision concerning the presence of the signal. In Section 4, the detectors have a choice of taking more measurements or declaring whether the signal is present or not. In Section 5, the signal "arrives" at the world at a random time, and the detectors must establish its presence when it arrives. Section 6 discusses briefly some areas of future research. The work presented here is discussed in greater detail in references [1]-[6].

2. DISTRIBUTED DETECTION OF KNOWN SIGNALS IN NOISE

2.1 Problem Formulation

We assume that the i-th sensor has observations under the two hypotheses which are given by

$$\left.\begin{aligned} H^1:&\quad y_i(t) = \sqrt{E}_i s_i(t)+n_i(t) \quad (2\text{-}1a)\\ H^0:&\quad y_i(t) = n_i(t) \quad (2\text{-}1b)\end{aligned}\right\} T_o \leq t \leq T_f$$

The signals $s_i(t)$ are assumed to be known, to have unit energy and to be zero outside the interval $[0,T]$ where $T_o \leq 0 < T \leq T_f$. The $n_i(t)$ are assumed to be zero-mean Gaussian processes where

$$E\{n_i(t)n_k(\tau)\} = K_{ik}^n(t,\tau) \tag{2-2}$$

and we assume (to avoid singular detection)

$$K_{ii}^n(t,\tau) = K_i^c(t,\tau)+N_i^2\delta(t-\tau) \tag{2-3}$$

The objective is to minimize the global cost, i.e.

$$\min E\{J(u_1,u_2,H)\} \tag{2-4}$$

where the only decision rules allowed are of the form

$$u_i = \begin{cases} 0 & , H^0 \text{ is declared to have occured} \\ 1 & , H^1 \text{ is declared to have occured} \end{cases}$$

with u_i a function of $y_i(t)$ only.

The log likelihood ratio of hypothesis H, with respect to hypothesis H_o is given by

$$\ell_i = -\int_{T_o}^{T_f} \left(y_i(t) - \frac{1}{2}\sqrt{E}_i s_i(t)\right) g_i(t)dt \tag{2-5}$$

where

$$g_i(t) \triangleq \sqrt{E}_i \int_{T_o}^{T_f} Q_i(t,u)s_i(u)du \tag{2-6}$$

and $Q_i(t,u)$ satisfies

$$\int_{T_o}^{T_f} Q_i(u,v)K_{ii}^n(t,u)du = \delta(t-v) \tag{2-7}$$

Since ℓ_i is simply the log likelihood for a detection problem in which $y_i(t)$ is the only received observation, we denote the class of distributed decision laws

$$\ell_1 \underset{1}{\overset{0}{\gtrless}} \tau_1 \tag{2-8a}$$

$$\ell_2 \underset{1}{\overset{0}{\gtrless}} \tau_2 \tag{2-8b}$$

by the term distributed likelihood ration test (DLRT) laws.

In [1] we derive necessary conditions for the optimality of thresholds τ_1 and τ_2. These conditions take the form of two nonlinear coupled equations in τ_i. In general these equations have multiple solutions, in which case all must be evaluated to determine the optimal DLRT.

2.2 Examples

We now consider an example which illustrates the performance of a fusion center using optimal DLRT laws. We assume that each sensor send its local decision u_i to a fusion center where a global decision us is made. The fusion center law is defined as u = 1 if and only if $u_1 = u_2 = 1$. We display the performance of the fusion center via receiver operating characteristic (ROC) curves which plot probability of detection versus probability of false alarm. In addition to plotting the performance of the optimal DLRT we also plot the performance of the optimal centralized detection law. This allows us to determine how the performance of the detection system is degraded by requiring that only local decisions rather than sensor data be transmitted to the fusion center.

---

*This work was supported by AFOSR under Contract Number F49620-81-C-0015 and by DOE under Contract Number DOE/DE-AC01-804A50418.

We assume that the observations are given by

$$H^1: y_i(t) = \sqrt{2E_i}\,\sin(2\pi t)+n_i(t) \quad 0\leq t\leq 1 \qquad (2.9)$$

$$H^0: y_i(t) = \qquad n_i(t) \quad 0\leq t\leq 1 \qquad (2.10)$$

where $n_i(t)$ is zero-mean unit variance white Gaussian noise with

$$E\{n_1(t)n_2(\tau)\} = \rho\delta(t-\tau) \qquad (2.11)$$

and that

$$p^0 = p^1 = 1/2 \qquad (2.12)$$

Figures 2 through 4 illustrate the behavior, for asymetric sensors, of the centralized decision law, the optimal DLRT law and the locally optimal distributed law. In these plots $E_1 = 1$ and $E_2 = 4$. We see from Figure 2 that for the optimal DLRT the performance degrades as $\rho \to 1$.

Note however that, while the optimal DLRT no longer performs as well as the centralized law, it does perform better than the locally optimal DLRT and the difference increases as $\rho \to 1$. This is not surprising since the locally optimal law does not account for the correlation at all, while the optimal DLRT adjusts the thresholds as ρ changes.

3. DISTRIBUTED DETECTION OF UNKNOWN SIGNALS IN NOISE

3.1 Problem Formulation

We assume that the observations are as follows:

$$H^1: dy_i(t) = s_i(t)dt+dn_i(t) \quad 0\leq t\leq T \qquad (3\text{-}1a)$$

$$H^0: dy_i(t) = \qquad dn_i(t) \quad 0\leq t\leq T \qquad (3\text{-}1b)$$

where $s_i(t)$ is a sample path from a zero-mean Gaussian stochastic process with covariance function

$$E\{s_i(t)s_j(\tau)\} = K^s_{ij}(t,\tau) \quad , \; i,j = 1,2 \quad , \; 0\leq t,\tau\leq T \qquad (3\text{-}2)$$

and where $\dot{n}_i(t)$ are independent zero-mean white Gaussian noise processes with variances N_i^2. It is well known ([7], [8]) that the log likelihood ratio for detection of s_i by the i-th sensor is given by

$$\ell_i = \int_0^T \hat{s}_i(t)dy_i(t) - \frac{1}{2}\int_0^T \hat{s}_i^2(t)dt \qquad (3\text{-}3)$$

where the first integral on the right hand side is interpreted as an Itô integral and where $s_i(t)$ is the optimal estimate of $s_i(t)$ given the observation to time t

$$s_i(t) = E\{s_i(t)|y_i(\tau),\; 0\leq\tau<t \;,\; H^1\} \qquad (3\text{-}4)$$

The DLRT law thus is given by

$$\ell_i \underset{0}{\overset{1}{\gtrless}} \tau_i \qquad (3\text{-}5)$$

and the optimal DLRT law is generated by selecting the τ_i so as to minimize the expected cost. Unfortunately, since ℓ_i is not a linear function of the observations, ℓ_i is not Gaussian and in fact has a p.d.f. which is not readily computed. If, however, T is large compared to the inverse bandwidth of $s_i(t)$ then we expect that ℓ_i will be approximately Gaussian. This has been verified numerically for the scalar example of the next subsection*, subject to the restriction that the Gaussian approximation is <u>not</u> accurate in the tails of the distribution. Thus we <u>cannot</u> accurately compute probabilities of detection for <u>very</u> low (less than 1.0%) probability of false alarm constraints.

3.2 Example

Let us turn now to a simple example. We assume that the signal s(t) is a sample path from a first order Gauss-Markov process which has reached statistical equilibrium. Thus,

$$H^1: dy_i(t) = s(t)dt+dv_i(t) \qquad (3\text{-}6a)$$

$$H^0: dy_i(t) = \qquad dv_i(t) \qquad (3\text{-}6b)$$

where

$$s(t) = cx(t) \qquad (3\text{-}7)$$

$$dx(t) = -ax(t)dt+ 2ad\omega_i(t) \qquad (3\text{-}8)$$

where $\dot{\omega}_i(t)$ is zero-mean, unit variance white Gaussian noise independent of all $\dot{v}_i(t)$.

Figure 5 plots the probabiliy of miss, P_M, (i.e., u = 0 when H^1 true) as a function of signal energy and time-bandwidth product, when the probability of false alarm, P_F, (i.e., u = 1 when H^0 true) is fixed equal to o.1. Three curves are given on this plot. That labeled "two sensors (centralized)" corresponds to assuming that both $y_1(t)$ and $y_2(t)$ are available at the fusion center and are used in making an optimal decision. The energy required to maintain a given P_M increases as the bandwidth of the Kalman filter increases which allows more of the white noise power to affect the ℓ_i.†

The curve labeled "one sensor" corresponds to assuming that only $y_1(t)$ is available for making a decision. The effect of this, in the scalar problem, is to reduce by half the energy available for decisionmaking. Note that P_M increases with bandwidth, indicating an increased sensitivity of P_M with respect to energy as bandwidth increases.

The performance of the optimal DLRT law is given by the curve labeled "two sensors (distributed)". Note that the DLRT performance degrades with increasing time-bandwidth product with respect to the centralized two sensor case and improves with respect to the one sensor case. This indicates that the requirement of using a distributed law effectively decreases the energy on which a decision is based.

4. DISTRIBUTED DETECTION WITH THE OPTION OF MULTIPLE MEASUREMENTS

4.1 Problem Formulation

We consider two hypotheses H^0 (z=0) and H^1 (z=1) with a priori probabilities

$$\left.\begin{array}{l} p(H^0) = p \\ p(H^1) = 1-p \end{array}\right\} \qquad (4\text{-}1)$$

and two detectors.

We make the following assumptions:

1. There is no communication between the detectors.
2. The observations of the two detectors are independent conditioned on the hypothesis, i.e.,

$$p(y_1(t)\; y_2(t)|H) = p(y_1(t)|H)p(y_2(t)|H) \qquad (4\text{-}2)$$

where $y_i(t)$ represents a measurement of detector i (i=1, 2).

---

* That is, the Gaussian assumption yields ROC results which agree with those in [6].

† Time-diversity tradeoffs generally require increased energy for small enough time bandwidth products, however, for problems where the Gaussian assumption is reasonable the energy required to maintain a fixed P_M increases with bandwidth.

3. Conditioned on the hypothesis, the observations of each detector are independent, i.e.,

$$p\left(y_i(t)\ \ldots\ y_i(1)|H\right) = \prod_{j=1}^{t} p\left(y_i(j)|H\right) \qquad (4\text{-}3)$$

$(i=1,\ 2)$

4. Each detector is allowed to take observations and has to decide which hypothesis is true the latest by $t=N$.
5. The final decision u_i ($u_i=0$ or 1, $i=1,\ 2$) of each detector is based solely on his own information, i.e., if detector i decides to stop at time t, then

$$u_i(t) = \gamma_i\{y_i(1)\ \ldots\ y_i(t)\}\ ,\quad i=1,2 \qquad (4\text{-}4)$$

6. For each detector each additional observation after $t-1$ costs c.
7. If u_i ($i=1,\ 2$) are the decisions of the detectors and are made at times t_i ($i=1,\ 2$), then the cost incurred by these decisions is

$$ct_1+ct_2+J(u_1,u_2,H) \qquad (4\text{-}5)$$

The problem we want to solve is the following:

$$\left.\begin{array}{l} \underset{\substack{\gamma_1\in\Gamma_2 \\ \gamma_2\in\Gamma_2}}{\text{Minimize}}\ E\{ct_1+ct_2+J(u_1,u_2,H)\} \\ \\ \text{Subject to} \\ \\ \text{Eqs. 1 and 2, assumptions 1 - 7} \\ \text{and } \Gamma_i = \text{set of all stopping rules} \\ \text{which are measurable functions of} \\ \text{the data of detector } i \quad i=1,2 \end{array}\right\} (P) \quad (4\text{-}6)$$

The solution of the previous problem can be expressed as a stopping problem for each detector, where each detector compute his conditional probability that hypothesis 1 is time after k measurements, π_k^1. This probability is compared to two thresholds, α_k^1 and β_k^1 determined from the analysis of the cost of the various options, as illustrated in Figure 6. The optimal decision rule is:

If $\pi_k^1 < \alpha_k^1$ stop and choose H^1

If $\alpha_k^1 < \pi_k^1 < \beta_k^1$ continue

If $\beta_k^1 < \pi_k^1$ stop and choose H^0

The thresholds α_k^1 and β_k^1 at time $t=k$ are determined by the following equations:

$$\begin{aligned} &\alpha_k^1 \sum_{u_2^*} p(u_2^*|H^0)\ [ct_2(u_2^*) + J(1,u_2^*,H^0)] \\ &+ (1-\alpha_k^1) \sum_{u_2^*} p(u_2^*|H^1)\ [ct_2(u_2^*) + J(1,u_2^*,H^1)] \\ &= c + E_{y_1(k+1)}\left\{\bar{J}_{k+1}\left(\frac{\alpha_k^1\ p(y_1(k+1)|H^0)}{\alpha_k^1\ p(y_1(k+1)|H^0)+(1-\alpha_k^1)\ p(y_1(k+1)|H^1)}\right)\right\} \end{aligned} \qquad (4\text{-}7)$$

and

$$\begin{aligned} &\beta_k^1 \sum_{u_2^*} p(u_2^*|H^0)\ [ct_2(u_2^*) + J(0,u_2^*,H^0)] \\ &+ (1-\beta_k^1) \sum_{u_2^*} p(u_2^*|H^1)\ [ct_2(u_2^*) + J(0,u_2^*,H^1)] \\ &= c + E_{y_1(k+1)}\left\{\bar{J}_{k+1}\left(\frac{\beta_k^1\ p(y_1(k+1)|H^0)}{\beta_k^1\ p(y_1(k+1)|H^0)+(1-\beta_k^1)\ p(y_1(k+1)|H^1)}\right)\right\} \end{aligned} \qquad (4\text{-}8)$$

respectively.

Note that, in equations 4-8 and 4-9, the values of the decisions of the second detector are included. This decision depends on the thresholds used by the second detector, resulting in a coupled set of equations describing the 4N-2 thresholds required by both detectors. The important aspect of the solution in equations (4-6)-(4-8) is that each detector can summarize his decision in terms of the conditional probability that hypothesis 1 is true, given only his measurements. A detailed derivation of the results is given in [5].

5. DISTRIBUTED DETECTION OF DYNAMIC EVENTS

5.1 Problem Formulation

Consider a Markov chain $\{x_t,\ t=1,\ldots N\}$ with values in $\{0,\ 1\}$, whose transitions are described by:

$$\text{Prob}\ \{x_{t+1}=0\,|\,x_t=0\} = 1-p,\ p>0 \qquad (5\text{-}1)$$

$$\text{Prob}\ \{x_{t+1}=1\,|\,x_t=1\} = 1 \qquad (5\text{-}2)$$

and initial condition

$$\text{Prob}\ \{x_0=0\} = p_0 \qquad (5\text{-}3)$$

We make the following assumption:

$\{w_t^1,\ w_t^2,\ t=1,\ 2,\ldots\}$ is a collection of independent random variables which are also independent of the $\{x_t\}$ process.

Consider two detectors, and let the i-th detector's observation at time t b

$$y^i(t) = f^i(x_t,\ w_t^i),\ i=1,2 \qquad (5\text{-}4)$$

Let

$$\theta = \min_t\ \{t:\ x_t=1\} \qquad (5\text{-}5)$$

The two detectors do not communicate. Based on his own observations each detector declares a time τ_i at which he believes the Markov chain jumps from state 0 to state 1. After time τ_i detector i stops taking any further measurements. Since each detector uses only his own measurements to make a decision, the decision rules for the two detectors are measurable functions of their own data:

$$\tau_1 = \gamma_1(y^1(1)\ldots\ y^1(\tau_1)) \qquad (5\text{-}6)$$

$$\tau_2 = \gamma_2(y^2(1)\ldots\ y^2(\tau_2)) \qquad (5\text{-}7)$$

Let the cost associated with the decisions of the two detectors be $J(\gamma_1,\tau_2,\theta)$. In this paper we consider the following type of cost:

$$\begin{aligned} J(\tau_1,\tau_2,\theta) &= 1(\tau_1<\theta)\ 1(\tau_2<\theta) \\ &+ c\cdot(\tau_1-\theta)\ 1(\tau_1\geq\theta)\ 1(\tau_2<\theta) \\ &+ c\cdot(\tau_2-\theta)\ 1(\tau_2\geq\theta)\ 1(\tau_1<\theta) \\ &+ c\cdot(\tau_1-\theta+\tau_2-\theta)\ 1(\tau_1\geq\theta)\ 1(\tau_2\geq\theta) \end{aligned} \qquad (5\text{-}8)$$

This cost has the following interpretation. For both detectors all false alarms are uniformly penalized. For each detector the penalty for delays in detecting the time of the jump is proportional to the difference $\tau_1-\theta$ and $\tau_2-\theta$. Under these assumptions the decentralized quickest detection problem can be stated as follows:

$$\begin{array}{l} \text{Minimize } EJ(\tau_1, \tau_2, \theta) \\ \gamma_1 \varepsilon \Gamma_1 \\ \\ \gamma_2 \varepsilon \Gamma_2 \end{array} \tag{Q}$$

Subject to (3.1) - (3.8), (A2) and

Γ_i = set of all measurable functions of the data of detector i (i=1,2)

The solution of this problem was developed in [6]. The main result in [6] is

Theorem

At each instant of time t, a member-by-member optimal policy of detector 1 can be described as follows:

$$\text{Stop If } \pi_t^1 \geq \alpha_t^1(\gamma_2^*) .$$

$$\text{Continue Otherwise}$$

The threshold α_t^1 of detector 1 at time t is coupled with the thresholds of $\{\alpha_\tau^2\}_{\tau\varepsilon N}$ of detector 2 via a nonlinear algebraic equation of the form

$$(1-\alpha_t^1)\text{Prob}(\theta>\tau_2^*|\theta>t) = c\alpha_t^1 + E_{y^1(t+1)}\{G(f(\alpha_t^1 , u_t^1))|\mathcal{F}_1^t\} , \tag{5-9}$$

Where $E_{y^1(t+1)}\{\tilde{G}(f(\alpha_t^1, u_t^1 = u^c))|F_1^t\}$ is given by

$$E_{y^1(t+1)}(\tilde{G}(f(\pi_t^1 , u_t^1))|\mathcal{F}_1^t) =$$

$$= E_{y^1(t+1)}\min((1-\pi_{t+1}^1)\text{Prob}(\theta>\tau_2^*|\theta>t+1), c\pi_{t+1}^1 + E_{y^1(t+2)}\min((1-\pi_{t+2}^1)$$

$$\times \text{Prob}(\theta>\tau_2^*|\theta>t+2), c\pi_{t+2}^1+E_{y^1(t+3)}\min((1-\pi_{t+3}^1)\text{Prob}(\theta>\tau_2^*|\theta>t+3) ,$$

$$c\pi_{t+3}^1+E_{y^1(t+4)}\min((1-\pi_{t+4}^1)\text{Prob}(\theta>\tau_2^*|\theta>t+4), c\pi_{t+4}^1+E_{y^1(t+5)}\min(\ldots$$

$$\ldots)\ldots |\mathcal{F}_1^{t+2})|\mathcal{F}_1^{t+1})$$

Notice that, whereas in the centralized quickest detection problem the threshold is time invariant, in the decentralized case the thresholds are in general time-varying. To understand why this happens, let us examine 5-9 closely. In order for detector 1 to compute his threshold, it must take into account the probabilities of when he expects the other detector to decide. These probabilities are time-varying, and so are the thresholds.

Computation of the thresholds described in the Theorem is a very difficult problem. The equations for the thresholds of each detector are coupled across time and with the other detector's.

We are currently investigating feasible ways of computing approximate solutions for thresholds in 5-9.

6. CONCLUSION

In this paper, we have provided an overview on how to optimally design a distributed detection system for three classes of problems: Problems with static events, observed once; static events, observed with many observations, and dynamic events, observed with many observations The solutions to these problems are increasingly complex. For the problems in Sections 2 and 3, we were able to compute solutions and evaluate their performance. In Section 4, the solution was described by a finite set of coupled algebraic equations, and no performance comparisons were possible. In Section 5, the solution was described by an infinite set of coupled equations. It is clear that, in the more complicated cases, suboptimal solutions will be required for implementation.

REFERENCES

[1] Tenney, R. R. and N. R. Sandell, Jr., "Detection with Distributed Sensors," IEEE Transactions on Aerospace and Electronic Systems, Volume AES-17, Number 4, 1981, pp. 501-509

[2] Lauer, G. S. and N. R. Sandell, Jr., "Decentralized Detection Given Waveform Observations," TP-122, ALPHATECH, Inc., 3 New England Executive Park, Burlington, Massachusetts, Feb. 1982

[3] Lauer, G. S. and N. R. Sandell, Jr., "Distributed Detection of Known Signals in Correlated Noise," TP-131, ALPHATECH, Inc., 3 New England Executive Park, Burlington, Massachusetts, March 1982

[4] Lauer, G. S., "Static Fusion Center Design for Known Signals in Noise," TP-134, ALPHATECH, Inc., 3 New England Executive Park, Burlington, Massachusetts, April 1982

[5] Teneketzis, D., "The Decentralized Wald Problem," Internal Report, ALPHATECH, INC., 3 New England Executive Park, Burlington, Massachusetts, February 1982

[6] Teneketzis, D., "The Decentralized Quickest Detection Problem," TP-124, ALPHATECH, Inc., 3 New England Executive Park, Burlington, Massachusetts, March 1982

[7] Van Trees, H. L., Detection, Estimation and Modulation Theory, Part I, J. Wiley and Sons, New York, 1971

[8] Kailath, T., "A General Likelihood-Ratio Formula for Random Signals in Gaussian Noise," IEEE Transactions on Information Theory, Volume IT-15, Number 3, 1969, pp. 350-361

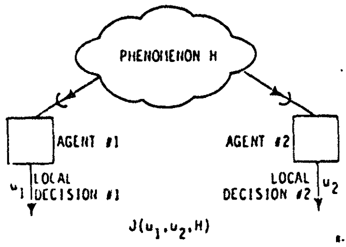


Figure 1. Distributed Decisionmaking.

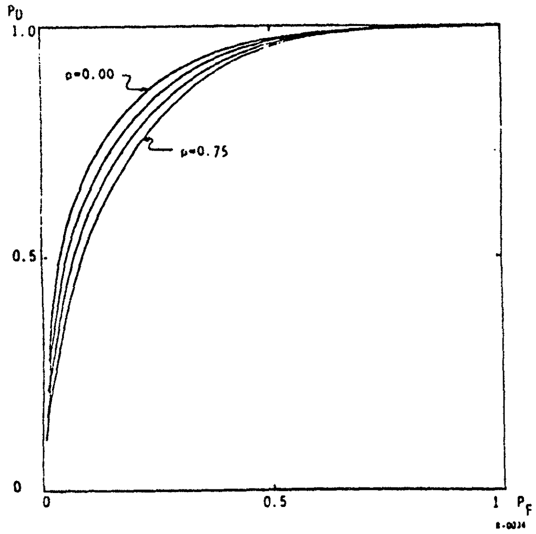


Figure 2. DLRT ROC Curves for ρ=0.0, 0.25, 0.50, and 0.75 with E_1=1, E_2=4.

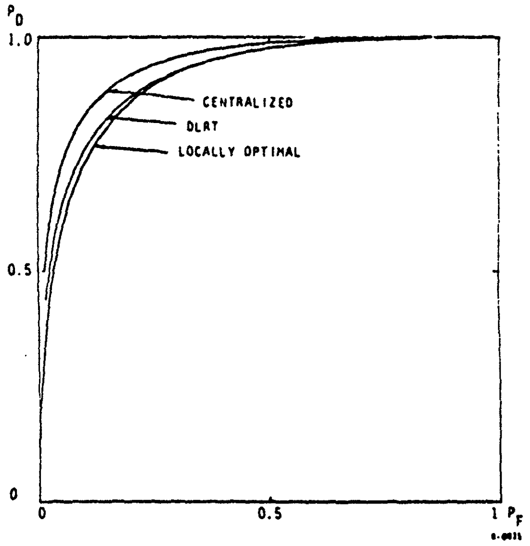


Figure 3. Centralized DLRT and Locally Optimal ROC Curves for $E_1=1$, $E_2=4$ and $\rho=0.0$.

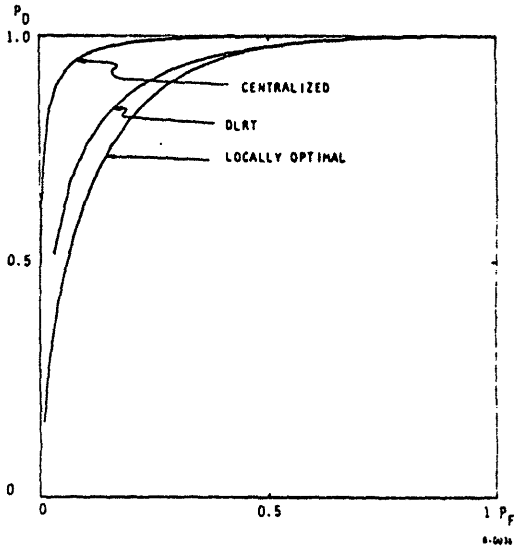


Figure 4. Centralized DLRT and Locally Optimal ROC Curves for $E_1=1$, $E_2=4$ and $\rho=0.5$.

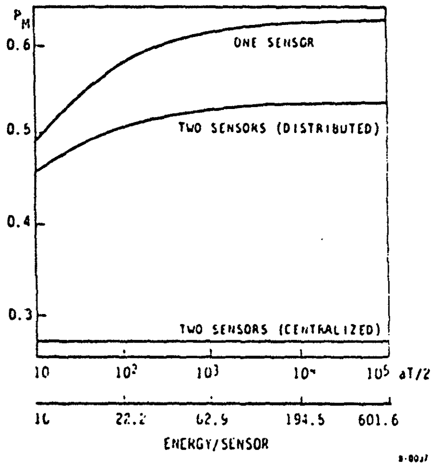


Figure 5. P_M for Various Decision Laws with $P_F=0.1$.

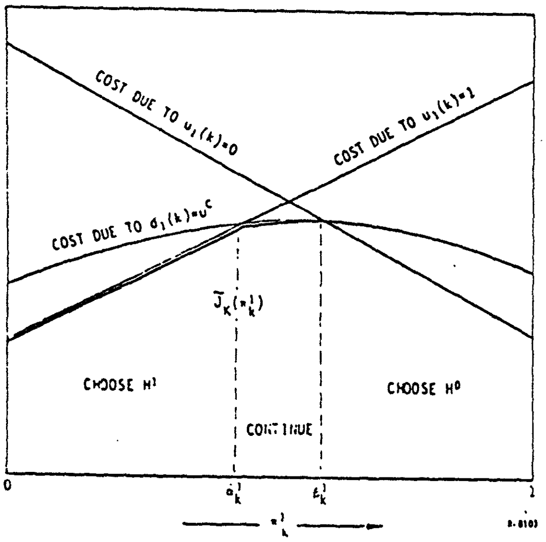


Figure 6.

DETECTION AND COMMUNICATION NETWORKS

L.K. Ekchian and R.R. Tenney

MIT/LIDS
Rm. 35-409

MIT/LIDS
Rm.35-213

Abstract. Two example problems involving joint detection and communication are solved. The structure of the decision rules is a clear generalization of the centralized and noncommunicating, decentralized cases.

OVERVIEW

This paper elaborates on the decentralized detection problems introduced in [1]. A similar effort is in Tenney and Sandell [2], where the authors show that even extending the simple centralized binary hypothesis testing problem is a fairly involved procedure with various surprising results that arise due to the coupling between the two detectors. The binary hypothesis testing problem there served to motivate our examination of the sequence of examples that are presented in this chapter.

A system is comprised of two detectors monitoring a phenomenon $H \in \{H^o, H^1\}$, by each taking an (independent) measurement. The critical result is that when the measurements y_1 and y_2 are statistically independent, the detectors each involve local likelihood ratio test whereby the respective thresholds are determined via a coupled computation. Note that the threshold structure of the decision policies has carried over from the centralized case to this special decentralized one. We shall see in the analysis of the sequence of examples in this paper that the structure carries over to a wide class of binary hypothesis testing networks.

TWO TANDEM DECENTRALIZED SENSORS

Two hypotheses are H^o and H^1 with a priori probabilities

$$p(H^o) = p^o \qquad p(H^1) = p^1$$

Conditioned on each hypothesis, the sensor observations y_1 and y_2 have distributions

$$p(y_1|y_2,H) = p(y_1|H)$$
$$p(y_2|y_1,H) = p(y_2|H)$$

Each agent D1 or D2 makes a decision $u_1 \in \{u_1^o, u_1^1\}$.

The "upstream" decision u_2 is available to agent 1 through a communication channel. The selection of u_2 seeks to minimize a global cost function $J(u_1,H)$ which is not a function of u_2. u_2 is only an intermediate decision variable whose sole purpose is to aid D1 in making the "best" possible decision u_1.

Moreover, in the terminology of team theory the information structure of the tandem organization is nonclassical because the whole signal y_2 is not passed on to D1. Instead a 1-bit encoded description of y_2 (i.e. u_2) is transmitted. The optimal local decision rule for D1 is given by the likelihood ratio

$$\frac{p(y_1|H^o)}{p(y_1|H^1)} \overset{u_1=0}{\underset{u_1=1}{\gtrless}} \frac{p(H^1)p(u_2|H^1)[J(0,H^1)-J(1,H^1)]}{p(H^o)p(u_2|H^o)[J(1,H^o)-J(0,H^o)]}$$

$$\triangleq \begin{cases} t_1^o & \text{if } u_2 = 0 \\ t_1^1 & \text{if } u_2 = 1 \end{cases}$$

while the decision rule for D2 is given by

$$\frac{p(y_2|H^o)}{p(y_2|H^1)} \overset{u_2=0}{\underset{u_2=1}{\gtrless}} \frac{\sum_{u_1} p(H^1)J(u_1,H^1)}{\sum_{u_1} p(H^o)J(u_1,H^o)}$$

$$\frac{[p(u_1|u_2=0,H^1)-p(u_1|u_2=1,H^1)]}{[p(u_1|u_2=1,H^o)-p(u_1|u_2=0,H^o)]}$$

$$\triangleq t_2$$

The reader is referred to [3] for details. Moreover, we note in above equation that

$$\ell r(y_2) \underset{u_2=1}{\overset{u_2=0}{\gtrless}} t_2 \overset{\Delta}{=} f_2(\gamma_1(\cdot))$$

where $\gamma_1(\cdot)$ is the decision rule for D1. **Similarly**, there are two simultaneous equations relating t_1^1 to $\gamma_2(\cdot)$ and t_1^o to $\gamma_2(\cdot)$, where $\gamma_2(\cdot)$ is the decision rule for D2.

HIERARCHICAL STRUCTURES: FUSION CENTER

In this section we extend the results of the previous sections to obtain the optimal decision policy for a hierarchical fusion structure. As in the above section, the two hypotheses are H^o and H^1 with apriori probabilities

$$p(H^o) = p^o \qquad p(H^1) = p^1$$

Conditioned on each hypothesis, the sensor observation y_o, y_1 and y_2 have joint probability distributions which are assumed to be statistically independent:

$$p(y_o|y_1,y_2,H) = p(y_o|H)$$

$$p(y_1|y_o,y_2,H) = p(y_1|H)$$

$$p(y_2|y_o,y_1,H) = p(y_2|H)$$

Each decision-maker has to make a decision u_1, u_2 or u_3, based on available <u>local</u> information. D1 generates the decision $u_1 \in \{0,1\}$ based on y_1 to minimize the penalty function $J(u_o,H)$ given by

$$J:\{0,1\} \times \{H^o,H^1\} \to R$$

Similarly D_2 generates the decision $u_2 \in \{0,1\}$ based on y_2 to minimize the same system penalty function $J(u_o,H)$. However, D_o's information set is $\{u_1,u_2,y_o\}$ with $J(u_o,H)$ the function to be minimized by the selection of u_o.

Note that the structure of the tandem detectors of the above section is embedded in the this problem also. In particular, the latter is comprised of two parallel tandem configurations, where the downstream detector (D_1) is in common. Here, the decision rules are also coupled likelihood ratio tests. In particular, the thresholds are coupled to the two thresholds t_1 and t_2. The optimal decision rules are:

$$\frac{p(y_1|H^o)}{p(y_1|H^1)} \underset{u_1=1}{\overset{u_1=0}{\gtrless}} \frac{\sum_{u_o} p(H^1)j(u_o,H^1)[p(u_o|u_1=0,y_o)-p(y_o|u_1=1,y_o)]}{\sum_{u_o} p(H^o)J(u_o,H^o)[p(u_o|u_1=1,y_o)-p(u_o|u_1=0,y_o)]}$$

$$\overset{\Delta}{=} t_1$$

and

$$p(y_2|H^o) \underset{u_1=1}{\overset{u_2=0}{\gtrless}} \frac{\sum_{u_o} p(H^1)J(u_o,H^1)}{\sum_{u_o} p(H^o)J(u_o,H^o)}$$

$$\frac{[p(u_o|u_2=0,y_o) - p(u_o|u_2=1,y_o)]}{[p(u_o|u_2=1,y_o)-p(u_o|u_2=0,y_o)]}$$

$$\overset{\Delta}{=} t_2$$

and, for the fusion center itself

$$\frac{p(y_1|H^o)}{p(y_1|H^1)} \underset{u_1=1}{\overset{u_1=0}{\gtrless}} \frac{p(H^1)p(u_2,u_3|H^1)}{p(H^o)p(u_2,u_3|H^o)} \cdot$$

$$\cdot \frac{[J(0,H^1)-J(1,H^1)]}{[J(1,H^o)-J(0,H^o)]}$$

$$\overset{\Delta}{=} \begin{cases} t_1^{00} & \text{if } u_2 = 0, \quad u_3 = 0 \\ t^{01} & \text{if } u_2 = 0, \quad u_3 = 1 \\ t_1^{10} & \text{if } u_2 = 1, \quad u_3 = 0 \\ t_1^{11} & \text{if } u_2 = 1, \quad u_3 = 1 \end{cases}$$

Hence the decision structure is defined by six parameters $\{t_0^{00}, t_0^{lo}, t_1^{ol}, t_o^{11}, t_1, t_2\}$ which are tightly coupled by the three decision rules.

CONCLUSION

Along with the examples discussed thus far a host of other problems have been studied. The goal has been to explore these problems and extract fundamental underlying issues, (e.g. general structures for decision rules, performance measures, organization, decomposition structure) to aid the designers of large C^3 systems to better understand the complex architecture, control and processing problems they face.

REFERENCES

[1] M. Athans (ed),"Proceedings of the Second MIT/ONR Workshop on Distributed Communication and Decision Problems Motivated by Naval C^3 Systems", Vol. II, LIDS-R-967, M.I.T., March 1980.

[2] R.R. Tenney and N.R. Sandell, Jr., "Detection with Distributed Sensors", IEEE Trans. on Aerospace and Electronic Systems, Vol. AES-17-No.4, pp. 501-510, 1981.

[3] L.K. Ekchian, "Control-Communication Issues in Distributed Surveillance Systems", Ph.D. thesis, MIT EECS, Oct. 1981.

ESTIMATION AND CONTROL APPROACHES TO SENSOR CONTROL IN C^3S SYSTEMS

V. S. Samant

ORINCON Corporation, 3366 North Torrey Pines Court, La Jolla, California, USA

Abstract. In this paper the issues related to the sensor resource allocation function of the command and control (C^2) process are investigated. Useful techniques and results in the estimation and control theory are explored and modified to define mathematical models for the sensor control problem. These techniques are then used to construct algorithms which are useful in real-time systems. In particular, laws for continuous-time evolution of the *a posteriori* probability density function for the state of the system are derived under the "null" report from the sensor system. Numerical results are given to show the effects of the "null" information on the *a posteriori* density function. For practical implementation a discrete version of these rules of evolution are derived. It is shown that for a specific class of exponential detection functions the discrete evolution of the density function can be conveniently computed from a bank of Kalman filters. A stochastic control problem is then defined for the purpose of solving the sensor control problem. Cost functions related to the detection probability are discussed. It is shown that a separation control policy can be devised to allocate sensor resources optimally for a class of quadratic and exponential quadratic cost functions.

INTRODUCTION

Control of sensor systems in a dynamic environment is one of the major functions of the command and control (C^2) process in a total command/control/communications and surveillance (C^3/S) system. In generic terms, the sensor control problem is that of optimizing the actions of controllable sensor systems which are available to the C^3/S system. This optimization is performed using the information given to the C^2 process by the surveillance process. Thus, the sensor control function constitutes an important link between the C^2 process and the surveillance process.

The objective of this paper is to bring forth the techniques of estimation and control theory which can play a role in solving this vital problem.

Among the information collection functions of the surveillance process are those to use the sensors to search a desired area for targets, to detect and acquire targets, and to track acquired targets. In the search mode the sensor systems are given a vague description of the target states; the controlled sensor systems have not detected the target yet, and the sensor controls are generated by the C^2 process using the "null" information to optimize the actions of the sensor system. The "null" information is given to the C^2 process in the form of a report which states that the control actions dictated by the C^2 process resulted in a failure to detect a signal. The detection mode is used to transition the control actions from the search mode to the track mode. In the track mode the sensor systems generate positive reports in the form of measurements which are functionally related to the state of the system.

The objectives of the control are different in the search and track modes. In the search mode the C^2 process opts to optimize sensor configurations to obtain a first detection. In the track mode it optimizes the sensor configuration to avoid a first missed detection. Therefore, ideally, the system desires to minimize the time of first detection or to maximize the time for first failure to detect a tracked target. Unfortunately, these optimal times are not easily computed because of a lack of computational structures for these times. Therefore, other suitably formulated and tractable measures of performance are used by the C^2 process in obtaining sensor control strategies. In the search mode, the detection probability is one such measure. In the track mode, estimation accuracies are used as a measure of performance.

In the sections which follow, we present a quantitative formulation of the above issues. The models used are for a continuous state space and a continuous control space. Extension to discrete state spaces are conceptually similar, but will differ in details. Both continuous and discrete time models are considered. Only the optimization relevant to the search problem is considered.

This research was supported by DARPA Contract No. N00039-82-C-0197.

SEARCH, DETECTION AND MEASURES OF PERFORMANCE

Let $x(t)$ denote the state of a target. The dynamics of such a target can be adequately described by a stochastic differential equation:

$$dx(t) = f(x(t),t)dt + g(x(t),t)d\beta(t) \qquad (1)$$

where $\beta(t)$ is an independent increment process defining the noise. This analysis is restricted to either a Wiener process or a generalized Poisson jump process or a sum of these two processes as the noise processes in Eq. (1). Then under fairly non-restrictive hypotheses on Eq. (1), the solution to the above stochastic differential equation, $x(t)$, is a Markov process. For obvious reasons, the discussion in this paper is restricted to Markov processes.

Let I_{t,t_0} denote all the information available at time t. This information is collected by M sensors in the system. Let $y_\alpha(t)$ denote the state of the α-th sensor. The sensor generates two types of reports:

i) A positive report is given when the sensor detects the target. Over the time interval $[0,t]$ a sensor may detect the target several times. $N_\alpha(t)$ denotes the number of detections reported by the α-th sensor over the interval $[0,t]$. Some types of sensors also provide further information about the target. Let $z_\alpha(t)$ denote the measurement generated by the α-th sensor when it detects a target. $\underline{N}(t)$ and $\underline{z}(t)$ will be the composite information in the form of vector processes.

ii) A negative report at time t is one in which no detections are recorded by the α-th sensor over the interval $[0,t]$; i.e., $N_\alpha(t) = 0$. Therefore, there is no $z_\alpha(t)$ associated in this case.

For the purpose of analysis, the information I_{t,t_0} used in computing the a posteriori transition probability density function $p(x(t),t|x(t_0),t_0,I_{t,t_0})$ is equivalently described by the sub-σ-field $\mathscr{G}^t_{t_0}$ generated by $\underline{N}(t)$ and $\underline{z}(t)$ over the interval $[t_0,t]$. For notational convenience, when $t_0 = 0$, we will denote $\mathscr{G}^t_{t_0}$ simply by $\mathscr{G}^t$.

In the search mode the optimization problem is to compute a search policy $Y_t = \{y(\tau), 0 \le \tau \le t\}$ for each sensor in the system to maximize the detection performance of the overall system. The trajectory of the α-th searcher is described by the controlled stochastic differential equation:

$$dy_\alpha(t) = g_\alpha(y_\alpha,u_\alpha,t)dt + dn_\alpha(t) \qquad (2)$$

where $dn_\alpha(t)$ describes the noise process. The detection performance of a system of searchers described by (2) and a target described by (1) can be defined using Koopmans' laws of detection. If instantaneous probability of detection during the time interval $[t,t+\Delta t]$ is given by $\lambda(x(t),y_\alpha(t))\Delta t$ for the α-th searcher, then the probability of detecting a target at time t for the first time is given by

$$P_D(t) = 1-\exp\left\{-\int_{t_0}^{t} d\tau \sum_\alpha \int_X p_o(x,\tau|x_o, t_0, \mathscr{G}^t_{t_0})\,\lambda(x,y_\alpha(\tau))dx\right\} \qquad (3)$$

where $p_o(x,\tau|x_o,t_0, \mathscr{G}^t_{t_0})$ denotes the transition probability density function for target state from (x_o,t_0) to (x,τ) given that the search during (t_0,t) was unsuccessful. The purpose of the C^2 process is to compute an optimal control policy $\{u^*(s),t_0 \le s \le t_0+T\}$ over a finite interval $[t_0,t_0+T]$ to maximize the detection probability in (3). Practical solutions to this control problem are difficult and to derive them is not the objective of this paper. An important quantity of interest in designing useful control policies is the transition density $p_o(x,t|\ \mathscr{G}^t_{t_0})$ itself. One of the objectives of this paper is to derive the rules of evolution for the above a posteriori transition probability density function. A partial solution to a single target/single sensor problem was first given in Hellman. In this solution, effects of only the "null" information (i.e., $N_\alpha(t) = 0$) were considered for a single sensor. The derivation given here is much simpler and more general.

The integro-partial differential equations given in this paper result from applications of a sequence of well-known results for Markov processes. Results on both continuous and jump processes are used. These results are stated without proof.

There are M searchers surveilling the area. During the time interval $[t,t+\Delta t]$ the α-th searcher counts the number of detections $dN_\alpha(t)$. The probability that the α-th searcher detects the source during $[t,t+\Delta t]$ is given by $\lambda^*_\alpha(x(t), y_\alpha(t))dt$ where $x(t)$ is the state of the source and $y_\alpha(t)$ is the state of the α-th sensor at time t. Let

$$\lambda^*(x(t),y(t)) = \begin{bmatrix} \lambda^*_1(x,y_1) \\ \cdot \\ \cdot \\ \lambda^*_M(x,y_M) \end{bmatrix} \qquad (4)$$

where $y(t)$ is the vector representing the state of all searchers.

Let $dN(t)$ denote the composite report from all searchers:

$$dN(t) = \begin{bmatrix} dN_1(t) \\ \cdot \\ \cdot \\ dN_M(t) \end{bmatrix} \qquad (5)$$

Assuming that the searchers are efficiently deployed, the probability that two sensors will detect the source simultaneously in an interval Δt is infinitesimal and will be ignored. Thus possible outcomes for $dN(t)$ are:

(1) $dN(t) = \underline{0}$, in which no detections are reported; or

(2) $dN(t) = \underline{e}$, in which α-th sensor reports a detection, and all others report no detections.

<u>Theorem:</u> (Evolution of density under "pure" search by multiple sensors)

Let $x(t)$ be the vector Markov process defined in Eq. (1) describing the behavior of the signal source. Let the measurement process consist of unit jump process defined by Eqs. (3) and (4). Under the assumption that only one sensor detects the source at any given time, the density $p \triangleq p(x,t|x_o,t_0,\mathcal{G}^t_{t_0})$ satisfies the following equation (Snyder's equation):

$$\frac{\partial p}{\partial t} = L^+(p) + \sum_{\alpha=1}^{M} \Bigg\{ [\lambda^*_\alpha(x(t),y_\alpha(t)) - E\{\lambda^*_\alpha(x(t),y_\alpha(t))\}] \cdot [E\{\lambda^*_\alpha(x(t),y_\alpha(t))\}]^{-1} \cdot \left[\frac{dN_\alpha(t)}{dt} - E\{\lambda^*_\alpha(x(t),y_\alpha(t))\}\right] p \Bigg\} \tag{6}$$

where the expectation $E\{\lambda^*(x(t),y_\alpha(t))\}$ is with respect to the process $x(t)$. The operator $L^+(\cdot)$ is the forward operator in the Fokker-Planck equation.

An important case of interest is one in which no sensor detects the source over the time interval $[t_0,t_0+T]$. In this case, $dN(t) \equiv 0$ for $t \in [t_0,t_0+T]$ and the conditional density evolves according to:

$$\frac{\partial p_o}{\partial t} = L^+(p_o) - \sum_{\alpha=1}^{M} [\lambda^*_\alpha(x(t),y_\alpha(t)) - E\{\lambda^*_\alpha(x(t),y_\alpha(t))\}] \cdot [E\{\lambda^*_\alpha(x(t),y_\alpha(t))\}]^{-1} p_o . \tag{7}$$

This is the partial integro-differential equation which describes the evolution of the transition density of the state of the source when M sensors are searching and have failed to detect the target. The special case for a single searcher follows immediately by using M=1 and coincides with the equation derived by Hellman.

The above integro-partial differential equation can be transformed into a non-linear partial differential equation (PDE) by a suitable transformation. The solutions to the resulting PDE are computationally expensive and, in general, do not provide a practical solution to tactical decision-making problems faced by the C^2 system. These solutions, however, do provide an insight into how the transition probability density function evolves. An example is given below.

The target is assumed to follow linear dynamics with constant speed.

$$\dot{x} = Ax + w \tag{8}$$

where x denotes the geographical state of the system.

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

and $w(t)$ is a standard Gaussian white noise process. The target is moving at unit speeds along the east and north axes.

The sensor searching the target is assumed to follow the following geographical trajectory:

$$y_e(t) = 1.5 + 2t \tag{9a}$$

$$y_N(t) = y_e(t) + \tfrac{1}{2}\sin(4\pi t) \tag{9b}$$

The target starts at [3.0, 3.0] at time 0 and the searcher starts at [1.0, 1.0] at time 0. The instantaneous detection function at the searcher is given by

$$\lambda(x(H,y(t)) = \frac{C_\lambda}{2\pi\sqrt{Q}} e^{-\frac{1}{2}\|y-Dx\|^2_{Q^{-1}}}$$

when

$$D = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

and C_λ and Q are constants defining the sensor characteristics.

The evolution of the resulting density function for the target state is shown in 1(a), (b), and (c) for three time instances. The significant characteristic of the density function is the depression resulting from the "null" information provided by the searcher as it moves along its trajectory. This is a direct result of the information contained in the observation that the sensor did not detect a target when it searched for a target. Results for the discrete-time system are considered next.

Discrete Time Models

The continuous time evolution of the transition density discussed above provides an analytical mechanism to evaluate various search paths $\{y(t)\}$ used by the searcher. However, the computational complexity of evaluating the detection probabilities corresponding to a search path prohibit use of a continuous model in practical applications. A time-discrete version of the same problem provides an approach that may be used in real-time applications.

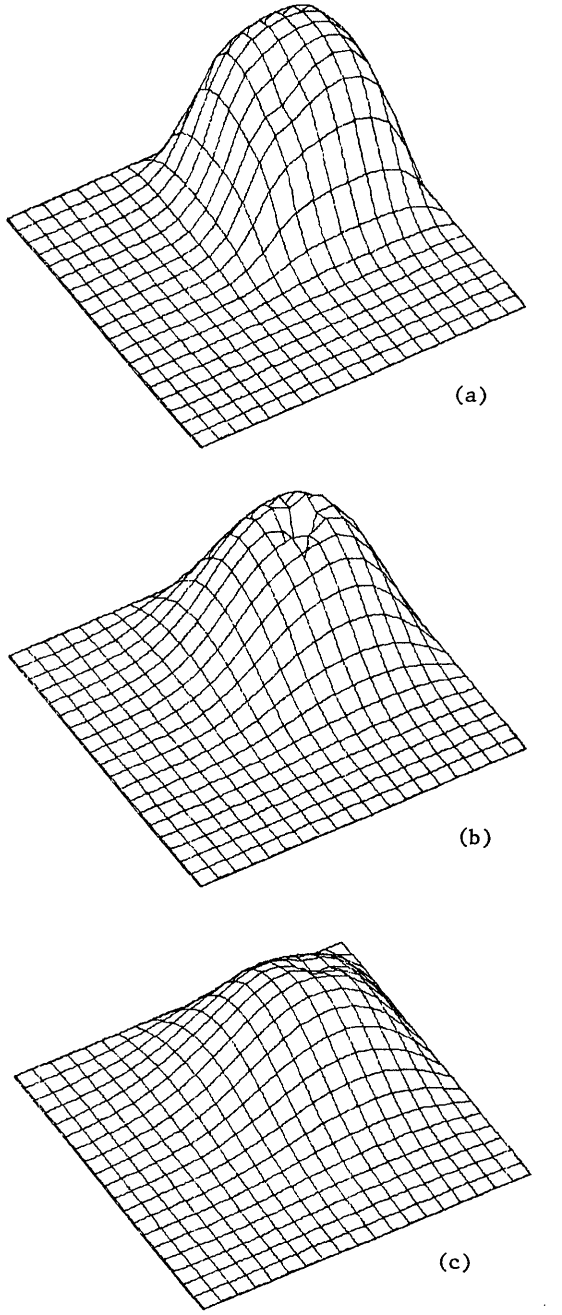


Fig. 1 Evolution of a posteriori density function (continuous).

Let $\{t_k\}_0^N$ be a partitioning of the time interval $[0,T]$. Assume the target dynamics to be linear:

$$x_{k+1} = \varphi_k x_k + w_k \tag{10}$$

where φ_k is the known state transition matrix and $\{w_k\}$ is a zero mean Gaussian white noise sequence with auto-covariance matrix Q_k. Assume that the searcher trajectory is controlled through the dynamics

$$y_{k+1} = g_k y_k + b_k u_k \tag{11}$$

and that $\{y_k\}$ is perfectly known if control policy $\{u_k\}$ is known. The detection model is described through a one-glimpse detection probability $\lambda(x_k,y_k)$. Also,

$$0 \le \lambda(x_k,y_k) \le 1 \ .$$

The objective of the searcher is to determine the optimal strategy $\{u_k\}_{k=0}^{N-1}$ to maximize the resulting first-time detection probability $P_D(N)$. If at any time t_D the target is positively detected, then the search is terminated. If the searcher has failed to detect the target during $[t_0,t_k]$, then the search is to continue until the target is detected or t_N is reached.

Let $p_o(x_k|\mathscr{G}^k)$ denote the a posteriori probability density function given the observation that for the searcher using trajectory $\{y_i\}_{i=0}^k$ the search was unsuccessful. Then, the first time detection probability $P_D(k)$ is given by:

$$P_D(k) = 1 - \prod_{i=0}^{k} \{1-P_D(t_{i-1},t_i)\} \tag{12}$$

where $P_D(t_{i-1},t_i)$ is the one-glimpse probability that the target is detected in the time interval $[t_{k-1},t_k]$. Also,

$$P_D(t_{k-1},t_k) = \int_{X_k} \lambda(x_k,y_k)p_o(x_k)dx_u$$

$$= E_o\{\lambda(x_u,y_u)\} \ .$$

The maximization of $P_D(N)$ thus can be equivalently achieved by maximization of

$$J = \sum_{k=0}^{N} E_o\{\lambda(x_k,y_k)\} \ . \tag{13}$$

Typically, $\lambda(x_k,y_k)$ is a monotonically decreasing function of the weighted distance $r_k = \|y_k-D_kx_k\|_{R_k^{-1}}$ between the geographical coordinates y_k and D_kx_k. For exponential functions,

$$J = \sum_{k=0}^{N} E_o\left\{e^{-\|y_k-D_kx_k\|^2_{R_k^{-1}}}\right\}. \tag{14}$$

Analytically defined separation control strategies for sums of exponentials are generally not possible. In the following sections problems of minimizing a related performance measure

$$J_o = \sum_{k=0}^{N} E_o\left\{\|y_k-D_kx_k\|^2_{R_k^{-1}} + \|u_k\|^2_{P_k^{-1}}\right\} \tag{15}$$

are considered. Problems with

$$J_E = E\left\{\exp\left\{\sigma\sum_{k=0}^{N}\left(\|y_k-D_kx_k\|^2_{R_k^{-1}} + \|u_k\|^2_{P_k^{-1}}\right)\right\}\right\} \qquad (16)$$

can also be treated to obtain similar separation controls.

In the following paragraphs rules are derived for the evolution of $p_o(x_k|\mathcal{G}^k)$ and corresponding separation control policies $\{u_k^*\}_{k=0}^{N-1}$ for the performance criteria defined in (15).

Evolution of A Posteriori Density

First, consider the rules of evolution for the a posteriori density for target state given the fact that the search has been unsuccessful. Formally,

$$p(x_{k+1}|\mathcal{G}^{k+1};y^{k+1}) \triangleq p(x_{k+1}|N_{k+1}=0,\ \mathcal{G}^k;y_{k+1},y^k)$$

represents this density. Using Bayes' rule:

$$p(x_{k+1}|\mathcal{G}^{k+1};y^{k+1}) = \frac{P(N_{k+1}=0|x_{k+1},y_{k+1})p(x_{k+1}|\mathcal{G}^k;y_{k+1},y^k)}{\int_{x_{k+1}}(\text{Numerator})dx_{k+1}}.$$

In the above expression, the first term in the numerator on the right-hand side is the probability of an unsuccessful attempt in detecting the target, given that the target state is x_{k+1} and the sensor state is y_{k+1}. This is given by

$$P(N_{k+1}=0|x_{k+1},y_{k+1}) = 1-\lambda(x_{k+1},y_{k+1})$$

where $\lambda(x_{k+1},y_{k+1})$ is the one-glimpse detection probability for the sensor. For the purpose of analysis below, we assume the following form for $\lambda(x_k,y_k)$:

$$\lambda(x_k,y_k) = c^kN(y_k|D_kx_k,R_k)$$

where

$$D_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

thus projecting the $\mathbb{R}^4$ vector on the coordinate plate (x_1,x_2) in $\mathbb{R}^2$.

Assuming Gaussian noise in target dynamics and the fact that

$$p(x_{k+1}|\mathcal{G}^k;y_{k+1},y^k) = \int_{x_k} p(x_{k+1},x_k|\mathcal{G}^k;y_{k+1},y^k)dx_k\,,$$

a recursive form for $p(x_{k+1}|\mathcal{G}^{k+1})$ can now be derived.

In the derivation the following well-known relationships are used:

1. $$\int_{x_k} N(x_{k+1}|\varphi_kx_k,Q_k)N(x_k|\mu_i^k,\Sigma_i^k)dx_k = N(x_{k+1}|\varphi_k\mu_i^k,Q_k+\varphi_k\Sigma_i^k{}_k^T) \triangleq N(x_{k+1}|\mu_i^{k'},\Sigma_i^{k+1'})$$

where

$$\mu_i^{k+1'} = \varphi_k\mu_i^k$$

$$\Sigma_i^{k+1'} = Q_k + \varphi_k\Sigma_i^k\varphi_k^T$$

2. $$N(y_{k+1}|D_{k+1}x_{k+1},R_{k+1})N(x_{k+1}|\mu_i^{k+1'},\Sigma_i^{k+1'}) = N(y_{k+1}|D_{k+1}\mu_i^{k+1'},M_i^{k+1})\cdot N(x_{k+1}|\mu_i^{k+1},\Sigma_i^{k+1})$$

where

$$M_i^{k+1} = R_{k+1} + D_{k+1}\Sigma_i^{k+1'}D_{k+1}^T$$

$$K_i^{k+1} = \Sigma_i^{k+1'}D_{k+1}^T(M_i^{k+1})^{-1}$$

$$\Sigma_i^{k+1} = (I-K_i^{k+1}D_{k+1})\Sigma_i^{k+1'}$$

$$\mu_i^{k+1} = \mu_i^{k+1} + K_i^{k+1}(y_{k+1}-D_{k+1}\mu_i^{k+1'})\,.$$

[Note that the above equations are identical to Kalman filter equations.]

With the form of $\lambda(x_k,y_u)$ as chosen above, the density $p(x_{k+1}|\mathcal{G}^{k+1};y^{k+1})$ can be computed recursively as a sum of Gaussian densities. Let

$$p(x_{k+1}|\mathcal{G}^{k+1};y^{k+1}) \quad \sum_{i=1}^{\delta_{k+1}}\alpha_i^{k+1}N(x_{k+1}|\mu_i^{k+1},\Sigma_i^{k+1})\,.$$

Then, the recursive computations require determining δ_{k+1} and $\{\alpha_i^{k+1},\mu_i^{k+1},\Sigma_i^{k+1}\}_{i=1}^{\delta_{k+1}}$ from the corresponding quantities at k. These recursions are given below.

$$P(x_{k+1}|\mathcal{G}^{k+1}) \triangleq \sum_{i=1}^{\delta_{k+1}}\alpha_i^{k+1}N(x_{k+1}|\mu_i^{k+1},\Sigma_i^{k+1})$$

where

$$\delta_{k+1} = 2\delta_k$$

$$\beta_{k+1} = 1 - \sum_{\ell=1}^{\delta_k} c_\lambda^{k+1} \alpha_\ell^k N(y_{k+1} | D_{k+1}\mu_\ell^{k+1'},$$

$$D_{k+1}\Sigma_\ell^{k+1'} D_{k+1}^T + R_{k+1})$$

for $i = 1,2,\ldots,k$.

$$\alpha_i^{k+1} = \alpha_i^k / \beta_{k+1}$$

$$\mu_i^{k+1} = \mu_i^{k+1'}$$

$$\Sigma_i^{k+1} = \Sigma_i^{k+1'}$$

for $i = \delta_k+1,\ \delta_k+2,\ \ldots,\ \delta_{k+1}$.

$$j = i - \delta_k$$

$$\alpha_i^{k+1} = -c_\lambda^{k+1} \alpha_j^k N(y_{k+1} | D_{k+1}\mu_\ell^{k+1'},$$

$$D_{k+1}\Sigma_j^{k+1'} D_{k+1}^T + R_{k+1})/\beta_{k+1}$$

$$\mu_i^{k+1} = \mu_j^{k+1'} + K_j^{k+1}(y_{k+1} - D_{k+1}\mu_j^{k+1'})$$

$$\Sigma_i^{k+1} = (I - K_j^{k+1})\Sigma_j^{k+1'}$$

$$K_j^{k+1} = \Sigma_j^{k+1'} D_{k+1}^T (R_{k+1} + D_{k+1}\Sigma_j^{k+1} D_{k+1}^T)^{-1}$$

The equations for the evolution of the a posteriori density function when M independent sensors are reporting "null" information have the same form as above but with a larger number of terms.

The above recursive equations were used to compute the probability density functions for the state of a stationary target and a searcher moving along a spiral of the form $r = k\theta$ toward the a priori estimate of the target. Densities at three time instances are shown in Figs. 2(a), (b), and (c).

Stochastic Control of Sensors

With the recursive evolution of the density function and the linear-quadratic system described by Eqs. (10) through (15), a separation policy can now be generated. Standard techniques are used in deriving this policy. Only summarized equations are given below.

The optimal control u_{k-1}^* is given through a backware induction of the form:

$$u_{k-1}^* = \Lambda_{k-1}^1 {}_{k-1}\mu_{k-1} + \Lambda_{k-1}^2 g_{k-1} y_{k-1}$$

with $\mu_{k-1} = E_o\{x_{k-1} | \mathscr{G}^{k-1}\}$

and

$$\Lambda_{k-1}^1 = M_{k-1}^{-1} b_k^T (R_{k+1} - I_{N-k-1}^{21})\varphi_k$$

$$\Lambda_{k-1}^2 = M_{k-1}^{-1} b_k^T (R_{k+1} + I_{N-k-1}^{22}) g_k$$

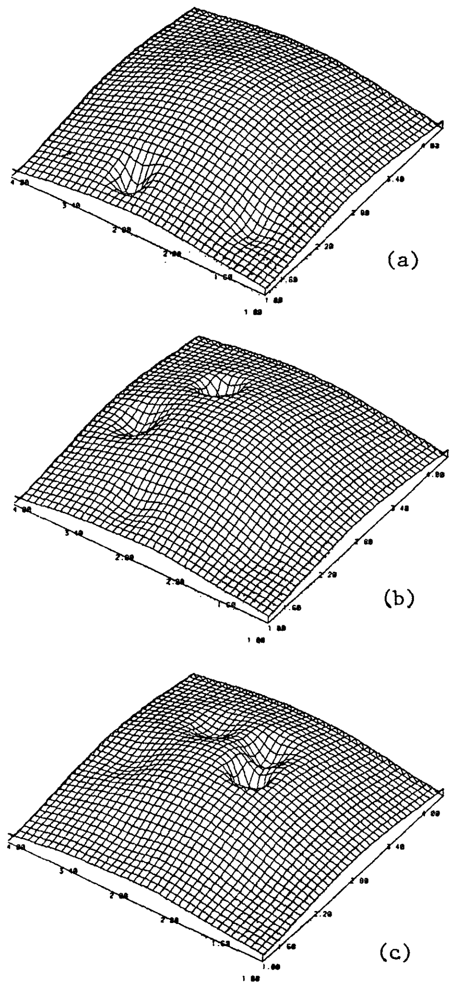


Fig. 2 Density at times 1, 3, and 5, respectively.

$$M_{k-1} = b_k^T (R_{k+1} + I_{N-k-1}^{22}) b_k + P_k$$

where I_{N-k-1}^{21} and I_{N-k-1}^{22} can be recursively computed.

Similar separation control policies can be derived for the linear exponential-quadratic problem. Details of these policies and numerical results will be given in a forthcoming report.

Conclusions

Results are presented for estimation and control of dynamic sensor systems operating in a tactical environment. These control rules determine sensor trajectories until a successful detection is achieved. Recursive computation of the a posteriori density make the separation control possible. Further numerical studies are planned for the future.

REFERENCE

Hellman, Olavi (1970). On the effect of a search upon the probability distribution of a target whose motion is a diffusion process. Annals of Math. Stat., 41:5, 1717-1724.

OPTIMAL SENSOR SCHEDULING FOR MULTIPLE HYPOTHESIS TESTING[1]

Robert R. Tenney[2]

Laboratory for Information and Decision Systems
Massachusetts Institute of Technology, Cambridge, Massachusetts 02139 USA

Abstract. The generic problem of selecting the sequence of sensors which optimizes the information received about a number of discrete hypotheses is considered. The optimization criterion penalizes the uncertainty present about pairs of hypotheses in a form which has an eigenfunction property with respect to a Bayes update of the conditional probability distribution. Application of the Pontryagin minimum principle yields elegant solutions to an interesting class of problems. Applications in surveillance, failure detection, and nondestructive testing are possible.

I. INTRODUCTION

The problem: Often several competing hypotheses exist about the state of a particular entity, and real time observations must be used to discriminate between them. Once the set of sensors to be used has been specified, the observations can be used to update prior information in a number of ways, although Bayes' theorem underlies some of the most common techniques [1]. In this framework, the net information is captured in the posterior probability distribution over the hypotheses.

In cases where several sensors are available but are mutually exclusive in their use (either due to interference, or because one physical sensor must be pointed in one of a number of directions), an additional problem arises in determining, also in real time, that sequence of sensors which should be activated to provide the above information. The efficacy of a particular sensor sequence must be related to the character of the resulting posterior probabilities; these should clearly discriminate among the hypotheses.

Mathematically, this can be viewed as a problem of selecting, at each point in time, one of M sensors to obtain information about a set of K hypotheses. By defining an interesting cost function on the set of posterior distributions, we can seek an optimal sensor scheduling procedure.

Overview: The strategy taken in the sequel is to first pose the problem in discrete time, with emphasis on the new cost function and its interpretation. Then the stochastic problem is reduced to a deterministic control problem, in continuous time, with a convex control set. The Pontryagin minimum principle can be brought to bear on the problem, and the resulting necessary conditions for the optimal schedule define a two point boundary value problem with sectorwise linear dynamics. The general structural assumptions lead to an iterative solution for the general case, and an elegant solution for a more restricted set of problems. An interesting side result is a geometrical characterization of each sensor by a vector of parameters describing its capabilities to distinguish between various pairs of hypotheses.

II. PROBLEM STATEMENT

Hypotheses: The K hypotheses, one of which may be valid, are denoted H_k, $k = 1,\dots,K$.

Prior knowledge provides a probability distribution at time $t = 0$, denoted by $\vec{\pi}(0)$, where

$$\vec{\pi}_k(0) = p(H_k)$$

Observations: Sensor outputs obtained at time t from sensor j are denoted $\vec{y}_j(t)$.

The statistics of $\vec{y}_j(t)$ are independent of everything except the sensor j and the underlying hypothesis H_k; in particular

$$p(\vec{y}_j(t)\,|H_k,\ \vec{y}_\ell(s)) = p(\vec{y}_j(t)|\ H_k) \qquad s \neq t$$

---

[1] Support for this work is from the Office of Naval Research, under contract N00014-77-0532 is gratefully acknowledged.

However, no assumption of stationarity of these distributions need be made.

If

$$\pi_k(t) = p(H_k|\vec{y}(1),\ldots,\vec{y}(t))$$

then by Bayes' law

$$\pi_k(t+1) = \frac{p(y(t+1)|H_k)\pi_k(t)}{p(\vec{y}(t+1))}$$

where subscripts denoting sensor choices have been omitted. These equations give the dynamic equations for the evolution of the posterior distribution as observations are obtained.

Cost: The objective of a detection or idenification algorithm is to produce correct estimates of the true state of a systems. It is also beneficial if these estimates come with high confidence levels. Thus, if one is seeking to drive posterior distributions to some values, the best values are near the extremes, where the true hypothesis is known with almost certainty.

Consider the binary hypothesis case. Figure 1 shows three candidate penalty functions which have reasonable qualitative characteristics - they are all minimum at the extremes and convex downward. Number 1 is the minimum probability of error incurred if a decision between H_1 and H_2 had to be made. Number 2 is a direct measure of the uncertainty in the distribution: it is the entropy (scaled by 1/2)

$$-\frac{1}{2}\sum_{k=1}^{z} \pi_k \log_2 \pi_k$$

The third is similar to the second

$$v(\pi_1, \pi_2) = \sqrt{\pi_1 \pi_2}$$

Note that the last two indicate that an improvement in the probability of error from, say, 10% to 1% is much more rewarding than one from 49% to 40% and thus greatly encourage extremal distributions.

The third form posesses unique analytical properties, as will be seen in section III. It can be generalized to the form

$$v(\pi_1,\ldots,\pi_k) = \prod_{k=1}^{K} \pi_k^{r_k}$$

$$1 = \sum_{k=1}^{K} r_k \qquad r_k \geq 0$$

and to sums of terms of this form without compromizing these properties.

Definition: An obscurity function $v(\vec{\pi})$ is of the form

$$v(\vec{\pi}) = \sum_{i=1}^{N} b_i\, v_i(\vec{\pi})$$

with each $v_i(\vec{\pi})$ having the form of above

The obscurity function measures the lack of knowledge about the hypotheses. It is minimum when $\vec{\pi}$ is pure, i.e. when all but one component are zero. The coefficients b_i represent weights attached to varying types of obscurity as indicated by the form of the form of the associated $v_i(\vec{\pi})$.

Example: Consider the ternary hypothesis testing problem. Two candidate obsurity functions are

$$v^1(\vec{\pi}) = (\pi_1\, \pi_2\, \pi_3)^{1/3}$$

and

$$v^2(\vec{\pi}) = (\pi_1\, \pi_2)^{1/2} + (\pi_1\, \pi_3)^{1/2} + (\pi_2\, \pi_3)^{1/2}$$

Both are zero for pure $\vec{\pi}$; both have their maxima at $\vec{\pi} = [1/2\ 1/3\ 1/3]^T$. However, the former includes as minima all distributions with one component zero, the latter has only three minimum points (Figure 2). The former is minimized when any hypothesis is eliminated; the latter, when any hypotheis is confirmed.

The above definition is quite general; in the sequel we will assume all terms in the obscurity function are of the form

$$v_i(\vec{\pi}) = (\pi_{k_1}\, \pi_{k_2})^{1/2}$$

All results will generalize to the earlier case, but this makes clear that each term in $v(\vec{\pi})$ represents the degree to which a pair of hypotheses can be distinguished, and thus a different type of obscurity.

The selection of the obscurity function provides a great deal of flexibility. For instance, if one is only interested in determining whether or not H_1 is true, a function of the form

$$v(\pi_i) = \sum_{k=2}^{K} [\pi_1\, \pi_k]^{1/2}$$

is appropriate, as it penalizes ambiguity between H_1 and any other hypothesis without including the obscurity between the others.

III. REDUCTION TO A CONTINUOUS DETERMINISTIC OPTIMAL CONTROL PROBLEM

The problem stated above is a stochastic optimization problem, where the original, imperfectly observable state H has been replaced with the conditional probability $\vec{\pi}$, which can be determined exactly. State transitions are still stochastic, due to the appearance of $\vec{y}(t)$, but possess a Markov property. This is a standard approach [2] to dealing with this type of problem; the next step might be to use dynamic programming to obtain a feedback solution, where j(t) would be selected on the basis of $\vec{\pi}(t)$.

Due to the lack of success of this approach in producing implementable solutions for the general, multiple hypothesis problem consider a less ambitious goal: finding the optimal <u>open loop</u> schedule (i.e. select the best sequence of sensors based only on the prior distribution). Not only are solutions of this form applicable in some cases where feedback solutions cannot be implemented, but they can be used as open loop feedback solutions where the entire schedule is effectively recomputed at each time, using $\vec{\pi}(t)$

<u>Reduction to Deterministic Dynamics</u>: The for of the obscurity function was selected for its qualitative properties and because of:

Theorem 1: Terms of an obscurity function are eigenfunctions of the expectation/Bayes update operation and the associated eigenvalue is completely determined by sensor characteristics.

Proof: Let[2]

$$v_i(\vec{\pi}(t+1),\ t+1) = x_i(t+1)\ (\pi_1\ \pi_2)^{1/2}$$

Then

$$v_i(\vec{\pi}(t),\ t) = \underset{\vec{y}(t+1)}{E} \{v_i(\vec{\pi}(t+1),\ t+1)\}$$

Substituting,

$$v_i(\vec{\pi}(t),\ t) = \int_{\vec{-\infty}}^{\vec{\infty}} p(\vec{y}_j(t+1)\ x_i(t+1)$$

$$\frac{p(\vec{y}_j(t+1)|H_1)\pi_1(t)p(\vec{y}_j(t+1)|H_2)\pi_2(t)}{(p(\vec{y}_j)t+1)))^2}^{1/2}$$

$$= \alpha_{ij}(t+1)x_i(t+1)(\pi_1(t)\pi_2(t))^{1/2}$$

where j is the sensor selected for t+1 and

$$\alpha_{ij}(t+1) = \int_{\vec{-\infty}}^{\vec{\infty}} (p(\vec{y}_j(t+1)\ H_1)$$

$$p(\vec{y}_j(t+1)|H_2))^{1/2}\ d\vec{y}_j$$

Thus, $v_i(\vec{\pi}(t+1),\ t+1)$ is an eigenfunction of the update with $\alpha_{ij}(t+1)$ is eigenvalue.

Now, for a fixed sequence of sensors $\tilde{j} = j(t),\ t = 1,\ldots,T$, define the expected cost-to-go at time t with conditional distribution $\vec{\pi}(t)$ as

$$V_{\tilde{j}}(\vec{\pi}(t),\ t) = E\{\sum_{s=t+1}^{T} v(\vec{\pi}(s))\}$$

where $V_{\tilde{j}}(\vec{\pi}(T),T) = 0$ at the terminal time. The key result is then

Theorem 2: At each time t, the cost-to-go takes the form

$$V_{\tilde{j}}(\vec{\pi}(t),t) = \sum_{i=1}^{N} x_i(t)v_i(\vec{\pi}) \qquad (*)$$

where

$$x_i(t) = \alpha_{ij}(t+1)\ x_i(t+1) + b_i$$

$$x_i(T) = 0$$

Proof: By reverse induction. At t = T, the cost-to-go is uniformly zero. Assume (*) holds at time t+1, so

$$V_{\tilde{j}}(\vec{\pi}(t),t) = E\{V_{\tilde{j}}(\tilde{\pi}(t+1),\ t+1\} + b_i\ v_i(\vec{\pi}(t))$$

$$= E\ \{\sum_{i=1}^{N} x_i(t+1)v_i(\vec{\pi}(t+1))\} +$$

$$+ b_i\ v_i(\vec{\pi}(t))$$

$$= \sum_{i=1}^{N} \alpha_{ij}(t+1)x_i(t+1)v_i(\vec{\pi}(t)) +$$

$$+ b_i\ v_i(\vec{\pi}(t))$$

$$= \sum_{i=1}^{N} (\alpha_{ij}(t+1)x_i(t+1)+b_i)v_i(\vec{\pi}(t))$$

and $v_i(\vec{\pi})$ is of the fundamental form.

This gives a deterministic <u>linear</u> dynamical problem with states $x_i(t)$ representing the amplitudes of a finite number of modes of the cost-to-go function excited by the terms of the obscurity function. The coefficints $\alpha_{ij}(t)$ represent the decay of $x_i(t)$ when sensor j is selected at time t, and the driving terms b_i representing the relative importance of each term. Moreover, the $x_i(t)$ are truely states as their evolution depends only on selections j made between t and T, although this property holds in <u>reverse time</u>.

Corollary 2a: The total cost is

---

[2] For notational simplicity here, assume the i^{th} term of the obscurity function involves H_1 and H_2.

$$V_j(\vec{\pi}(0,0)) = \sum_{i=1}^{N} x_i(0)v_i(\vec{\pi}(0))$$

Proof: Immediate from Theorem 2 when t = 0.

If (known) parameters c_i are defined as

$$c_i = v_i(\vec{\pi}(0))$$

then the equivalent deterministic optimal control problem is to select $\tilde{j}$ to minimize

$$\sum_{i=1}^{N} x_i(0)\ c_i$$

subject to

$$x_i(t-1) = \alpha_{ij}(t)\ x_i(t) + b_i$$

$$x_i(T) = 0$$

Interpretation of the α_{ij}: These parameters measure the ability of sensor j to contribute to the reduction of each term of the obscurity function. The set $\{\alpha_{ij}\ i=1,\dots,N\}$ describe the information gathering ability of j in all directions which are contained in $v(\vec{\pi})$. For example, sensor 1 may be able to distinguish H_1 from H_2 and H_3, but not between the latter, while sensor 2 only separates H_2 from H_3. The information from each sensor alone is incomplete; the set above paves the way towards a geometric interpretation of information.

Gross properties of the α_{ij} are

Theorem 3: For all i,j,t,

$$0 \le \alpha_{ij}(t) \le 1$$

with the lower limit obtained iff it is possible to completely eliminate one of the hypotheses in v_i with any single observation $\vec{y}_j(t)$, and the upper iff $\vec{y}_j(t)$ is independent of the hypothesis in $v_i(\vec{\pi})$.

Proof: Since $p(\vec{y}_j(t)|H_k) \ge 0$ for all $\vec{y}_j$, for all $\vec{y}_j$

$$\int_{-\vec{\infty}}^{\vec{\infty}} (p(\vec{y}_j(t)|H_{k_1})p(\vec{y}_j(t)|H_{k_2}))^{1/2}\ d\vec{y} \ge 0$$

with equality iff

$$p(\vec{y}_j(t)|H_{k_1})\ p(y_j(t)|H_{k_2}) = 0$$

for all $y_j(t)$, i.e. iff the set of $\vec{y}_j(t)$ which may result when H_2 is true is disjoint from that possible when H_{k_2} is true, and hence $\vec{y}_j(t)$ provides perfect information to distinguish between them. Since also

$$\int_{-\vec{\infty}}^{\vec{\infty}} p(\vec{y}_j|H_k)d\vec{y} = 1$$

the integral

$$\int_{-\vec{\infty}}^{\vec{\infty}} p(\vec{y}_j(t)|H_{k_1})p(\vec{y}_j(t)|H_{k_2})d\vec{y} \le 1$$

with equality iff

$$p(\vec{y}_j(t)|H_{k_1}) = p(\vec{y}_j(t)|H_{k_2})$$

for all $\bar{y}_j(t)$.

Thus, qualitatively speaking, good schedules use sensors where the α_{ij} are small for terms where c_i or b_i are large.

In preparation for the transition to continuous time, introduce

Definition: The clarification coefficient of sensor j with respect to $v_i(\vec{\pi})$ is $a_{ij}(t)$

$$a_{ij}(t) = -\ln \alpha_{ij}(t)$$

Corollary 3a: Clarification coefficients are nonnegative and unbounded with equality to zero holding iff the sensor produces outputs which are independent of the hypotheses of the associated term.

Proof: Properties of ln.

Reformulation in Continuous Time: The remainder of this section deals with improving the analytic properties of the problem by replacing the discrete time and control sets with continuous equivalents. The problem as posed above can be solved using the discrete time minimum principle but the solution has implicit proerties which are less cumbersome in a continuous time framework.

Consider the formal continuous time analog: minimize

$$\sum_{i=1}^{N} c_i\ x_i(0)$$

with

$$\frac{dx_i}{d(-t)} = -a_{ij}(t)x(t) + b_i(t) \qquad x_i(T) = 0$$

where again the dynamics appear in reverse time. Integrating from time t to time t − δ gives approximately

$$x_1(t-\delta)-x_i(t) = -a_{ij}(t)x_i(t)\delta + b_i(t)$$

$$x_i(t-\delta) = (1-a_{ij})(t)\delta)x_i(t) + b_i(t)\delta$$

$$\tilde{\sim} e^{-a_{ij}(t)\delta} k_i(t) + b_i(t)\delta$$

provided δ is sufficiently small that second order terms can be neglected, i.e.

$$a_{ij}(t)\delta << 1$$

Setting $\delta = 1$

$$x(t-1) \tilde{\sim} \alpha_{ij}(t)x_i(t) + b_i(t)$$

Provided the $a_{ij}(t)$ are the clarification coefficients and α_{ij} (t) the eigenvalues, these may be valid approximations.

Convexification of Control Variable: In the problem thus far there have been a discrete set of sensors from which to select. It will be convenient to convexify this set by introducing the M control variables $u_j(t)$, which specify what fraction of an infinitesimal cycle is devoted to each sensor j. Thus

$$\sum_{j=1}^{M} u_j(t) = 1 \qquad u_j(t) \geq 0$$

are the constraints which admissible controls must satisfy.

With this interpretation, the dynamics become

$$x_i(t-\delta) \tilde{\sim} e^{-\sum_j a_{ij}(t)u_j(t)\delta} x_i(t) + b_i\delta$$

or, as $\delta \to 0$

$$\frac{dx_i}{d(-t)} = -a_i(\vec{u})x_i(t) + b_i$$

where

$$a_i(\vec{u}) = \sum_{j=1}^{M} a_{ij}(t)u_j(t)$$

This convexification of the control set allows the above interpretation of polling sensors with u_j being the fraction of time devoted to sensor j. Mixed controls (some $u_j \notin \{0,1\}$) do arise in the optimal solution. Were a solution attempted without convexification, the optimal solution would still be forced to achieve this mixture by infinitesimal "time sharing". In practice, either this polling can be approximated or, in open loop feedback uses, it will almost never occur as the set of $\vec{\pi}(0)$ for which it is initially required is of measure zero.

Reversal of the Time Index: Finally, the reverse dynamics that naturally arose above are a notational nuisance; replacing the time variable t with another t'

$$t' = T - t$$

yields an identical problem more in line with standard optimal control problems. The only caveat is that the solution to the resulting problem, $\vec{u}^*(t')$, is the reverse of the optimal schedule.

Conclusion: This section has reduced the original problem to:

minimize

$$\sum_{i=1}^{N} c_i\ x_i(T) \qquad c_i = v_i(\vec{\pi}(0))$$

with

$$\dot{x}_i(t') = -a_i(\vec{u}(t'))x_i(t') + b_i \qquad x_i(0) = 0$$

The system dynamics $a_i(\vec{u}(t'))$ will always be nonnegative, and larger values correspond to greater clarification by the selected sensor.

IV. OPTIMAL SOLUTION OF REDUCED PROBLEM

Here the above problem can be interpreted using the Pontryagin minimum principle and the geometric structure of the solution emerges. The first section introduces the type of results obtained by examining the binary hypothesis problem where only one dimension of obscurity exists. After stating necessary conditions which the optimal schedule must satisfy and deriving some of its properties, the interpretation of these con-

---

[5] As the discrete sample rate goes to zero, this expression is exact for all stationary processes, as well as for nonstationary Gaussian and Poisson processes. It provides a piecewise linear approximation for other sampled, nonstationary independent increments processes; however, the sequel will assume $\alpha_{ij}(t)$ to be twice differentiable and thus a more advanced approximation, such as splines, may be necessary.

ditions in terms of sensor clarification coefficients provides some preliminary tests for eliminating sensors from consideration. Further examination of singular (mixed) control arcs yields more basic structure of the schedule as well as a classification of problems in terms of the sensor sets. These will be the general results; section V will exploit the necessary conditions to compute the optimal schedule.

Preview - Binary Hypothesis Testing: This special case illustrates some of the conclusions that can be drawn about optimal schedules. The obscurity functions between the two hypotheses H_1 and H_2 has one term

$$v_1 = (\pi_1 \; \pi_2)^{1/2}$$

so the continuous problem to be solved is to minimize

$$c_1 \; x_1(T)$$

with

$$\dot{x}_1(t) = a_1(\vec{u}(t))x_1(t) + 1 \qquad x_1(0) = 0$$

Here the solution is obvious: choose $\vec{u}(t)$ to maximize the coefficient $a_1(\vec{u}(t))$ at each time t. This corresponds to selecting the sensor with maximal instantaneous clarification coefficient $a_{ij}(t)$ at each t.

Thus $\vec{u}$ is chosen to maximize (a function of) $a_1(t)$ at each time, and the selected sensor may vary as $a_{ij}(t)$ changes with time. No mixed controls are required here, but multiple terms in $v(\vec{\pi})$ will induce a directionality which requires mixing.

General Necessary Conditions: Necessary conditions for the dynamic optimization problems can be obtained from the Pontryagin minimum principle . They are summarized in

Theorem 4 : The optimal solution $\vec{u}(t)$ satisfies:

$$\sum_{i=1}^{N} z_i(t)a_i(\vec{u}^*(t)) \geq \sum_{i=1}^{N} z_i(t)a_i(\vec{u}(t)) \qquad \text{(H)}$$

$$\text{for all } \vec{u}(t)$$

$$\dot{z}_i(t) = q_i(t) \qquad z_i(0) = 0$$

$$\dot{q}_i(t) = a_i(\vec{u}^*(t))q_i(t) \qquad q_i(T) = b_i c_i$$

Proof: see [21].

Because of their associations with variables in the proof, (H) will be referred to as the Hamiltonian condition, $z_i(t)$ as states, and $q_i(t)$ as costates. In addition, introduce the M vectors

$$\vec{a}_j(t) = [a_{ij}(t) \quad a_{2j}(t) \; \ldots\ldots \; a_{Nj}(t)]^T$$

of clarification coefficients for each sensor and the NxM composite matrix

$$\underline{A}(t) = \begin{bmatrix} \vec{a}_1 & \vdots & \vec{a}_2 & \vdots & & \vdots & \vec{a}_M \\ & \vdots & & \vdots & \cdots & \vdots & \end{bmatrix}$$

so that

$$\vec{a}(\vec{u}(t)) = \underline{A}(t)\vec{u}(t)$$

is the vector of system coefficients $a_i(\vec{u}(t))$.

The necessary conditions can then be rewritten as

$$<\vec{z}(t), \vec{a}(\vec{u}(t))> \;\; \geq \;\; <\vec{z}(t), \vec{a}(\vec{u}(t))>$$

$$\text{for all } \vec{u}(t) \qquad \text{(H)}$$

$$\dot{\vec{z}}(t) = \vec{q}(t) \qquad \vec{z}(0) = \vec{0}$$

$$\dot{\vec{q}}(t) = \vec{a}(\vec{u}(t)) \cdot \vec{q}(t) \qquad \vec{q}(T) = \vec{b} \cdot \vec{c}$$

where $<\cdot,\cdot>$ is standard inner product in $\mathbb{R}^N$ and $\cdot$ is componentwise multiplication.

V. COMPUTATION OF OPTIMAL SCHEDULES

Returning from the optimal control problem above to the original scheduling problem involves reinterpreting the previous results in the context originally developed in sections II and III.

Begin by returning to forward time, so becomes

$$\dot{z}(t) = -\vec{q}(t) \qquad \vec{z}(T) = \vec{0}$$

$$\dot{\vec{q}}(t) = -\vec{a}(\vec{u}(t))\vec{q}(t) \qquad \vec{q}(0) = \vec{c} \cdot \vec{b}$$

as the two point boundary value problem. are static conditions and remain unchanged. The solution $\vec{u}(t)$ specifies the fraction of effort to be devoted to each sensor at time t. This section will discuss the numerical solution of these equations using structural knowledge obtained in section IV.

Numerical solution: Solution of equations can be achieved by a number of techniques, such as an iterative strategy which refines

---

guesses of the unknown boundary conditions at each end of the time interval. While the suitability of various techniques will depend heavily on the dynamics of the $\vec{a}_i(t)$, the following procedure is sugested for those problems where these coefficients vary slowly.

1. Initial guess: $\vec{z}(t) \equiv 0, \quad 0 \leq t \leq T$
2. Integrate the equation for $\vec{q}$ forward in time, selecting $\vec{u}(t)$ using an approximation based on the partitions induced on $\vec{q}$ space by

   $$\vec{u}(t) = \arg\max_{\vec{u}} \langle \vec{q}(t) \ , \ \vec{a}(\vec{u}(t)) \rangle$$

   to get an initial guess of $\vec{q}(T)$, denoted $\vec{q}(0;T)$
3. Integrate forward using $\vec{q}(k:T)$ as terminal condition $\vec{q}(T)$; store the switch schedule. Obtain $\vec{z}(k;0)$ as an estimate of the initial condition on $\vec{z}$.
4. Integrate backward using $\vec{z}(0) = \vec{z}(k:0)$ and the schedule obtained in (2); obtained $\vec{q}(k+1;T)$ and repeat (3) and (4).

The advantages of this technique are that only the switch times (and mixes over singular arcs) need be stored from iteration to iteration, rather than either $\vec{z}$ or $\vec{q}$ trajectories. For certain classes of problems it converges in one step. No other general properties of this solution are known at this point.

This provides a technique to apply to complex problems with little structure; special cases with strong structure can lead to much simpler solutions.

REFERENCES

1. H.L. Van Trees, Detection, Estimation and Modulation Theory; Part 1, Wiley, 1965.
2. D.P. Bertsekas, Dynamic Programming and Stochastic Control, Academic Press, 1976.
3. R.R.Tenney, "Optimal Sensor Scheduling for Multiple Hypothesis Testing", LIDS-P-1179, September 1981.

OPTIMAL PLATFORM MANEUVERS FOR PASSIVE TRACKING WITH TIME DELAY MEASUREMENTS

Pan-Tai Liu
Mathematics Department
University of Rhode Island
Kingston, Rhode Island 02881

Paul L. Bongiovanni
Naval Underwater Systems Center
Newport Laboratory
Newport, Rhode Island 02840

Abstract. This paper describes a technique for determining optimal platform maneuvers for passive tracking when time delay measurements are available from a linear array. Unlike the bearings-only-ranging problem, an additional time delay proportional to range makes a platform maneuver unnecessary from a mathematical viewpoint. However, experience has shown that even in this situation an appropriately selected maneuver can improve convergence of the estimation process. This technique is designed to be used in conjunction with an estimation algorithm such as the extended Kalman filter.

Basically, the optimal maneuver is defined as that course which maximizes the trace of the information matrix. If one examines this performance index, it will be noted that there is a trade-off between a course perpendicular to the line-of-sight (LOS) vector for good range measurements and a course parallel to the LOS for better bearing measurements. The resulting optimal course is a compromise between these two objectives. A numerical example that serves to illustrate the theoretical results is presented.

I. INTRODUCTION

This paper describes a technique for determining optimal course maneuvers by an observer who is trying to solve the target motion problem with measurements from a passive, linear array (Murphy et al., 1981). The target motion problem of interest is one that restricts vehicle motion to take place in a plane. To obtain an estimate of target parameters (position and velocity), the observer uses an extended Kalman filter (EKF) with measurements from the linear array. For this acoustic sensor, two noisy time delay measurements are available that are related to bearing and horizontal range to the target. The linear array processing system is capable of providing a time delay measurement proportional to target range because of its ability to determine the amount of wavefront curvature over the aperture of the array. Therefore, unlike the bearings-only tracking problem, which requires an observer maneuver to make the process observable, the availability of this additional measurement makes the process observable without a maneuver. However, it is known from practical experience that an appropriate observer maneuver can improve the convergence of the estimation process. Since the estimation process is basically nonlinear, one should not be surprised to find that the estimation error for the target parameters is sensitive to the actual dynamics of the tracking problem. For the duration of the tracking problem, the observer's track is comprised primarily of constant velocity segments (or legs), while it is assumed that the target has constant velocity (i.e., no maneuvers).

Since it has been established that an observer maneuver is appropriate, it behooves the observer to choose the optimal maneuver. For purposes of this paper, the maneuver will consist of an observer course change. Since the EKF is being dealt with, an applicable performance criterion would be a linear functional, such as the trace, of the information matrix. This paper will show that the observer should choose the course that maximizes the predicted trace of the information matrix associated with the estimation process. If one examines this performance index, it will be noted that there is a trade-off between a course perpendicular to the line-of-sight (LOS) vector for good range measurements and a course parallel to the LOS for better bearing measurements. The resulting optimal course is a compromise between these two objectives.

The general system model and analysis of the performance criterion is presented in Section II. In Section III, the results developed are applied to the specific problem of target motion with time delay measurements. Finally, in Section IV, a numerical example that serves to illustrate the usefulness of the results is presented for a typical vehicle geometry.

II. GENERAL MODEL AND ANALYSIS

Consider a linear discrete time control system:

$$\underline{x}(k+1) = A\underline{x}(k) + B\underline{u}(k), \quad \underline{x}(o) = \underline{x}_0 , \qquad (1)$$

where $\underline{x}(k)$ and $\underline{u}(k)$ are the state and the control, respectively; A and B are constant matrices of appropriate dimensions; $\underline{x}_0$ is a vector Gaussian random variable with known mean $E[\underline{x}_0]$ and covariance P_0.

It is desired to estimate $\underline{x}(k)$ from nonlinear noisy measurements:

$$\underline{y}(k) = \underline{h}[\underline{x}(k)] + \underline{v}(k), \qquad (2)$$

where $\{\underline{v}(k)\}$ is a sequence of vector Gaussian noises with zero mean and covariance N(k).

When $\underline{y}$ is linear, the Kalman filter yields the minimum variance (unbiased) estimate of $\underline{x}(k)$. In this case, the estimation error is independent of the control, and the choice of $\underline{u}(k)$ is irrelevant.

When $\underline{h}$ is nonlinear, the Kalman filter can be extended by linearizing about a nominal trajectory. Let $\hat{\underline{x}}(k/k)$ be the estimation of $\underline{x}(k)$ based on measurements up to k, with $\hat{\underline{x}}(0/0) = E[\underline{x}_0]$. Also, let $\hat{\underline{x}}(k+1/k)$ be the predicted estimate for $\underline{x}(k+1)$. The EKF algorithm linearized about $\hat{\underline{x}}(k+1/k)$ can be summarized as follows.

Define:

$$\hat{\underline{x}}(k/k) = E[\underline{x}(k)] ,$$

$$P(k/k) = E\{[\underline{x}(k)-\hat{\underline{x}}(k/k)][\underline{x}(k)-\hat{\underline{x}}(k/k)]'\}, \qquad (3)$$

$$H(k+1) = \frac{\partial h}{\partial \underline{x}}[\hat{x}(k+1/k)] , \qquad (4)$$

where the prime denotes the transpose.

Then,

$$\hat{\underline{x}}(k+1/k) = A\hat{\underline{x}}(k/k) + B\underline{u}(k) , \qquad (5)$$

$$P(k+1/k) = AP(k/k)A' , \qquad (6)$$

$$K(k+1) = P(k+1/k)H'(k+1)[H(k+1)P(k+1/k) H'(k+1)+N(k+1)]^{-1} , \qquad (7)$$

$$\hat{\underline{x}}(k+1/k+1) = \hat{\underline{x}}(k+1/k)+K(k+1)[\underline{y}(k+1) -\underline{h}(\hat{\underline{x}}(k+1/k))] , \qquad (8)$$

$$P(k+1/k+1) = [I-K(k+1)H(k+1)]P(k+1/k) . \qquad (9)$$

Such an algorithm provides the minimum variance estimate of $\underline{x}(k)$.

It can be seen from eq. (4) and (5) that the observation matrix H(k+1) is determined by $\underline{u}(k)$. Proper choice of $\underline{u}(k)$ in steering the state $\underline{x}(k)$ to increase the observability of the system is therefore important in reducing the updated error covariance P(k+1/k+1).

The magnitude of the error covariance is usually indicated by the norm of the matrix $\|P\|$. Two typical definitions of $\|P\|$ are the trace of P and the determinant of P, i.e., Tr(P) and $|P|$, respectively.

The reduction in error covariance due to information update at k is denoted by $\Delta P(k+1/k)$; i.e.,

$$\Delta P(k+1/k) = P(k+1/k) - P(k+1/k+1). \qquad (10)$$

ΔP is obviously a positive definite matrix. Substituting eq. (9) into (10) yields:

$$\begin{aligned}\Delta P(k+1/k) &= K(k+1)H(k+1)P(k+1/k)\\ &= PH'[HPH'+N]^{-1}HP\\ &= [I+PH'N^{-1}H]^{-1}PH'N^{-1}HP , \qquad (11)\end{aligned}$$

where, in the last two expressions in eq. (11), P = P(k+1/k), H = H(k+1), and N = N(k+1).

The information matrix M is now introduced:

$$M(k+1) = H'(k+1)N^{-1}(k+1)H(k+1) . \qquad (12)$$

Then, ΔP in eq. (11) becomes:

$$\Delta P(k+1/k) = [I+P(k+1/k)M(k+1)]^{-1} P(k+1/k)M(k+1)P(k+1/k). \qquad (13)$$

Thus, at each stage k, $\underline{u}(k)$ can be chosen to maximize $\|\Delta P(k+1/k)\|$ through eq. (4), (5), and (13). The sequence $\{\underline{u}(k)\}_{k=0}^{n}$ then achieves the overall optimization of

$$\sum_{k=0}^{n-1} \|P(k+1/k)\|$$

and results in the best estimation for $\underline{x}(n)$. In other words, the control is optimal in the sense of steering the state along the most observable trajectory.

For the underwater tracking problem, the situation is different from what has been described above. A fundamental assumption is that the target always travels with constant velocity. The observer varies its velocity from one segment to another to more efficiently track and to close with the target. The observer's velocity is now the control for the estimation process. During each segment the observer travels with a constant velocity, while taking measurements to determine the position and velocity of the target. This process is referred to as target motion analysis. A common practice in target motion analysis is to choose a fixed velocity $\underline{u}(k)$, k = 0, 1, ... n-1, more or less arbitrarily for the first leg. At the end of the first leg, k = n, a new velocity is chosen based on the information obtained during the first constant-velocity segment or leg. At this moment, an optimal velocity is to be chosen to improve the convergence of the estimation process and to obtain the best estimate of target position at the end of the second leg.

Let n and m be the number of measurements during the first and second legs, respectively. Minimizing

$$\sum_{k=n}^{m+n-1} \|P(k+1/k)\| ,$$

i.e., optimizing over the second leg, cannot be achieved because $\underline{u}(n)$ is to be determined, and fixed, throughout the second leg without measurement data at $k, k = n+1, n+2, \ldots n+m-1$. While a discrete model is necessary for updating the information matrix when the data measurements are available, a continuous model is convenient for studying the error covariance behavior (rate of decrement) at any time t in (t,t_n) as a function of $\underline{u}(n)$. The information matrix is now evaluated along the predicted trajectory defined by $\hat{\underline{x}}(t,t_n)$.

Let $P(t,t_n)$ be the predicted error covariance for any t in (t_n,t_m), given the initial error covariance $P(t_n/t_n)$. It can then be shown that $P(t,t_n)$ satisfies the following:

$$\frac{dP(t,t_n)}{dt} = AP(t,t_n) + P(t,t_n)A' - P(t,t_n)M(t)P(t,t_n) , \qquad (14)$$

where

$$M(t) = H(t)N^{-1}H'(t) , \qquad (15)$$

and

$$H(t) = \frac{\partial h}{\partial \underline{x}} [\hat{\underline{x}}(t,t_n)], \qquad t_n < t < t_m . \qquad (16)$$

Equation (14) resembles the matrix Riccati equation in linear filtering except that the information matrix M is to be evaluated at $\hat{\underline{x}}(t,t_n)$, which depends on the maneuver at t_n; i.e., $\underline{u}(t_n)$. Obviously, $\underline{u}(t_n)$ is to be chosen to minimize $P(t_{n+m},t_n)$ via M(t). But the latter is a function of t, while $\underline{u}(t_n)$ has to be chosen *a priori*. It is therefore necessary to select a representative time t_k between t_n and t_{n+m}, at which instant M is to be maximized with respect to $\underline{u}(t_k)$. A reasonable choice is $t = t_{n+(m/2)}$; i.e., the midpoint of the second leg. Obviously, if t_k is too close to t_n, the error covariance will drop rapidly right after the maneuver. But such a decrement tends to level off as the observer proceeds in a direction that becomes less and less optimal. On the other hand, if t_k is too close to t_{n+m}, the error covariance will not drop significantly until near the end of the second leg, at which time it will be too late for the observer to gain an overall reduction in error covariance. Therefore, a reasonable—but not necessarily optimal—choice for t_k is $t_{n+(m/2)}$.

Letting $\|P\| = Tr(P)$, and taking the traces of all matrices on both sides of eq. (14), one obtains:

$$\frac{d\{Tr[P(t,t_n)]\}}{dt} = 2Tr[AP(t,t_n)] - Tr[P^2(t,t_n)M(t)] . \qquad (17)$$

It is seen from eq. (17) that $\underline{u}(t_n)$ can be chosen to maximize $Tr[P^2(t,t_n)M(t)]$ at $k = n+(m/2)$ so as to make $Tr[P(t,t_n)]$ decrease rapidly and to achieve a small $Tr[P(t,t_n)]$ at $t = t_{n+m}$.

Suppose P is a 2 x 2 matrix. A different approach is to define $\|P\|$ as $|P|$, the determinant of P. Pre- and post-multiplying of every term in eq. (14) by $P^{-1/2}(t,t_n)$ yields:

$$P^{-1/2} \frac{dP}{dt} P^{-1/2} = P^{-1/2}AP^{1/2} + P^{1/2}AP^{-1/2} - P^{1/2}MP^{1/2}, \qquad (18)$$

where $P = P(t,t_n)$ and $M = M(t)$.

Again, taking the trace of all matrices in eq. (18) and simplifying yields:

$$\frac{d|P|/dt}{|P|} = 2Tr(A) - Tr(PM) . \qquad (19)$$

Minimizing the normalized rate of change is known to be more effective than just minimizing $d|P|/dt$. Thus, from eq. (19), $\underline{u}(t_n)$ can be chosen to maximize $Tr[P(t,t_n)M(t)]$ at $k = n+(m/2)$ so as to make $|P(t,t_n)|$ decrease rapidly and to achieve a small $|P(t,t_n)|$ at $t = t_{n+m}$.

As a result of the above analysis, it should be pointed out that for best estimation results, without knowing $P(t_n/t_n)$, the optimal value of $\underline{u}(t_n)$ should be selected to maximize Tr(M). However, knowing $P(t_n/t_n)$, one should actually choose to maximize the Tr(PM), thus incorporating *a priori* information into the performance criteria. Basically, this implies that the amount of information should be maximized in conjunction with the direction of uncertainties as expressed by the eigenvectors of P.

III. TRACKING WITH TIME DELAY MEASUREMENTS

A specific problem of target tracking with time delay measurements is now considered. Let (R_x,R_y) be the relative position between the observer and the target in the North-East-oriented Cartesian coordinate system. Let $\underline{V}_t = (V_{xt},V_{yt})$ and $\underline{V}_o = (V_o \sin C_o, V_o \cos C_o)$ be the velocities of the target and the observer, respectively, in the same coordinates (see Fig. 1).

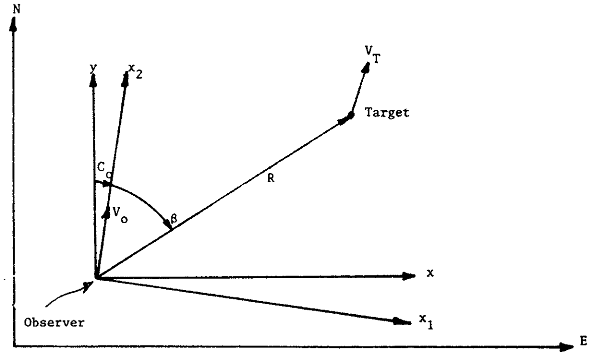


Fig. 1 Various coordinate systems for the tracking problem (z-axis points into page).

It is assumed that target velocity is constant but imprecisely known to the observer at the beginning of the problem. The observer travels at a fixed speed and changes course only at the time of a maneuver.

The state vector is now:

$$\underline{x}'(k) = [R_x(k), R_y(k), V_{xt}(k), V_{yt}(k)] \ . \quad (21)$$

The matrices A and B and the control $\underline{u}(k)$ in eq. (1) are:

$$A = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} -1 & 0 \\ 0 & -1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix},$$

$$\underline{u}(k) = \begin{bmatrix} V_o \sin C_o \\ \\ V_o \cos C_o \end{bmatrix}, \quad (22)$$

where Δt is the uniform time increment between data samples. Let $[x_1(k),x_2(k)]$ be the relative position in the coordinate system with axes parallel and perpendicular to the observer's course (see Fig. 1). Obviously (R_x,R_y) is related to (x_1,x_2) via a rotation by the angle C_o.

To obtain time delay measurements, the observer has six sensors equally spaced on the two sides of the vehicle. Difference of signal arrival times (time delay) between two adjacent sensors on one side is measured. These time delays are indicated by T_1 and T_2.

Using the law of cosines and referring to Fig. 2, it can be shown that:

$$T_1(R_x,R_y) = \frac{1}{c}[R-(R^2+D^2-2DR\cos\beta)^{1/2}]+n_1 \ , \quad (23)$$

$$T_2(R_x,R_y) = \frac{1}{c}[(R^2+D^2+2DR\cos\beta)^{1/2}-R]+n_2 \ , \quad (24)$$

where

c = sound velocity in the water,

$R = (R_x^2+R_y^2)^{1/2} = (x_1^2+x_2^2)^{1/2}$,

$\beta = \tan^{-1}(R_x/R_y) - C_o = \tan^{-1}(x_1/x_2)$, and

D = distance between two adjacent sensors.

Also, n_1 and n_2 are the measurement noises. They are assumed to be independent white Gaussian noises with zero mean and variance of σ^2.

The measurements for the filter consist of the sum and difference of time delays T_1 and T_2; thus, eq. (2) now becomes:

$$\underline{y}(k) = \begin{bmatrix} \tau_+(k) \\ \\ \tau_-(k) \end{bmatrix} = \begin{bmatrix} T_1(k)+T_2(k) \\ \\ T_2(k)-T_1(k) \end{bmatrix} + \begin{bmatrix} v_+(k) \\ \\ v_-(k) \end{bmatrix} \quad (25)$$

where

$v_+(k) = n_1(k) + n_2(k)$,

$v_-(k) = n_2(k) - n_1(k)$;

and, since n_1 and n_2 are uncorrelated, the covariance of v_+ and v_- is $2\sigma^2$.

The observation matrix H for the linearized measurements is:

$$H(k+1) = \begin{bmatrix} \frac{\partial\tau_+}{\partial R_x} & \frac{\partial\tau_+}{\partial R_y} & 0 & 0 \\ \\ \frac{\partial\tau_-}{\partial R_x} & \frac{\partial\tau_-}{\partial R_y} & 0 & 0 \end{bmatrix}_{[\hat{R}_x(k+1/k),\hat{R}_y(k+1/k)]} \quad (26)$$

and the information matrix M is:

$$M(k+1) = H'(k+1)H(k+1)/2\sigma^2 \ . \quad (27)$$

One can express H(k+1) and M(k+1) as follows:

$$H(k+1) = [H_1 \mid 0] \ , \quad (28)$$

$$M(k+1) = \frac{1}{2\sigma^2}\left[\begin{array}{c|c} H_1'H_1 & 0 \\ \hline 0 & 0 \end{array}\right] = \left[\begin{array}{c|c} M_1 & 0 \\ \hline 0 & 0 \end{array}\right] . \quad (29)$$

To determine M, especially Tr(M), it is convenient to utilize the invariance property of M under rotation of axes. Define T to be the transformation between (x_1,x_2) and (R_x,R_y); i.e.,

$$T = \begin{bmatrix} \cos C_0 & -\sin C_0 \\ \sin C_0 & \cos C_0 \end{bmatrix} . \qquad (30)$$

Also, define

$$G(t_{k+1}) = \begin{bmatrix} \frac{\partial \tau_+}{\partial x_1} & \frac{\partial \tau_+}{\partial x_2} & 0 & 0 \\ \frac{\partial \tau_-}{\partial x_1} & \frac{\partial \tau_-}{\partial x_2} & 0 & 0 \end{bmatrix}_{[\hat{x}_1(k+1/k), \hat{x}_2(k+1/k)]} \qquad (31)$$

and write this as $G = [G_1 \mid 0]$. Then,

$$H_1 = G_1 T, \quad M_1 = \frac{1}{2\sigma^2} T' G_1' G_1 T, \quad \text{and}$$

$$Tr(M) = \frac{1}{2\sigma^2} Tr(G_1' G) .$$

To compute G, one can express T_1 and T_2 as functions of (x_1, x_2):

$$T_1 = T_1(x_1, x_2) = \frac{1}{c}[R - (R^2 + D^2 - 2Dx_2)^{1/2}] + n_1, \quad (32)$$

$$T_2 = T_2(x_1, x_2) = \frac{1}{c}[(R^2 + D^2 + 2Dx_2)^{1/2} - R] + n_2, \quad (33)$$

where it is convenient to use (x_1, x_2) instead of (R_x, R_y).

After carrying out all partial differentiation, one obtains:

$$G(k+1) = \begin{bmatrix} \sin\alpha_1 - \sin\alpha_2 & \cos\alpha_2 - \cos\alpha_1 & 0 & 0 \\ -2\sin\beta + \sin\alpha_1 + \sin\alpha_2 & -2\cos\beta + \cos\alpha_1 + \cos\alpha_2 & 0 & 0 \end{bmatrix}_{(k+1/k)} \qquad (34)$$

where α_1 and α_2 are shown in Fig. 2, and (k+1/k) indicates that all variables are to be evaluated at the predicted value for k+1. If P_{11} is defined to be the error covariance of the relative position (R_x, R_y), i.e.,

$$P = \left[\begin{array}{c|c} P_{11} & P_{21} \\ \hline P_{12} & P_{22} \end{array} \right] ,$$

then, eq. (19) becomes

$$\frac{d|P_{11}|/dt}{|P_{11}|} = 2 - Tr(P_{11} M_1) . \qquad (35)$$

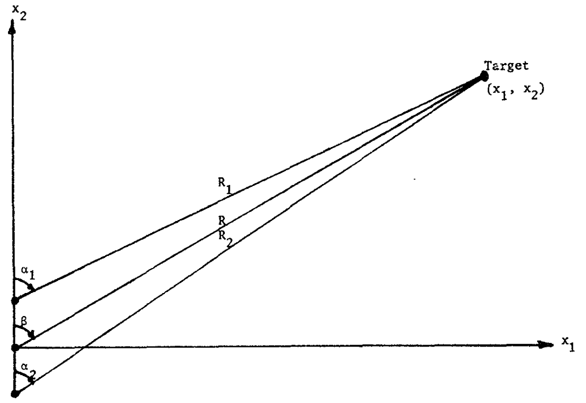


Fig. 2 Target position with respect to linear array in observer's coordinate system.

Now, recall that

$$\hat{\beta}(t, t_n) = \hat{\theta}(t, t_n) - C_0(t_n) , \qquad (36)$$

$$\hat{\theta}(t, t_n) = \tan^{-1}[\hat{R}_x(t, t_n)/\hat{R}_y(t, t_n)] , \qquad (37)$$

$$\hat{R}_x(t, t_n) = \hat{R}_x(t_n/t_n) + [\hat{V}_{xt}(t_n/t_n) - V_0(t_n)\sin C_0(t_n)](t - t_n) , \qquad (38)$$

$$\hat{R}_y(t, t_n) = \hat{R}_y(t_n/t_n) + [\hat{V}_{yt}(t_n/t_n) - V_0(t_n)\cos C_0(t_n)](t - t_n) . \qquad (39)$$

Equation (35) indicates that $Tr(P_{11}M_1) = Tr[P_{11}(t, t_n)M_1(t)]$ must now be maximized at $k = n + (m/2)$. From eq. (34) and eq. (36)-(39), this can clearly be done by choosing $C_0(t_n)$. Although an analytical expression for an optimum $C_0(t_n)$ cannot be found, it is not difficult to determine it by using numerical computation, as will be demonstrated in the next section.

As mentioned at the end of the preceding section, maximizing the trace of the information matrix is a general guide for good estimation. After evaluating $Tr(H'_1 H_1)$ from eq. (34), and performing some manipulation, one obtains:

$$Tr[M_1(t)] = \frac{D^2 \sin^2\hat{\beta}(t, t_n)}{2\sigma^2 c^2 \hat{R}^2(t, t_n)} . \qquad (40)$$

Thus, to maximize $Tr[M_1(t)]$, one must maximize $|\sin\hat{\beta}(t, t_n)/\hat{R}(t, t_n)|$. This implies a trade-off between going broad-sided (i.e, making β as close to 90° as possible) and reducing the relative distance between the observer and the target. Certainly, a more precise approach is to maximize $Tr[P_{11}(t, t_n)M_1(t)]$, in which case procurement of information in both directions is to be weighted by elements of $P_{11}(t, t_n)$.

IV. A NUMERICAL EXAMPLE

To illustrate the results in section III, the following scenario is presented. The observer starts from the origin at $t_0 = 0$, while the target is at an initial position of $[R_x(0), R_y(0)] = [0, -15000 \text{ yards}]$ (see Fig. 3). The speeds of the observer and the target are 10 and 20 knots, respectively.

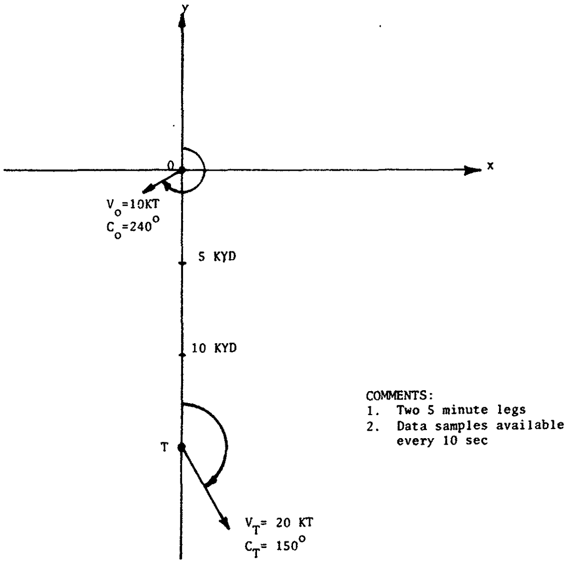


Fig. 3 Observer-target geometry for the numerical example.

During the first leg, the observer's course is chosen to be $C_0 = 240°$. An optimum course for the second leg is to be determined. The target is assumed to maintain a constant direction of $C_T = 150°$ throughout the entire operation; i.e., $(V_{xt}, V_{yt}) = (20 \sin 150°, 20 \cos 150°)$.

The duration of each leg is 5 minutes with sampling time $\Delta t = 1/6$ minute and, hence, $n = m = 30$. It is assumed that $\sigma^2 = 0.25 \times 10^{-10}$, and the initial error covariance is given by:

$$P(0/0) = \begin{bmatrix} 0.162\text{x}10^9 & 0 & 0 & 0 \\ 0 & 0.162\text{x}10^9 & 0 & 0 \\ 0 & 0 & 225 & 0 \\ 0 & 0 & 0 & 225 \end{bmatrix}.$$

At the end of the first $t = t_{30} = 5$ minutes the upper triangular portion of the error covariance matrix is reduced to:

$$P(300/300) = \begin{bmatrix} 0.935\text{x}10^4 & -0.508\text{x}10^5 & 140.0 & -171.0 \\ & 0.278\text{x}10^6 & -25.6 & 31.5 \\ & & 0.088 & -0.0867 \\ & & & 0.106 \end{bmatrix}$$

with estimated target speed and course of 20.11 knots and 150.13°, respectively.

One then computes $\text{Tr}[M_1(45/30)]$ and $\text{Tr}[P_{11}(45/30)M_1(45/30)]$ for different values of $C_0(30)$ ranging from 0° through 360°. It can be seen from Fig. 4 that there are two maxima for $\text{Tr}(M_1)$—at $C_0 = 84°$ and $C_0 = 253°$. This is due to the fact that the relative bearing $\beta_1(45/30)$ corresponding to $C_0 = 84°$ is exactly the negative of the bearing $\beta_2(45/30)$ corresponding to $C_0 = 253°$. Hence, from eq. (40), the values of $\text{Tr}(M_1)$ are equal at the two angles of C_0. Fig. 4 also shows that $\text{Tr}[P_{11}(45/30)M_1(45/30)]$ has a unique maximum at $C_0 = 84°$.

Now, the extended Kalman filter (EKF) processing is carried out using computer simulation and the Monte Carlo method. The results for three different values of C_0 in the second leg are compared. Table 1 shows that, at $C_0 = 84°$, $\text{Tr}[P_{11}(60/60)]$ is strictly better than at 240° and 253°. Further computation also shows that $C_0 = 84°$ is indeed the optimum course that renders the minimum value for $\text{Tr}[P_{11}(60/60)]$. This confirms the assertion in the preceding section that maximizing Tr(PM) is the actual criterion.

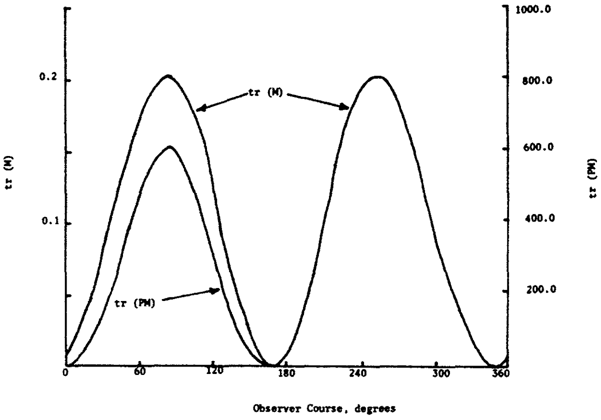


Fig. 4. Variation of trace of M and PM at t=7.5 minutes as a function of observer course.

TABLE 1 Error Covariance from the EKF for the Range and Range Components (Average for 30 Monte Carlo Simulation Runs)

Observer Course		Error Covariance at 10 Minutes (yards²)		
Leg 1	Leg 2	R_x	R_y	R
240	240	28100.	261000.	289100.0
240	84	106.	3210.	3160.0
240	253	25300.	231000.	256300.0

Figures 5 and 6, respectively, compare the time histories of the EKF results for the x and y range components with the optimal

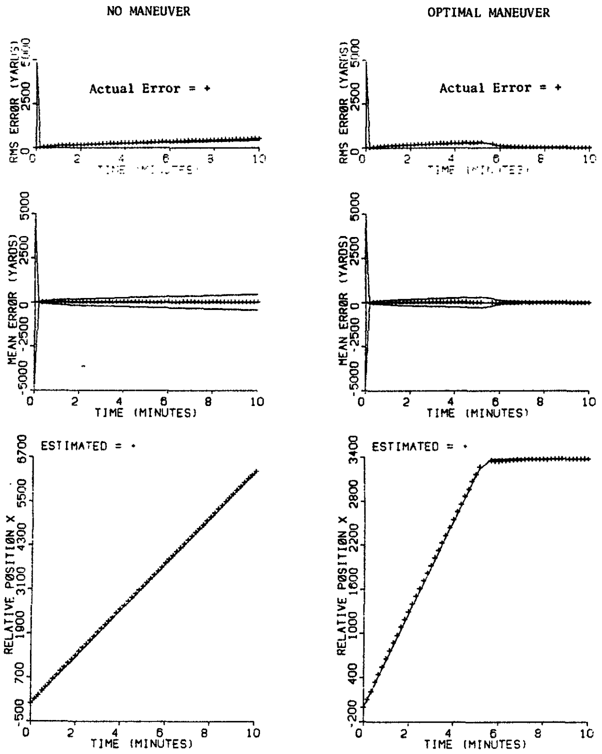


Fig. 5. Comparison of EKF results for the x-component of range with the optimal maneuver and without a maneuver (averaged results for 30 Monte Carlo simulation runs).

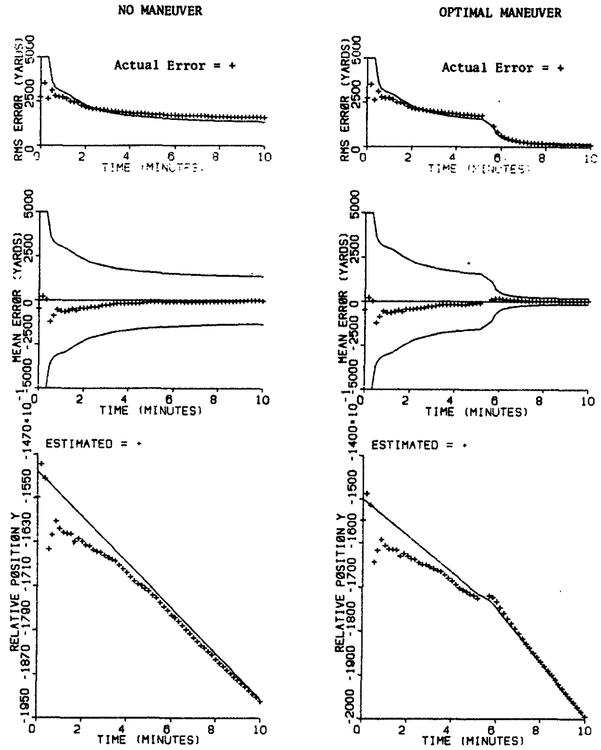


Fig. 6. Comparison of EKF results for the y-component of range with the optimal maneuver and without a maneuver (averaged results for 30 Monte Carlo simulation runs).

maneuver and without a maneuver. Because of the particular target geometry chosen, it is seen in Fig. 5 that the x-component of range can be estimated quite closely, and the error is somewhat insensitive to observer maneuvers. However, the y-component of range is known or measured with a larger degree of uncertainty, as evidenced by the two uppermost plots in Fig. 6. Without a maneuver, the error covariance associated with R_y has a small rate of decrease after the initial transient behavior of the filter. The optimal maneuver at t = 5 minutes causes the error covariance to decrease rapidly to zero, and the actual estimate of R_y is improved, as evidenced by comparing the lowermost plots in Fig. 6.

V. CONCLUDING REMARKS

This paper has shown that in general an optimum maneuver at the end of one leg (for the purpose of minimizing the error covariance at the end of the next leg) can be achieved by maximizing the predicted value of the trace of information matrix M. In terms of estimation with time delay measurements, this means that the observer must seek a trade-off between going broadside and reducing the relative distance.

When a priori knowledge about the error covariance P is to be taken into consideration, the observer should seek to maximize the predicted value of Tr(PM). This implies maximizing the predicted value of Tr(M) and minimizing the angle between eigenvectors of P and M corresponding to their minimum eigenvalues.

REFERENCES

Murphy, D.J., B.W. Guimond, and D.W. Moore (1981). Linear array tracking of a maneuvering target. Fifteenth Asilomar Conference on Circuits, Systems, and Computers. pp. 416-423.

A SUMMARY OF RESULTS IN MULTIPLATFORM CORRELATION AND GRIDLOCK

M. A. Kovacich

COMPTEK Research, Inc., 1C0 Corporate Place, Suite B,
Vallejo, California 94590

Abstract. Near term, automated (1984) solutions to the gridlock and intership track correlation problem in the naval task force are discussed. The force tracking problem and limitations in current fleet performance are reviewed. It is shown, based upon simulation evaluation, that fleet performance can be substantially improved, even in the constrained core/timing environments of current fleet computers, if the intership bias estimation and correlation functions operate with a high degree of internal coupling and operate continuously.

INTRODUCTION

With the increased reliance on automatic tracking systems, the increased size and dispersion of the naval task force and the growth in expected complexity of the tactical environment, it has become especially crucial to the coordinated operation of the task force to maintain, automatically and continuously, accurate intership coordinate transformations (gridlock) and reliable intership track pairings (correlation). Gridlock is a well known, well documented problem that has remained unsolved for over 20 years. The correlation problem has attracted considerable attention in recent years due to the increased volume of tracks generated by automatic tracking systems that must be exchanged over the digital data link. The purpose of this paper is to discuss some results in the design of a coupled gridlock-plus-correlation system, denoted a force track alignment (FTA) system, and to show that a near term (1984) solution exists in the constrained hardware environments of current fleet computers if a high degree of internal coupling between the intership bias estimation and correlation functions is effected and if the system operates continuously and automatically. In addition, it is shown that substantial performance improvement over existing systems can be obtained even when constrained to the use of simple, definitely suboptimal, algorithms.

CORRELATION AND GRIDLOCK - AN OVERVIEW

Generally, the naval task force is dispersed over hundreds of square miles with the high value units, e.g. the carrier, in the center of the force and AAW and ASW units, e.g. cruisers and destroyers, providing a protective screen around the high value units. E2C aircrafts (AWACs) provide airborne surveillance coverage and a varied mix of other units carry out specialized missions within and outside the force boundary. The surface and airborne platforms exchange data over a digital data network that operates, in many cases, in a roll call fashion with each unit, in turn, broadcasting on the communication net the tracks for which he is responsible; that is, tracks for which he has the most accurate data.

The platforms participating in the net increasingly rely on automated tracking systems to generate their vehicular picture of the surveillance volume. A large volume of track data is generated by such systems since the tactical environment is richly populated with detectable friendly, ambient and hostile traffic. Track stores in current automatic tracking systems often become saturated unless special safeguards such as limiting surveillance coverage (sectoring) are enacted. Estimates of future track volume exceed by many times current capacities. It thus becomes especially important for the coordination of the fleet assets that this high volume of track data emanating from each unit on the link net be accurately transformed and correlated between the units. Otherwise, the force picture degrades (e.g. multiply defined tracks, miscorrelated tracks, link saturation) to the point that it cannot support the force C^2 process.

The flow of track data in the force tracking environment is shown in Fig. 1 for the case of three participating units on the data link. Each platform, by means of its suite of automatic and manual tracking systems, establishes a local track file comprising its view of the tactical environment. The local track file is correlated with the remote tracks broadcast by other platforms to form the force track file. The force track file, conceptually, is a single track file that is represented on each participating unit in coordinates proper to each participating unit. The force track file is the composite view of the tactical environment derived by combining and correlating the distributed local track files of the participating units.

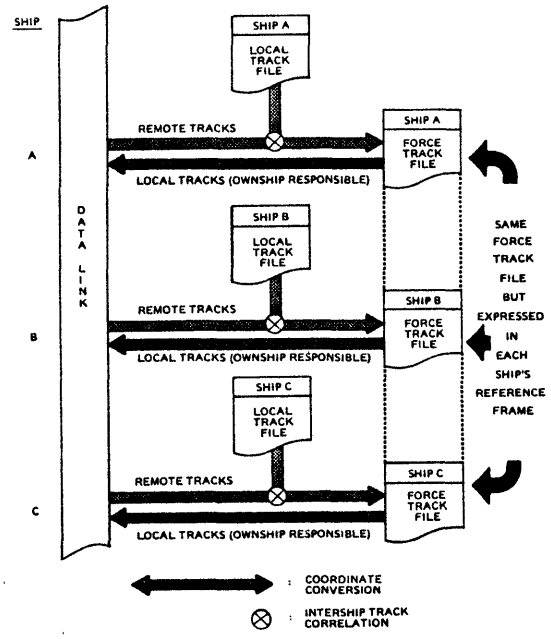


Fig. 1. Track Data Flow in Force Tracking Environment.

To complete the track data flow, each participating unit broadcasts over the data link those tracks for which he holds the most accurate data. Ideally, the force track file then is in 1-1 correspondence with the vehicles in the environment with only a single participating unit reporting on a vehicle at any one time.

The coordinate conversions that are carried out to interchange force track data between platforms is shown in Fig. 2. Before describing the coordinate transformations, it is important that two concepts be clearly defined: platform coordinate frame, or simply platform frame and a platform's earth coordinate grid, or, simply a platform's earth grid. The platform frame is a three dimensional cartesian coordinate frame on the earth's surface whose X-Y plane is tangent to the earth's surface at the point of contact. The platform frame is completely characterized by specifying the latitude, and longitude of the platform and the orientation of the X-Y plane with respect to north. The platform's earth grid, on the other hand, is the grid of latitude and longitude lines on the earth's surface. The platform frame and platform earth grid are related. The platform's earth grid defines a family of coordinate frames one at each point on the grid. Also, each platform measures vehicle kinematics with respect to its platform frame, which implicitly defines the platform's earth grid. The problem faced in the force tracking environment is to move track data between platform coordinate frames when the platforms do not agree on the same earth coordinate grid either because of translational offset due to navigational bias, north orientation offset due to intership radar bias or both.

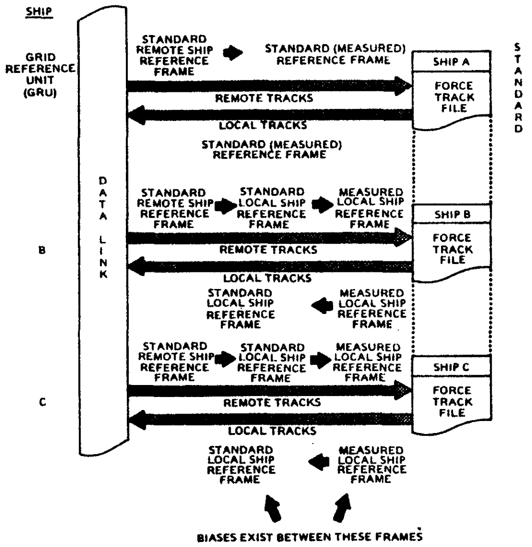


Fig. 2. Coordinate Conversions in Force Tracking Environment.

The coordinate transformations are carried out through the use of an intermediate earth coordinate grid, denoted the standard grid, that is measured by a specified but arbitrary participating unit, named the gridlock reference unit (GRU). It is important to note that the standard earth grid need not be the same as the true earth grid in order to carry out its role as intermediary in coordinate transformations. It simply provides a common grid to which each platform aligns.

Each platform has two coordinate frames: its standard frame and its measured frame. The platform's standard frame is a frame that is consistent with the standard earth grid, that is, the X-Y plane has the same orientation as the latitude/longitude lines of the standard grid and the position of the frame (its latitude, longitude values) is specified with respect to the standard grid. The platform's measured frame is the frame actually used by the platform to register data generated locally and is the coordinate 'language' of the platform.

Coordinate transformations between noncolocated standard frames is a three dimensional rotation plus translation that requires an estimate of the intership latitude and longitude bias. The transformations between the standard and measured frames of a platform consists of a rotation that requires an estimate of the intership radar alignment bias.

The steps used in receiving track data are as follows: track data arrives registered in the standard platform frame of the reporting platform; it is transformed into the standard platform frame of the receiving platform and then transformed into the measured platform frame of the receiving platform. On transmission, the track data is transformed from the measured frame of the reporting platform into the standard frame of the reporting platform and then broadcast. The GRU is handled differently in that it carries out the standard remote frame to standard local frame on reception and performs no conversion on transmission. The net effect is that all data sent or received over the data link are registered in a standard platform frame on the standard earth grid.

Differences between a platform's measured and standard frame are statistical in nature and are due to navigational bias and intership sensor misalignment bias between the platform and the GRU. Fleet experience, reported in Akita (1980) has shown that relative navigational errors can be as high as 10 dm. Intership alignment biases have been as much as 10° and live radar data analysis in Miller (1980) have found radar azimuth dependent radar bias in two AN/SPS-39 radars with an amplitude of 2°. Essentially, the gridlock process is devoted to maintaining an accurate estimate of these biases. This is done through the measurement and filtering of biases existing between track data from the GRU and track data from the platform on mutually held tracks. Accurate intership track correlation between local and GRU generated tracks thus enters as a basic requirement in establishing mutual tracks to be used in bias estimation.

CURRENT FLEET PERFORMANCE

Current fleet performance in the areas of gridlock and correlation are universally considered to be marginal today and inadequate for the future. The litany of problems is becoming standard:

- Intership track correlation is a manual process that cannot keep apace with the volume of track data generated by automatic trackers. The result is that automatic sensors are sectored to reduce overlapping coverage which has the effect of reducing track continuity.
- Bias estimation is generally a manual process that requires considerable operator attention (roughly 20% of the operator's attention in survey quoted in [3]) to maintain consistently accurate estimates. Not all biases are estimated, e.g. in manual gridlock, the intership sensor alignment bias is not estimated at all. The bias estimates are not updated often enough. (The same survey indicates that a 3.5 dm gridlock error is tolerated before updating the biases.) The 'automatic gridlock' option - manual correlation and automatic bias estimation for a fixed period of time - requires considerable force coordination so is not used. The net effect is that the bias estimates, in the fleet today, are highly error prone.

It has become obvious that the gridlock problem can be solved only if the bias estimation and intership track correlation are treated as a system and the system operates automatically and continuously. This has been a major feature of the design philosophy in the development of a force track alignment system for near term (1984) implementation to be discussed next.

FORCE TRACK ALIGNMENT

At a high level, the force track alignment system aligns the platform's local track file to the force track file. Alignment requires correlation of local tracks to force tracks and transformation of the local tracks to the force tracks to effect the alignment. The flow of data in a force track alignment system is shown in Fig. 3. The coordinate conversion function carries out the coordinate transformations shown in Fig. 2. The intership correlation function manages the correlation linkages between local and force tracks by correlating new local tracks, correlating new remote force tracks and monitoring all previously paired (mutual) and unpaired force tracks for continued validity. The bias estimation function filters the biases existing between local and remote track data on mutual tracks with the GRU. These bias estimates are then used in the coordinate transformations between standard and measured platform frames and between local and remote standard frames. Finally, the FTA system, once initialized, operates continuously and automatically.

The FTA system described above has a low level of functional coupling between the correlation and bias estimation functions. The bias estimation function is coupled to correlation through the track data on mutuals with the GRU and correlation is indirectly coupled to bias estimation

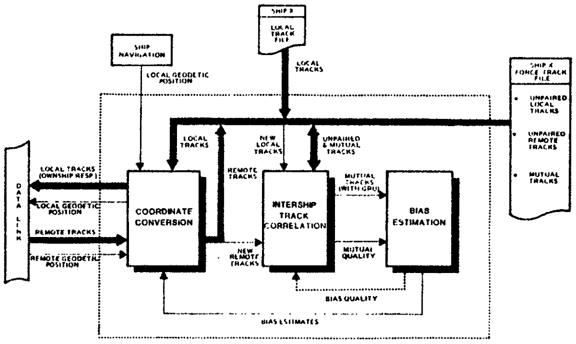


Fig. 3. **Data Flow in Force Track Alignment System.**

through the bias estimates used in coordinate conversion. The internal functional coupling between correlation and bias estimation can be increased if each function generates and exchanges quality figures. In particular, the correlation function could provide the level of confidence that each mutual track sent to the bias estimation function consists of correctly paired tracks and the bias estimation function could provide the accuracies of the filtered bias estimates. These data exchanges are also shown in Fig. 3.

The advantages in increasing the functional coupling in this manner are numerous: The correlation function can more accurately size the correlation gates used to generate possible correlation candidates if the variances in the bias estimates are known. Likewise, the correlation likelihood ratios, used to choose from a number of possible correlation candidates, are more accurate since they account for more sources of error between local and remote track data. Finally, the bias estimation function can use the correlation quality figures to choose the best mutual tracks to use in estimating the biases.

The purpose of this report is to show the performance improvement obtained by increasing the internal coupling in an FTA system.

SCOPE OF EVALUATION EFFORT

The evaluation effort was to determine the performance of eight candidate FTA systems for near term (1984) implementation in current shipboard computer systems, specifically, the AN/UYK-7 computer. The full effort is documented in [3]. The design philosophy used in determining candidate systems was driven by the substantial core and timing constraints of the shipboard computer and by the intuition that substantial gains in gridlock and correlation performance could be obtained, even by simple, definitely suboptimal algorithms, <u>if</u> the bias estimation and correlation algorithms operated as a system (high internal coupling) and operated continuously.

When it is considered that the bias estimation is a fairly simple tracking problem (slowly varying biases with a high update rate), it is conceivable that a simple constant gain bias filter, for example, would be attractive especially in a core and timing constrained environment.

Eight candidate FTA systems were considered and were obtained by choosing one option in each of three algorithm categories:

- Single Pass Correlation vice Mutlipass Correlation
- Constant Gain Bias Filter vice Least Squares Bias Filter
- Low Internal Functional Coupling vice Higher Internal Functional Coupling.

Single pass correlation refers to an algorithm which makes a correlation or no correlation decision on a single opportunity. A multipass correlation logic, on the other hand, allows for a wait decision in ambiguous situations (many likely correlation candidates) to allow the tracks to collect more data and exhibit characteristics that will help resolve the ambiguity (e.g. one candidate flies out of the correlation gate). The constant gain bias filter refers to a single pole filter that operates separately on latitude, longitude and azimuth bias. The innovation sequences for the three uncoupled filters are the average latitude separation between mutual track data, the average longitude separation and the average azimuth deviation, respectively. The least squares filter likewise operates separately on the latitude, longitude and azimuth bias; and uses the same innovation sequences. Levels of internal functional coupling are defined as follows: low internal coupling refers to an FTA system in which correlation and bias estimation interact through track data on mutual tracks and bias estimates; higher internal coupling refers to an FTA system in which correlation and bias estimation interact, in addition, through the confidence level for each mutual track and through accuracy figures for the bias estimates.

Each of the eight FTA systems were examined by stressing each system with the same vehicle and intership bias (navigational and intership radar biases) scenarios. The mechanism used for examining FTA performance was a Monte Carlo simulator that models multiple platforms on a spherical earth surface that observe and track straight line targets by means of a radar sensor model with automatic tracking logic. The platforms exchange track data over a simulated data link. The simulator is instrumented to maintain indicators of the decision performance of correlation and estimation performance of the bias estimation filters.

The results to be discussed in the next section concern the performance improvement (e.g. lower miscorrelation rate, improved bias estimation) obtained by increasing the internal coupling of a multipass correlation, and constant gain bias estimation filter.

DESCRIPTION OF RESULTS

The performance of the two FTA systems with respect to a single difficult vehicle and bias scenario is shown in Fig. 4. The scenario, itself, is presented in the center of the figure and consists of two platforms separated by 150 dm. The platform on the right is given an initial 3 degree azimuth bias. Five vehicles exist in the environment: 2 are widely separated and stationary, and three form a tight cluster of high speed, crossing vehicles. The combination of large initial azimuth bias, large intership and ship-to-track distances and a tight cluster make the scenario very difficult, though not worst case. Correlation performance is indicated to the left and right of the scenario picture. Correlation performance is measured by

- the number of miscorrelations
- the number of incorrect no correlations (dual designations)
- the average number of radar scans (8 sec period) before correlation decisions are made
- the number of incorrect decorrelations
- the number of tracks not decorrelated but should have been.

Bias estimation performance is indicated at the bottom of the figure and consists of

- the average longitude, latitude and azimuth bias estimate after each net cycle (4 sec period) for 56 net cycles
- the standard deviation in the latitude, longitude and azimuth bias estimate on the 56th net cycle.

All performance numbers reported in the figure refer to 10 Monte Carlo trials.

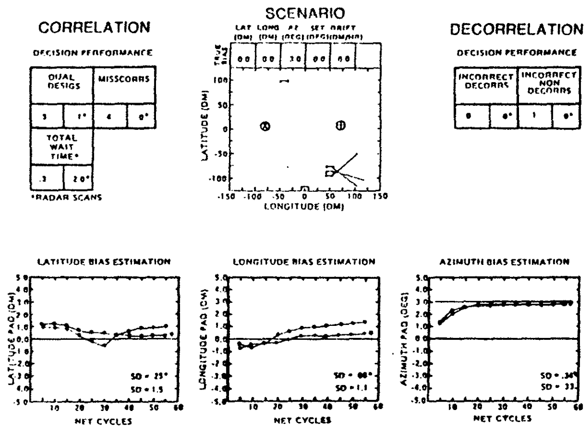


Fig. 4. **Comparison of Two FTA Systems with Multipass Correlation and Constant Gain Bias Estimation. Numbers with asterisks refer to system with higher internal coupling.**

Consider first the FTA system with low internal coupling. Because the correlation function does not have available the variances in the bias estimates, it assumes worst case biases in forming correlation gates and zero bias variance in forming the correlation likelihood ratios. Because the bias estimation process does not have available the quality of each mutual track it uses all mutual tracks. The effect of these features on performance can be seen in Fig. 4.

The correlation algorithm, since zero bias variance is incorporated into the likelihood ratios, tended to miscorrelate tracks in the cluster. This is due to the initial 3° azimuth bias that shifts the local track picture to cause incorrect local and remote track pairs to be nearest neighbors. The effect of the miscorrelations shows up in the bias estimation performance. After 56 net cycles, the latitude bias averaged (over 10 Monte Carlo trials) 1 dm, the longitude bias averaged 1.5 dm and the azimuth bias equaled the correct value of 3°. The latitude and longitude bias averages should have been 0 dm. so a tendency for the miscorrelated tracks to 'draw off' the bias estimates is apparent. What is also significant is the instability in the latitude and longitude bias estimates after 56 net cycles. The standard deviation in the latitude bias estimates was 1.5 dm and for the longitude bias estimates was 1.1 dm. These results indicate the uncertainty in the bias estimates from trial to trial due to the presence or absence of miscorrelations from trial to trial. Note that decorrelation eliminated 3 of the miscorrelations but left 1 still correlated after 56 net cycles. These uncorrected miscorrelations and the fact that the vehicles cross at high speed are the basis for the inaccurate and unstable bias estimation performance.

Consider next the FTA system with a higher level of internal coupling. The correlatin algorithm adjusts the correlation gates and modifies the correlation likelihood ratios to include the bias variance data. The bias estimation process uses only the high confidence correlations (mutual tracks) for estimating bias. The improvement in performance by using these quality measures is apparent in Fig. 4.

The correlation algorithm, since it incorporates an initial large bias variance in its processing, tended to postpone correlation decisions in the cluster (note increase in average time spent before making a correlation decision) until the bias variances were reduced. The isolated tracks were correlated immediately and used to begin estimating the biases. As the bias variance reduced, the correlation algorithm then made pairings. It is important to note that the pairings are made when the shift in the local picture due to the initial azimuth bias has been reduced and incorrect local and remote track pairings tend to become nearest neighbors. The net effect is that correlation performance improved markedly. As a result, the bias estimation performance is also significantly improved. The latitude and longitude bias estimates are more accurate (within .5 dm of their correct value) and the stability of the estimates is much greater the standard deviations of the latitude and longitude bias estimates after 56 net cycles are much lower).

It should be noted that these results are typical of the trends observed in the full collection of 12 experiments with different vehicle and intership bias scenarios.

CONCLUSIONS

The force tracking problem (gridlock and intership track correlation) and the limitations of current fleet systems to effectively handle the problem is discussed.

The definition of an automated system of functions (coordinate conversion, intership track correlation and bias estimation) is introduced and denoted a force track alignment system. The paper discusses the performance improvement obtained if the internal functional coupling between correlation and bias estimation in FTA is increased to include the exchange of quality figures for correlation decisions and bias estimates.

The paper also supports the thesis that substantial gains in gridlock and correlation performance can be obtained over current systems, even by using simple, definitely suboptimal algorithms, if the correlation and bias estimation algorithms operate as a system (high internal coupling) and operate continuously.

REFERENCES

1. Akita, R.M. and Schemesky, W.C. (1980). RIMPAC 80 Shipboard Navigation Performance Assessment Utilizing NAVSTAR GPS (U). NOSC Technical Report 594.
2. Miller, J.T. (1980). Gridlock Data Analysis - Initial NRL/APL Data (U). The Johns Hopkins University/Applied Physics Laboratory. Technical Report F2E-0-478.
3. Force Track Alignment Design Study. COMPTEK Report No. 1081G-1200-1 (1981).

AN APPROACH TO THE DATA ASSOCIATION PROBLEM THROUGH POSSIBILITY THEORY

I.R. Goodman

Surveillance Systems Department, Naval Ocean Systems Center,
San Diego, California

Abstract. An algorithm is developed for treating in a systematic and unified manner the nongeolocation attribute aspect of the multitarget data association problem. The technique is based upon three types of rigorous results. The first establishes that fuzzy set or possibilistic modeling of uncertainties is essentially a weakened form of random subset (or subset selection) modeling. This allows, with some information loss, the replacement of hard statistical descriptions by corresponding possibilistic ones, in addition to the many descriptions which may already be present in possibilistic form due to vague or semantic based sources. In addition, these results also permit -with some caution-the converse process of changing possibilistic descriptions into random set ones. The second category of results directly form the basis of the functional structure of the algorithm. These derive uniformly most accurate possibilistic estimators with respect to input information consisting of possibilisitc descriptions linked through ordinary two valued logical connectives -usually, conjunction. The third result,which may be used to justify use of the algorithm,is that the technique can be shown to have the same structure as a posterior distribution function relative to a fuzzy set version of Bayes' theorem.

INTRODUCTION

Most of the past and current approaches to the data association or contact correlation problem in ocean surveillance rely upon Bayesian or related statistical techniques. Some exceptions to this may be found in the heuristic systems mentioned in the two editions of the Naval Ocean-Surveillance Correlation Handbook (Wiener and others,1979; Goodman, Wiener, Willman,1980). On the other hand, Bowman's work (1981) and that of Bowman and Morefield (1980) are good examples of the total Bayesian approach to this problem. However, it appears among the various types of data inputs available, a non-negligable percentage involve vague or linguistic based information which is not readily amenable to simple statistical modeling, without making a great many assumptions that may not be really appropriate. In addition, the natural domains of values that the attributes can assume may often represent complicated and overlapping outcomes which cannot be assigned meaningful disjoint and exclusive elementary event probabilities. Thus at the very least in these situations, random subset-or equivalently, subset selection- modeling is called for. If, in addition, joint coverage probabilities for such a random set is not specified, it is natural to consider only the one point coverage probabilities and not specify any further the structure of the random set. But from earlier studies (see Goodman, 1982 a , for example), it can be shown that a fuzzy set or possibilistic approach for the above situation leads to the same result. Thus, one is lead to an alternative , yet related, approach to the treatment of uncertainties. More generally, a typical situation will consist of a mix of information,with some more naturally handled by statistical techniques,and some by modifications of such procedures, which in effect are possibilistic ones.

As an example of the above statements, consider the following four attributes which are commonly involved in informational inputs relative to tracking : A_1 = class , A_2 = frequency of signal at its source, A_3 = ship mode, and A_4 = geolocation with confidence ellipse. The natural domains of these attributes are typically : $\mathrm{dom}(A_1) = \{C_1,..,C_m\}$, each C_k a label for a category of ship ; $\mathrm{dom}(A_2)$ = interval $[0,M_o]$, where M_o is some suitably chosen upper bound(in hz.); $\mathrm{dom}(A_3) = \{D_1,..,D_n\}$, each D_k being a label for a mode of operation, noting the highly overlapping flavor in general possessed by the C_k's and by the D_k's , where some could actually represent subcategories with respect to others; $\mathrm{dom}(A_4) = \{(p,E_p) \mid$ p any point in R^2

(in some convenient latitude-longitude coordinates) and E_p any fixed level confidence ellipse centered at p} . Suppose now that two suitably updated(to a common time) track histories i and j are available containing information which can be distributed into the four types of attributes described above. It is desired to determine whether these two track histories, derived from different information sources, indeed belong to the same target or not. More generally, it is desired to determine the distribution (in either the classical probabilistic meaning or the possibilistic one) of the levels of data association between i and j,given all relevant information, and in turn use this, if a hard decision is required, to decide whether the two histories "correlate" or not. The "relevant information" mentioned above can be conveniently divided up into three parts :

1. Observed data.
2. Prior known distributions of matches between observed and true attribute values.
3. Prior known relations between the levels of matching outcomes for any attribute or group of attributes, relative to any pair of track histories, and the consequent correlation level.

Thus, some function or statistic (in the extended sense to include possibilities as well as probabilities) of the relevant information -or of part of the information, such as only involving the first two categories listed above- is sought which will estimate the unknown correlation between the two track histories.

This paper develops an algorithm for systematically treating both nongeolocation and geolocation attribute information in obtaining the posterior distribution of correlation between two track histories. All three kinds of relevant information described above are combined in the algorithm. The entire procedure is based upon three types of theoretical results:

(a) Fuzzy sets and their operators correspond in a natural way to random sets and their operators such that fuzzy set or possibilistic modeling may be considered as a weakened form of probabilistic modeling. In addition, mappings exist which enable one to interchange possibilistic modeling with probabilistic modeling.

(b) Given input information consisting of an ordinary two valued logical combination of possibilistic descriptions of some unknown parameter vector (typically, conjunctions or disjunctions, or even combinations of these with negations possible also), uniformly most accurate (in the sense of nested sets) possibilistic estimators exist which naturally correspond to the input information structure.

(c) A possibilistic or fuzzy set form of Bayes' theorem may be derived, which in turn can be used as the structure through which the posterior distribution of correlations is obtained. With the proper identification of relations in 3 above with conditional distributions, the optimal estimator according to (b) will coincide with the posterior distribution according to (c).

This paper is a further development of earlier work on the attribute problem. (See Goodman, 1981 .) Two other contributions of fuzzy set/possibility theory to military problems should be mentioned : Watson and others (1979) on command and control decision making and Dockery (1982) on designs of general military information systems.

BACKGROUND : FUZZY SET SYSTEMS

Although it is not possible to condense fuzzy set theory in terms of all of its major thrusts here, some relevant highlights can be touched upon. The basic building block is the membership function

$$\Phi_A : X \to [0,1] \quad , \qquad (1)$$

defining fuzzy subset A of base space X.

By making the range of Φ_A be a subset of $\{0,1\}$, A becomes a set in the ordinary sense. Operations among fuzzy susets of a base space extend those of ordinary subsets of the space. For example, one can define fuzzy intersection between two fuzzy sets by use of the pointwise operator min applied to the corresponding membership functions. On the other hand,one could just as well define other operations on fuzzy sets which might also reasonably be called fuzzy intersection since they also reduce to ordinary intersection when the fuzzy sets involved are also ordinary ones. One such example is the operator prod (for pointwise product operating on the corresponding membership functions). Similarly for fuzzy union, max or probsum (probability sum, where $\mathrm{probsum}(a,b) \triangleq a+b-ab = 1-(1-a)(1-b)$) can serve as definitions, from among an infinity of choices. Consider also fuzzy complement. A natural choice is the operator $1-(\cdot)$, but as in the above cases, many other different definitions could be used. Which ones to choose? Obviously, this basic problem must impinge upon all uses of fuzzy set theory ; a partial solution to this will be briefly considered below. (See also Goodman, 1982 a)

However the problem of obtaining fuzzy set membership functions is relatively simple, provided that the domain of discourse or base space is properly specified. For example,the fuzzy set representing the attribute "tall" clearly must be some nondecreasing or monotone increasing function over its domain. But the slope and increase is dependent upon whether "tall" refers to adult males now living in the Unites States, or to immature females who resided in India during the eighteenth century , or to ships , etc. Using proper sampling or survey techniques in conjunction with suitable parameterization, analagous to that employed in modeling probability distributions, empirical membership functions may also be constructed. (See the survey of procedures by Dubois and Prade, 1980,pp.255-264.)(Another modeling approach to fuzzy set membership functions can be through the empirical one point coverage functions of the equivalent random sets, the latter topic to be discussed later.)

One approach to the problem outlined above concerning nonuniqueness[1] of fuzzy set definitions is as follows: First, attempt to abstract the essential meaning-and no more than that- of complement (or negation), intersection (or conjunction), and union (or disjunction). In the case of the last two operators, a natural family of operators has been proposed and investigated by some researchers: the t-norms and t-conorms, respectively. (See Klement,1981 and Goodman,1982a, for further details.) Then for any triple of operators of interest

$$F \overset{d}{=} (\psi_n , \psi_\& , \psi_{or}) , \qquad (2)$$

where ψ_n is usually chosen to be $1-(\cdot)$ for complementation (though not necessarily so restricted) , $\psi_\&$ is a t-norm, and ψ_{or} is a t-conorm , compound fuzzy set definitions may also be defined, with structure not dependent on the specific choice of F. This leads to unified definitions for implication, equivalence, universal and existential quantifiers, subset relations, projections, and general functions and arithmetic operations, among many other concepts. Multivalued logic, as a formal extension of ordinary two-valued logic plays the central role in the above construtions. (See Goodman,1982a for an example of this approach to the construction of general fuzzy set systems.) Second, determine from theoretical considerations which subcollection of fuzzy set systems F leads to interpretation in terms of probability theory. As mentioned later, two families (the semi-distributive DeMorgan and the larger class, the J-copula DeMorgan) can be chosen for possible F . Specifically, these are characterized by their weak homomorphic relations to corresponding random set systems. "Weak" as used above means that equality as is usually used in the concept of homomorphism is replaced by (the weaker) equality with respect to one point coverage probabilities. Finally, use empirical procedures such as moment matching techniques to determine the most appropriate F from the reduced collection.

BACKGROUND: CONNECTIONS BETWEEN FUZZY SET SYSTEMS AND RANDOM SET SYSTEMS

The next set of results comprise type (a) basis for the correlation algorithm , where fuzzy set and random set descriptions may be interchanged (not without some information loss or increase) See Goodman,(1981,1982a), for background and mathematical details.

Define for any fuzzy subset A of X ,

$$S_U(A) = \Phi_A^{-1}[0,1] \quad . \qquad (3)$$

$S_U(A)$ is a random subset of X with all outcomes being nested with respect to each other, where U is any random variable uniformly distributed over [0,1]. Note the special case when Φ_A is monotone and the relation to r.v.'s. Extend the above definition, by considering any stochastic process $\underset{\sim}{U} \overset{d}{=} (U_j)_{j \in J}$ of uniform r.v.'s over [0,1] which is also a J-copula ,i.e., all joint marginal distributions depend in form only on the number of distinct arguments. In turn, it follows that for any collection $\underset{\sim}{A} \overset{d}{=} (A_j)_{j \in J}$ of fuzzy subsets A_j of base space X_j , $j \in J$,

$$S_{\underset{\sim}{U}}(\underset{\sim}{A}) \overset{d}{=} (S_{U_j}(A_j))_{j \in J} \qquad (4)$$

is a well defined random subset (of appropriate X_j's) process.

Theorem 1.

Let $\underset{\sim}{U}$ be arbitrary as above and define the fuzzy set operator $\psi_\&$ by , for any $\underset{\sim}{u} \overset{d}{=} (u_j)_{j \in J}$, $u_j \in [0,1]$, $j \in J$,

$$\psi_\&(\underset{\sim}{u}) \overset{d}{=} \Pr(\underset{j \in J}{\&}(U_j \leq u_j)) , \qquad (5)$$

noting that $\psi_\&$ will be well defined and the same as when defined recursively. Let ψ_{or} be the DeMorgan transform of $\psi_\&$,i.e.,

$$\psi_{or}(u,v) = 1- \psi_\&(1-u,1-v) , \qquad (6)$$

for all $u,v \in [0,1]$. Then let F denote any corresponding fuzzy set system formed from these definitions for $\psi_\&$ and ψ_{or} . Then:

System F is weak homomorphic, separately for all three operators, to the natural corresponding random set system through $\underset{\sim}{S}_{\underset{\sim}{U}}$.

Thus, for example, for fuzzy set intersection defined through $\psi_\&$,

$$\circledcap \underset{\sim}{A} \approx S_U(\circledcap \underset{\sim}{A}) \approx \cap S_{\underset{\sim}{U}}(\underset{\sim}{A}) \quad , \qquad (7)$$

where $\circledcap$ denotes fuzzy intersection and $\underset{\sim}{A}$ is an arbitrary collection of fuzzy subsets of X. The equivalence relation $\approx$ is defined by the one point coverage probabilities, in the case of random sets, and membership

---

[1] Manes(1982) has proposed a unifying theory of uncertainty modeling which contains as special cases fuzzy set theory with $\psi_\&$=min and ψ_{or}=max , probability theory, and topological neighborhood theory.

values, in the case of fuzzy sets. Thus eq.(7) is the same as

$$\phi_{\textcircled{\cap}\underset{\sim}{A}}(x) = \Pr(x \in S_U(\textcircled{\cap}\underset{\sim}{A})) = \Pr(x \in \cap S_{\underset{\sim}{U}}(\underset{\sim}{A})),$$

for all $x \in X$. (7')

□

Remarks

(i) S_U has the property that for any base space X and any fuzzy subset A of X,

$$A \approx S_U(A) \quad . \qquad (8)$$

Such a mapping is called a canonical choice function. $S_{\underset{\sim}{U}}$ is called a choice function family induced by S_U. Note that there can be infinitely many such families induced by the same canonical choice function, as is the case here, if different joint distributions can be constructed for the random sets involved.

(ii) Another canonical choice function T can be constructed by identifying T(A) with its ordinary set membership function-which is also random-where all $\phi_{T(A)}(x)$'s are statistically independent zero-one random variables with $\Pr(\phi_{T(A)}(x) = 1) = \phi_A(x)$, all $x \in X$.

In turn, choose first any semi-distributive DeMorgan fuzzy set system F - a semi-distributive system satisfies a form of distributivity formally similar to the intersection expansion of the probability of a union of events; any such DeMorgan system; letting $\psi_n = 1-(\cdot)$, has for its last two components, (min,max), (prod,probsum), or more generally any ordinal sum-a certain type of linear like combination of these two. (See Goodman, 1982a and Klement, 1981.) Then define the choice function family $\underset{\sim}{T}$ by using the technique as above for constructing T, but expanded in terms of another index involving ψ from F. This family yields weak homomorphic relations for $\psi_{\&}$.

(iii) For the special cases for $\underset{\sim}{U}$, as above, if $U_j = U$, for all $j \in J$, or all U_j's are statistically independent, and similarly, if $\psi_{\&}$ = prod in the construction of $\underset{\sim}{T}$, then both $S_{\underset{\sim}{U}}$ and $\underset{\sim}{T}$ yield not only for the corresponding system F to have weak homomorphic random counterparts, but also a wide variety of other homomorphic-like relations.

(iv) Other choice function families may be constructed yielding for semidistributive systems weak homomorphic relations for arbitrary combinations of $\psi_{\&}$ and ψ_{or}, as well as for fuzzy arithmetic operations.

Theorem 2.
Given any ordinary n-ary operator over a collection of power classes of base spaces and any choice function family, there exists a unique n-ary fuzzy set operator which is weak homomorphic to the ordinary one over the random sets induced through the choice function family. The latter operator is an extension of the former. All results can be explicitly constructed.

□

Thus, the conclusions from Theorems 1 and 2 emphasize that fuzzy sets may be identified with classes of random sets equivalent under the one point coverage functions to the former. These random sets may differ considerably, according to the choice function employed generating them, such as the nested S_U type and the very broken-up T type. Many fuzzy set operators correspond weak homomorphically to natural corresponding ordinary random set operators, which are also not uniquely determined as is the case for random sets relative to equivalent fuzzy sets, although by specifying both random set operator and choice function family, the weak homomorphic fuzzy set is uniquely determined and is an extension of the former.

BACKGROUND: UNIFORMLY MOST ACCURATE POSSIBILISTIC ESTIMATORS

This section corresponds to justification (b) given in the introduction for the basic correlation algorithm. All results in more expanded form may be found in section 10 of Goodman(1983) with some earlier work presented in Goodman, 1982a).

Theorem 3
For any collection of fuzzy subsets $\underset{\sim}{C}$ of X and any nondecreasing function $g: [0,1]^n \to [0,1]$ i.e., if $V_1 \leq V_2$ componentwise, $g(V_1) \leq g(V_2)$, and collection of confidence levels $\underset{\sim}{a}=(a_j)_{j\in J}$ with $a_j \in [0,1]$, $j \in J$, the conjunctive hypothesis $\cap S_{\underset{\sim}{a}}(\underset{\sim}{C})$ (using the previous notation for S_U with U replaced by nonrandom values) concerning unknown parameter θ has a unique uniformly most accurate $g(\underset{\sim}{a})$-level outer estimator of the form $S_{g(\underset{\sim}{a})}(C)$ for any fuzzy subset C of X. Accuracy here is with respect to full subset inclusion. The optimal estimator is obtained by choosing C_0 such that

$$\phi_{C_0}(x) \overset{d}{=} g(\phi_{C_1}(x), .., \phi_{C_n}(x)), \quad (9)$$

for all $x \in X$.

□

Remarks.

(i) A dual result to Theorem 3 holds for disjunctive hypothesis $\cup S_{\underset{\sim}{a}}(\underset{\sim}{C})$ and inner fuzzy set estimators.

(ii) A natural extension of the concept of uniform most accurate can be made with respect to any hypothesis concerning an unknown parameter θ which consists of an arbitrary mixture of confidence sets $\{x \mid \phi_{C_j}(x) \geq a_j\}$ containing θ with respect to the ordinary logical operations & and or. In addition, it should be noted that negatives on such confidence sets can be converted essentially back to the same form:

$$\text{not}(\theta \in \{x \mid \phi_{C_j}(x) \geq a_j\}) = (\theta \in \{x \mid \phi_{\textcircled{n}C_j}(x) \geq 1-a_j\}) \quad (10)$$

where fuzzy complement $\copyright$ is evaluated via $1-(\cdot)$. Modifications with respect to the projection out of nuisance parameters - i.e., the possible components of θ which are not to be finally described - may be carried out relative to the corresponding components of the optimal estimating fuzzy set.

Combining the results of Theorems 1-3 and the above remarks leads to the following important procedure :

Procedure

Given a collection of confidence statements $\underset{\sim}{C}$ about an unknown parameter θ with some statements C_j representing random sets and others representing fuzzy sets , convert the random set statements to their corresponding fuzzy set forms resulting from their one point coverage probabilities and then apply Theorem 3 or any of its extensions discussed above. Alternatively, $\underset{\sim}{C}$, by appropriate choice functions can be converted to pure random set forms. In either situation obviously a change in information content occurs.(An open research issue involves the measurement of this change.)

BACKGROUND:FUZZY SET FORM OF BAYES' THEOREM

Theorem 4 (Related to Goodman,1982 a,b.) For any fuzzy subset C of $X_1 \times X_2$ and system F, the X_1 projection $C_{(1)}$ of C into X_1 is

$$\Phi_{C_{(1)}}(x_1) \overset{d}{=} \underset{x_2 \in X_2}{\Psi_{or}}(\Phi_C(x_1,x_2)), \quad (11)$$

for any $x_1 \in X_1$. Similarly, for $C_{(2)}$ w.r.t. X_2.

For any $x_j \in X_j$, j=1,2, there exist fuzzy subset $C_{(1|x_2)}$ of X_1 and fuzzy subset $C_{(2|x_1)}$ of X_2 such that

$$\Phi_C(x_1,x_2) = \Psi_\&(\Phi_{C_{(1|x_2)}}(x_1), \Phi_{C_{(2)}}(x_2))$$
$$= \Psi_\&(\Phi_{C_{(2|x_1)}}(x_2), \Phi_{C_{(1)}}(x_1)). \quad (12)$$

If $\Psi_\&$ is monotone increasing in all of its arguments, then the conditional fuzzy sets $C_{(1|x_2)}$ and $C_{(2|x_1)}$ are uniquely determined. ◻

Theorem 5 Fuzzy Bayes' Theorem (Goodman,1982) Suppose that a fuzzy subset B of X_1 is given, calling B the prior set, and for each $x_1 \in X_1$, there is a fuzzy subset C_{x_1} of X_2 indexed by $x_1 \in X_1$, called the conditional data (on parameter) set. Then there is a fuzzy subset D of $X_1 \times X_2$, such that $D_{(1)} = B$, $D_{(2|x_1)} = C_{x_1}$; for all $x_1 \in X_1$, and such that $D_{(1|x_2)}$ (and D) are determined implicitly through eq.(12) in terms of B and $C_{x_1}; x_1 \in X_1$. $D_{(1|x_2)}$ is called the posterior set (conditioned on x_2). ◻

This result has been used to develop a theory of fuzzy set sampling . See Goodman (1982 b) for properties of fuzzy posterior sets for both small and asymptotically large samples.

BASIC ALGORITHM

With all of the basic theoretical groundwork established, the correlation algorithm may now be defined. It relies in form basically on Theorem 3 .

Let two track histories i and j be selected and the correlation level between them is sought. Assume that all observed data

$$\overset{\circ}{Z} \overset{d}{=} (\overset{\circ}{Z}^{(i)}, \overset{\circ}{Z}^{(j)}) \; ; \; \overset{\circ}{Z}^{(i)} \overset{d}{=} \begin{pmatrix} \overset{\circ}{Z}{}_1^{(i)} \\ \vdots \\ \overset{\circ}{Z}{}_m^{(i)} \end{pmatrix}, \; \overset{\circ}{Z}^{(j)} \overset{d}{=} \begin{pmatrix} \overset{\circ}{Z}{}_1^{(j)} \\ \vdots \\ \overset{\circ}{Z}{}_m^{(j)} \end{pmatrix} \quad (13)$$

where $A_1,..,A_m$ are m fixed preselected attributes and each $\overset{\circ}{Z}{}_k^{(i)}$ is an "observed" data vector (suitably updated) or outcome lying in dom(A_k) , k=1,..,m. Similar remarks hold for each $\overset{\circ}{Z}{}_k^{(j)}$ relative to A_k for track history j, for k=1,..,m.

Employ similar notation for any possible value that Z could have been, by dropping the super degree notation, everywhere.

Modify slightly the subscript notation for the C_k's making up the hypothesis collection $\underset{\sim}{C}$ of fuzzy sets used to describe the unknown parameter which here is

$$\theta = \begin{pmatrix} Z \\ -- \\ Q \end{pmatrix} , \quad (14)$$

where $Q \in [0,1]$ is the unknown level of correlation . Then let

$$\Phi_{C_{k,i}}(Z_k^{(i)}) \overset{d}{=} M_k(\overset{\circ}{Z}{}_k^{(i)}, Z_k^{(i)}) , \quad (15)$$

where $Z_k^{(i)} \in$ dom(A_k) is arbitrary, and where M_k is formally a membership function or possibility function corresponding to a fuzzy subset of dom(A_k) $\times$ dom(A_k), k=1,..,m.

Each such M_k is obtained previously and may be interpreted in two ways :

$M_k(x,y)$ = matching level between x and y in dom(A_k)

= possibility that y is the true value which gave rise to observation x .

M_k can be obtained either empirically by interrogating an established panel of experts if the attribute appears conducive to this such as for example classification or shape ; or analytically, as for example geolocation or

frequency, where for example classical hypotheses testing statistics may be developed in the first case and simple transformation of probability technique applied to the Doppler shift equation for the second case.
The same M_k 's are also used similarly for obtaining $\Phi_{C_{k,j}}(Z_k(j))$, for k=1,..,m.

Note that the data $\mathring{Z}$ is fixed ; the Z potentially a variable.

Next, let

$$\Phi_{C_{m+k}}(Z,Q) \overset{d}{=} R_k(Z,Q) , \qquad (16)$$

for any Z and Q as above. Each R_k is formally the membership function of a fuzzy subset of $\mathrm{dom}(A_{t_{k,1}}) \times \cdots \times \mathrm{dom}(A_{t_{k,n_k}}) \times [0,1]$

where $1 \le t_{k,1} < \cdots < t_{k,n_k} \le m$ represent

indices of a group of attributes used in the antecedent for rule k, with Q appearing in the consequent , for k=1,..,r. More specifically, R_k corresponds to:

> "If a match between i and j occurs relative to attribute $A_{t_{k,v}}$ to degree (or intensity,etc.) $\alpha_{t_{k,v}}$, for v=1, ...,n_k , then correlation occurs between i and j to degree $h_k(\underset{\sim}{\alpha}_k)$",

where

$$\underset{\sim}{\alpha}_k \overset{d}{=} (\alpha_{t_{k,1}}, \ldots, \alpha_{t_{k,n_k}}) \qquad (17)$$

and $h_k(\underset{\sim}{\alpha}_k)$ may be first obtained linguistically and then converted to some convenient scale such as the unit interval. The group of attributes, $\underset{\sim}{\alpha}_k$ and $h_k(\underset{\sim}{\alpha}_k)$ in general are obtainable from a panel of experts.

Within a possibilistic setting , it follows that R_k may be evaluated as

$$R_k(Z,Q) = \psi_{\Rightarrow}(G_k(Z), Q^{((h(\underset{\sim}{\alpha}_k)))}) , \qquad (18)$$

$$G_k(Z) = \underset{v=1,..,n_k}{\psi_{\&}} \left(\left(M_v(Z_v(i), Z_v(j)) \right)^{((\alpha_{t_{k,v}}))} \right) , \qquad (19)$$

where

$$\psi_{\Rightarrow}(x,y) \overset{d}{=} \psi_{or}(1-x,y) , \qquad (20)$$

for all $x,y \in [0,1]$, and where the superscripts $((\cdot))$ are powers evaluated thru a conveniently chosen transform from the unit interval to the positive real line. For example , one can reasonably choose

$$((x)) \overset{d}{=} x/(1-x) , \qquad (21)$$

for all $x \in [0,1]$. In this relationship, x=0.5 could correspond to "match" , with thus ((0.5)) = 1 , resulting in no change to intensification, whereas x=0.67 could correspond to "at least,a high match", with thus ((0.67))=2.0, an intensification,etc.

Thus, the conjunctive hypothesis concerning unknown parameter θ in Theorem 3 becomes

$$\bigcap S_{\underset{\sim}{a}}(\underset{\sim}{C}) = \bigcap_{k=1}^{m} S_{a_{k,i}}(C_{k,i}) \cap \bigcap_{k=1}^{m} S_{a_{k,j}}(C_{k,j}) \cap \bigcap_{k=1}^{r} S_{a_{m+k}}(C_{m+k}) . \qquad (22)$$

Choose now in Theorem 3, $g = \psi_{\&}$. Then

eq.(9) yields the uniformly most accurate $\psi_{\&}(\underset{\sim}{a})$-level outer estimator for parameter θ as given in eq.(14) , for any given set of confidence levels $\underset{\sim}{a}$, with Z arbitrary fixed nuisance value in $\underset{k=1}{\overset{m}{\times}} \mathrm{dom}(A_k)$; $\mathring{Z}$ completely fixed:

$$\Phi_{C_o}\left(\binom{Q}{Z}; \mathring{Z}\right)$$

$$= \underset{\substack{k=1,..,m ,\\ w=1,..,r}}{\psi_{\&}} \left(M_k(\mathring{Z}_k(i), Z_k(i)), M_k(\mathring{Z}_k(j), Z_k(j)), R_w(Z,Q) \right)$$

$$= \psi_{\&}\left(M(\mathring{Z},Z), R(Z,Q) \right) , \qquad (23)$$

where

$$M(\mathring{Z},Z) \overset{d}{=} \underset{k=1,..,m}{\psi_{\&}} \left(M_k(\mathring{Z}_k(i), Z_k(i)), M_k(\mathring{Z}_k(j), Z_k(j)) \right)$$

= overall matching table error effect, (24)

$$R(Z,Q) \overset{d}{=} \underset{w=1,..,r}{\psi_{\&}}(R_w(Z,Q))$$

= overall inference rule effect . (25)

Next, compute, by projecting out the nuisance value Z, for any possible Z, the optimal estimator describing Q :

$$\Phi_{B_o}(Q;\mathring{Z}) = \Phi_{(C_o)_{(1)}}(Q;\mathring{Z}) , \qquad (26)$$

where $(C_o)_{(1)}$ denotes the projection of C_o into [0,1] (and the projecting out of all $Z \in \underset{k=1}{\overset{m}{\times}} \mathrm{dom}(A_k)$, see eq.(11)).

It can be shown that with the proper identifications, optimal estimating set B_o is also a posterior fuzzy set:

Theorem 6.
Recall and define for the key variables Q,Z,$\mathring{Z}$:

$$Q \in X_1 \overset{d}{=} [0,1]$$

$$Z \in X_2 \overset{d}{=} \underset{k=1}{\overset{m}{\times}} \mathrm{dom}(A_k)$$

$$\mathring{Z} \in X_3 \overset{d}{=} X_2 .$$

Let L be that fuzzy subset of $X_1 \times X_2 \times X_3$

jointly describing the above three variables and let the natural correspondences yield

$$\begin{cases} R(Z,Q) = \Phi_{L(1\mid Z)}(Q) , \\ M(\mathring{Z},Z) = \Phi_{L(2\mid \mathring{Z})}(Z) . \end{cases} \qquad (27)$$

Suppose, finally, that Z is sufficient for Q relative to observation $\mathring{Z}$:

$$\Phi_{L((1\mid \mathring{Z})\mid(2\mid \mathring{Z}))}(Q) = \Phi_{L(1\mid Z)}(Q) , \qquad (28)$$

for all possible $Q,Z,\mathring{Z}$.

Then it follows, using eqs.(23),(26) and Theorem 4 ,

$$\begin{aligned} \Phi_{B_o}(Q;\mathring{Z}) &= \Psi_{\substack{or \\ Z\in X_2}} \left(\Phi_{C_o}\left(\left(\tfrac{Q}{Z}\right) ;\mathring{Z}\right)\right) \\ &= \Psi_{\substack{or \\ Z\in X_2}} \left(\Psi_{\&}\left(\Phi_{L(2\mid \mathring{Z})}(Z),\Phi_{L(1\mid Z)}(Q)\right)\right) \\ &= \Phi_{L(1\mid \mathring{Z})}(Q) \\ &= \text{posterior possibility function of } Q \text{ given } \mathring{Z} . \end{aligned} \qquad (29)$$

$\square$

A summary of the basic computations involved in the correlation algorithm is given below in Fig. 1.

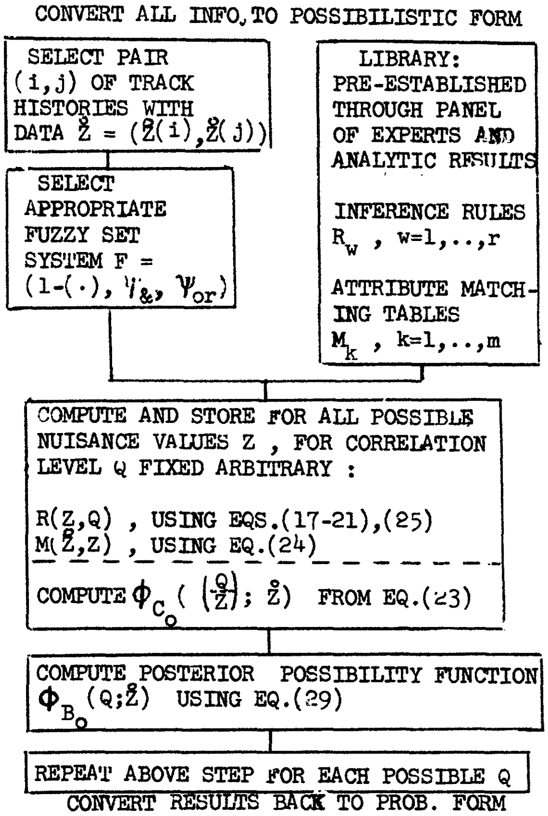


Fig. 1. Outline of the basic correlation algorithm.

CONCLUSION

A preliminary form of the algorithm is being developed . In conjunction with this, a panel of experts has been assembled and used in selecting the pertinant attributes for some specific problems of interest, determining attribute matching tables, and formulating some inference rules. In addition, analytic procedures have proven useful for modeling some of the matching tables. Complete flow charts for the general algorithm have been completed, embellishing upon Fig. 1. Because running times appear rather lengthy, an auxilary suboptimal, but simpler algorithm has been developed which produces as an output a function which pointwise is a lower bound to the posterior correlation possibilities. In summary, a procedure has been developed which treats both nongeolocation and geolocation attribute information in the target data association problem, from the viewpoint of possibility theory and its relations to probability theory. In the future, the problem of determining the most significant inference rules from the myriad possible choices will be addressed. Potential tie-ins with Artificial Intelligence techniques are likely.

REFERENCES

Bowman, C.L. (1981). An architecture for fusion of multisensor ocean surveillance data. Proc.20 IEEE Conf. Decis. & Cntrl., 1419-1420.

Bowman, C.L. and C.L. Morefield (1980). Multisensor fusion of target attributes and kinematics. Proc. 19 IEEE Conf. Decis. & Cntrl.,837-839.

Dockery, J.T. (1982). Fuzzy design of military information systems. Int. J. Man Mach. Stud.,16,1-38.

Dubois,D. and H. Prade (1980). Fuzzy Sets and Systems. Academic Press, New York.

Goodman,I.R. (1981). Applications of a combined probabilistic and fuzzy set technique to the attribute problem in ocean surveillance. Proc. 20 IEEE Conf. Decis. & Cntrl., 1409-1411

Goodman,I.R. (1982 a). Some fuzzy set operations which induce homomorphic random setoperations. Proc. 26 Conf. Gen. Sys. Res. (AAAS), 417-426. To be expanded, 1983.

Goodman, I.R. (1982 b). Some asymptotic properties of fuzzy set systems. Proc. 2 World Conf. Math.Serv. Man. Can. Is.

Goodman, I.R., H.L. Wiener, and W.W. Willman (1980). Naval Ocean-Surveillance Correlation Handbook,1979, NRL Rpt. 8402.

Klement, E.P. (1981). Operations on fuzzy sets and fuzzy numbers related to triangular norms. Proc. 11 Int.Symp.Mul Log,218-225.

Manes,E.G. (1982). A class of fuzzy theories. J. Math. Anal. & Applic., 85, 409-451.

Watson,S.R.,J.J Weiss, and M.J. Donnell (1979) Fuzzy decision analysis. IEEE Trans.Sys. Man & Cyber., vol. SMC-9,no.1, 1-9.

Wiener,H L., W.W. Willman,I.R. Goodman,J.H. Kullback (1979).Naval Ocean-Surveillance Correlation Handbook,1978,NRL Rpt.8430.

MYOPIC AND PRESBYOPIC APPROACHES TO A MULTI-SENSOR, MULTI-TARGET RESOURCE ALLOCATION PROBLEM

Harilaos N. Psaraftis

Anastassios N. Perakis

Department of Ocean Engineering, Massachusetts Institute of Technology,
Cambridge, Massachusetts 02139

Abstract. This paper takes advantage of recent results on the probabilistic modeling of the ocean acoustic detection process to develop two approximate procedures for tackling a simplified version of the m-sensor, n-target resource allocation problem. The first procedure is termed "myopic" (short-sighted) and applies if we are interested to maximize the expected number of targets held in the short-run. This second procedure is termed "presbyopic" (far-sighted) and applies if we are interested to obtain the maximum expected number of targets held in the long-run, that is, when the system is in the steady state. Both approaches are suboptimal because they neglect, each in a different way, the interdependence of allocation decisions through time.

We formulate the myopic case as a Linear Programming "Assignment" optimization problem whose inputs are dynamically updated through time. Then we do the same for the presbyopic case. It is seen that if a presbyopic policy is followed, no switching decisions will ever occur. We then extend the myopic formulation to incorporate a general target-holding reward function as well as switching costs. We present some illustrative examples, applying both approaches to simulated data. We also present a comparison of both methods with an exact Stochastic Dynamic Programming approach we had developed earlier for problems of very small size.

INTRODUCTION: THE PROBLEM

Consider the following oversimplified scenario in passive underwater acoustic surveillance: We are faced with the problem of tracking n stationary and independent targets with m sensors (with m usually, but not necessarily, smaller than n) so as to maximize a certain measure of performance, which will be shortly defined. To induce realism, we can think of each sensor as a passive sonar array, and for some exogeneous reason that need not concern us here, we assume that each one of the m sensors, can be allocated only to one of the n targets at any given point in time. However, we are free to change the allocation through time if we so decide. Suppose at time T sensor i is allocated to target j. If at that point in time the root mean square pressure ρ_{ij} of the signal received from target j at sensor i is above a specified detection threshold ρ^0_{ij}, then we say that the target in question is "held" by that sensor. Holding a target is generally assumed to be a desirable outcome. However, due to the randomness of the ocean acoustic fluctuation process, holding will not always occur. In that respect, there will be periods of time when ρ_{ij} will be above ρ^0_{ij} and periods when it will be below it. At those times when the signal received from target j at sensor i is below its threshold, it may make little or no sense keeping that sensor allocated to that target, especially if there is a certain chance of being able to hold some other target instead (or hold the same target with another sensor). The problem we will be trying to analyze and solve in this paper can be roughly phrased as follows: If we have some probabilistic information about when each signal is likely to be above or below its detection threshold, is there an "optimal" allocation (or switching) schedule to the m sensors among the n targets through time?

The above description is for the moment only sketchy. The reader familiar with passive underwater surveillance literature should have recognized that the problem belongs to the general category of "resource allocation" problems, which is of particular importance in that area in general, as well as in target tracking in particular. Resource allocation generally calls for simultaneously tracking a number of targets with a limited number of sensors already deployed in a geographical area of interest. Limitations may be due to a number of constraints, such as number of available sensors, number of available communication channels, information processing capacity, bandwidth, etc. There is an abundance of literature in this area, with each paper tackling the problem from a particular viewpoint. See for instance Alspach [1], Bar-Shalom and others [2, 3, 4, 5, 6], Fortmann and others [7, 8, 9, 10], Friedlander [11], Keverian and Sandell [12], [13] and Tenney [14].

This paper does not attempt to get involved with the intricate signal processing and tracking issues associated with the real world environment. In fact, some of the assumptions made may be quite restrictive or unrealistic. For instance, targets are assumed stationary, signals between any target-sensor pair are assumed independent and each sensor is assumed to be able to "listen" to only one target at a time. The scope of the paper is to take advantage of recent results in the modeling of the ocean acoustic detection process (Psaraftis and others [15, 16]). These results have shed some light into the timing of detection events in the ocean for the case of a phase-random multipath acoustic process (described in detail in Hamblen [17], Mikhalevsky and Dyer [18] and Mikhalevsky [19, 20]). In [15] it has been shown that under the phase-randomness assumptions the ocean acoustic detection process has quantifiable memory characteristics. Probability density functions of the time between two successive detection events and the time a target is held were derived. In [16] equivalent discrete time Markov models were developed. This paper will be using those Markov models, which will be reviewed for that purpose in the next section.

The first step toward using the above detection models in a resource allocation setting was made in Psaraftis and Perakis [21]. In that paper, an exact Stochastic Dynamic Programming algorithm was developed to solve the (m=1, n=2) case of the problem described above under a finite horizon of observations. A generalized user-supplied penalty/reward function and optional delays in switching from one target to other were examined. Despite the valuable insights obtained from the application of that model to small problem instances, it was also observed that any attempt to extend such an exact approach to the general m-sensor, n-target case, although straightforward from a formulation point of view, is doomed to encounter severe computational and storage difficulties for any but very small values of m and n (e.g. m>1, n>3). In that respect, it was argued that the focus for tackling the general m-sensor n-target problem should be placed on more tractable procedures that are of course suboptimal (heuristic).

This paper develops two such approximate procedures, one termed "myopic" and the other "presbyopic".* In the myopic approach we are only interested to maximize the expected number of targets held in the short-run. In the presbyopic approach we are interested to obtain the maximum expected number of targets held in the long-run, that is, when the system is in the steady state. Both approaches are suboptimal because they neglect, each in a different way, the interdependence of allocation decisions through time.

The rest of the paper is organized as follows: Section 2 formulates the myopic problem and presents a solution algorithm. Section 3 does the same for the presbyopic case. Section 4 extends the myopic formulation to incorporate a general reward function as well as switching costs. Section 5 presents some illustrative examples, applying both approaches to simulated data. It also presents a comparison of both methods with the exact approach, for problems of very small size (m=1, n=2). Finally, Section 6 summarizes the results of the paper and suggests directions for further work.

2. THE MYOPIC APPROACH

As mentioned before, previous work (Psaraftis and others [15, 16]) has shown that for a phase-random multipath acoustic process, fluctuations from each target to each sensor can be modeled as a discrete-time, two-state Markov process. Specifically, and referring to Fig. 1, we will assume that if the signal from target j to sensor i is above a specified threshold then the state of the signal in question is U_{ij} (for "up"), otherwise the state of the signal is D_{ij} (for "down"). The Markov transition probabilities a_{ij}, b_{ij} are assumed to be known for each sensor-target pair (i=1,...m; j=1,...n). Opportunities for a state transition occur simultaneously for each sensor-target pair every ΔT units of time. The reader is referred to [16] for details on how the transition probabilities can be calculated from the detection threshold and other signal parameters, as well as on how ΔT can be calibrated. For the purposes of this paper the above parameters will be assumed known and constant through time. We will also assume that all sensor-target pairs define mutually independent Markov processes. In other words, the fact that the signal from target i to sensor j might be "up" does not influence the state of the signal from the same target to another sensor, however close to j that other sensor might be. It should be

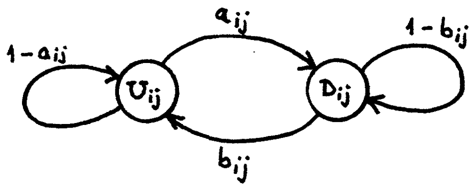


Fig. 1

---

* Myopic = short-sighted; presbyopic = far-sighted, the opposite of myopic.

mentioned that in realistic situations, spatial correlation (or coherence) between sensors may exist, and hence the above assumption may not hold in certain cases.

Each sensor i can be allocated only to one of the n targets at any point in time, say to target j. At that point in time the state of the signal from that target is known. (U_{ij} or D_{ij}). Of course during that time, the states of the signals from all other targets ($j' \neq j$) towards the same sensor (i) are not known with certainty. We assume that each sensor has the opportunity to switch targets every ΔT units of time, and that all switchings (if any) occur at the same time with potential Markov state transitions. In addition, we do not allow any target to be tracked by more than one sensor at a time.

It is clear that allocation decisions are coupled in time, for the allocation at stage k determines the initial conditions of the problem defined at stage k+1, and so on. This interdependence of decisions makes the exact solution of the problem (if we want to optimize over several time stages) especially difficult, as noted in [21].

In the <u>myopic</u> approach we investigate one possible way to <u>decouple</u> the decision-making process. In that respect, we will be interested in achieving a maximum expected "reward" only for the next stage of the process (i.e. after ΔT units of time). The short-sighted of this approach is obvious. The allocation that gives the maximum expected reward for the next stage might define a "bad" initial condition for stages to follow. The myopic approach neglects to take into account such implications. More explicitly, the myopic problem is defined as follows:

"Given a mxn matrix of a priori probabilities $[p_k(i,j)]$ with $p_k(i,j)$ the probability that at time stage k the signal from target j to sensor i is in the U_{ij} state, and the Markov transition probabilities $[a_{ij}]$ and $[b_{ij}]$ for each sensor-target pair, what is the mxn sensor-to-target allocation $[x_{k+1}(i,j)]$which maximizes the expected number of targets held at time stage k+1 (i.e. after ΔT units of time)?"

In the above problem definition $x_{k+1}(i,j) = 1$ if sensor i is allocated to target j at time stage k+1 and $x_{k+1}(i,j) = 0$ otherwise. The a priori probabilities $p_k(i,j)$ may be based either on the history of the system, (i.e. updated from past similar probabilities), or, be equal to 1 or 0 depending on whether the current allocation of sensor i to target j is successful or not. Details on this point will be presented later on.

To formulate the above myopic problem we note that the expected number of targets held at stage k+1 is equal to

$$\sum_{j=1}^{n} \sum_{i=1}^{m} x_{k+1}(i,j) \cdot p_{k+1}(i,j) \qquad (1)$$

The next-stage probabilities $p_{k+1}(i,j)$ can be obtained from $p_k(i,j)$ and the Markov transition probabilities a_{ij} and b_{ij} by the following formula:

$$p_{k+1}(i,j)=p_k(i,j)(1-a_{ij})+(1-p_k(i,j))b_{ij} \qquad (2)$$

Since each sensor can be allocated only to one target (at most) and since each target cannot be tracked by more than one sensor, it follows that

$$\sum_{i=1}^{m} x_{k+1}(i,j) \leq 1 \quad j=1,..n \qquad (3)$$

$$\text{and} \quad \sum_{j=1}^{n} x_{k+1}(i,j) \leq 1 \quad i=1,..m \qquad (4)$$

$$\text{Finally} \quad x_{k+1}(i,j) \geq 0 \quad \begin{matrix} i=1,...m \\ j=1,...n \end{matrix} \qquad (5)$$

The maximization of (1) subject to constraints (3), (4) and (5) <u>plus</u> the integrality constraints $x_{k+1}(i,j) = 0$ or 1 constitutes a zero-one integer programming problem. However, since the constraint matrix is unimodular, the integrality constraints are superfluous and hence can be dropped. The linear programming problem is nothing more than a classical <u>Assignment Problem</u>, solvable quite efficiently by a variety of methods, one of the most efficient of which is the "Hungarian method".

After solving the assignment problem we determine the next stage allocation $x_{k+1}(i,j)$. For those sensor-target pairs for which $x_{k+1}(i,j) = 1$ we can actually verify whether we hold the target or not. If we do hold the target, we <u>reset</u> $p_{k+1}(i,j) = 1$, otherwise we reset $p_{k+1}(i,j) = 0$. We then move to the following stage (set k=k+1) with the new probabilities determined from (2), and until the end N of the procedure.

The formal myopic algorithm is summarized as follows:

```
STEP 0: Initialize
        k=0
        Enter $p_k(i,j)$  i=1,..m; j=1,...n
```

STEP 1: Compute next stage probabilities from:

$$p_{k+1}(i,j) = p_k(i,j)(1-a_{ij}) + (1-p_k(i,j))b_{ij}$$

$$i=1,\ldots m;\ j=1,\ldots n$$

STEP 2: Determine next stage allocation $[x_{k+1}]$ by solving Assignment Problem AP_{k+1}:

$$\text{Max} \sum_{j=1}^{n} \sum_{i=1}^{m} x_{k+1}(i,j)\, p_{k+1}(i,j)$$

$$\text{subject to: } \sum_{j=1}^{m} x_{k+1}(i,j) \leq 1 \quad j=1,\ldots n$$

$$\sum_{j=1}^{n} x_{k+1}(i,j) \leq 1 \quad i=1,\ldots m$$

$$x_{k+1}(i,j) \geq 0 \quad i=1,\ldots m;\ j=1,\ldots n$$

STEP 3: Move to next stage and check termination.

$k = k+1$

if $k > N$ END; Else CONTINUE

STEP 4: Observe state of targets tracked: Observe or simulate those (i,j) for which $x_k(i,j)=1$, then reset $p_k(i,j) = 1$ if target j is held by sensor i, 0 otherwise.

Go to Step 1

3. THE PRESBYOPIC APPROACH

The presbyopic approach goes one step further in decoupling the allocation decision-making process. Not only does it not take into account the next-stage initial conditions into current allocation decisions, but it also neglects their effect into future decisions as well. To be more precise, the presbyopic approach assumes the process is in the steady state, that is, sees only far into the future. It then attempts to maximize the expected number of targets held in the long run, that is, based on the corresponding Markov steady state probabilities.

It is not difficult to see that it the presbyopic approach is followed, no switching decisions will every occur. This is so because the problem input, that is, the probabilities that sensor-target pairs will be "up" cannot be affected by allocation decisions. All we have to do to determine the steady-state allocation is to solve the following assignment problem once and for all:

$$\text{Maximize} \sum_{i=1}^{m} \sum_{j=1}^{n} x_\infty(i,j)\, p_\infty(i,j)$$

$$\text{subject to } \sum_{j=1}^{n} x_\infty(i,j) \leq 1, \quad i=1,\ldots m$$

$$\sum_{i=1}^{n} x_\infty(i,j) \leq 1, \quad i=1,\ldots n$$

$$x_\infty(i,j) \geq 0, \quad i=1,\ldots m;\ j=1,\ldots n$$

In the above problem, $x_\infty(i,j)$ is the steady-state allocation and $p_\infty(i,j)$ is the corresponding steady-state Markov probability, given by

$$p_\infty(i,j) = b_{ij}/(a_{ij}+b_{ij}) \qquad (6)$$

The presbyopic algorithm is therefore formalized as follows:

Presbyopic Algorithm

STEP 0: Compute steady state probabilities

$$p_\infty(i,j) = b_{ij}/(a_{ij}+b_{ij}) \quad i=1,\ldots m;\ j=1,\ldots n.$$

STEP 1: Determine steady state allocation $[x_\infty]$ by solving Assignment Problem AP_∞:

$$\text{Max} \sum_{j=1}^{n} \sum_{i=1}^{m} x_\infty(i,j)\, p_\infty(i,j)$$

$$\text{subject to: } \sum_{i=1}^{m} x_\infty(i,j) \leq 1 \quad j=1,\ldots n$$

$$\sum_{j=1}^{n} x_\infty(i,j) \leq 1 \quad i=1,\ldots m$$

$$x_\infty(i,j) \geq 0 \quad i=1,\ldots m;\ j=1,\ldots n.$$

END

4. SOME EXTENSIONS TO THE MYOPIC APPROACH

This section discusses several extensions to the myopic approach: Consider for instance the case where holding certain targets is more desirable than holding other targets; or the case where switching the sensor between targets involves a cost. These are features that can be easily incorporated into the algorithm presented in section 2.

Let w_j be the reward for holding target $j(j=1,\ldots n)$ and s_i be the cost involved whenever sensor i switches between targets. w_j can be easily introduced in the objective function of the assignment problem, as a multiplicative factor to $p_{k+1}(i,j)$ (Step 2). However, s_i cannot be introduced so easily, for that cost should appear only if $x_k(i,j) \neq x_{k+1}(i,j)$, in other words only if $x_k(i,j) = 0$ and $x_{k+1}(i,j) = 1$ or vice versa.

It is relatively easy to check that this can be done if we add the following term to the objective function of the assignment problem:

$$-\sum_{j=1}^{n}\sum_{i=1}^{m}\delta_{ij}(x_{k+1}(i,j) - x_k(i,j))$$

$$\text{with } \delta_{ij} = \begin{cases} 0.5\ s_i & \text{if } x_k(i,j) = 0 \\ -0.5\ s_i & \text{if } x_k(i,j) = 1 \end{cases}$$

The extended myopic approach is as follows:

Myopic Algorithm Extended

STEP 0: Same as before

STEP 0.5: if k=0 $\delta_{ij} = 0$

$$\text{Otherwise } \delta_{ij} = \begin{cases} 0.5\ s_i & \text{if } x_k(i,j) = 0 \\ -0.5\ s_i & \text{if } x_k(i,j) = 1 \end{cases}$$

$(i=1,\ldots m;\ j=1,\ldots n)$

STEP 1: Same as before

STEP 2: Determine next stage allocation $[x_{k+1}]$ by solving Assignment Problem AP_{k+1}:

$$\text{Max} \sum_{j=1}^{n}\sum_{i=1}^{m} x_{k+1}(i,j)[p_{k+1}(i,j)w_j - \delta_{ij}]$$

$$\text{Subject to} \sum_{i=1}^{m} x_{k+1}(i,j) \leq 1 \quad j=1,\ldots n$$

$$\sum_{j=1}^{n} x_{k+1}(i,j) \leq 1 \quad i=1,\ldots m$$

$$x_{k+1}(i,j) \geq 0 \quad i=1,\ldots m,\ j=1,\ldots n$$

STEP 3: Same as before

STEP 4: Same as before, except to go to Step 0.5.

5. NUMERICAL EXAMPLES

This section presents two simple illustrative numerical examples to test the algorithms presented in Section 2 and 3.

The first example is a comparison between the myopic algorithm and the exact Dynamic Programming algorithm presented in [21] for the (m=1, n=2) case. For this case the myopic policy is trivial: Always tune to the target that has the highest probability of being "up" at the next stage. The objective for the Dynamic Programming solution is to maximize the expected number of targets held over a specified duration of N time stages (here N=46). Fig. 2 refers to this comparison. The Markov transition probabilities for the two targets of this example are $(a_i, b_i) = (0.503, 0.312)$ and $(a_2, b_2) = (0.424, 0.366)$. The process is initialized at stage k=0 with $p_0(1,1) = 1.0$ and $p_0(1,2) = 0.463$ (which happens to be equal to the steady-state probability of the corresponding Markov process).

Fig. 2a presents the exact DP solution. $p_k(1,j)$ is plotted for $k=0,1\ldots,46$ and $j=1,2$. $p_k(1,j) = 1$ or 0 means that the sensor is allocated to target j at stage k. If $p_k(1,j) = 1$, this means that the sensor holds that target, while $p_k(1,j) = 0$ means that the sensor does not hold it. $0 < p_k(1,j) < 1$ means that the sensor is not allocated to target j at stage k. A switching can be recognized in this plot by identifying two consecutive stages, say k and k+1, for which $p_k(1,j) = 0$ or 1, and $0 < p_{k+1}(i,j) < 1$ for j=1 or 2. Fig. 2b represents a similar plot for the myopic approach. One can observe a significant similarity with Fig. 2a. It happens that in this example the myopic algorithm holds target 1 4 times and target 2 22 times, versus 7 times for target 1 and 17 times for target 2 obtained by the exact DP algorithm. Surprisingly enough, in this example the myopic algorithm holds a target for more time stages (26) than the exact DP algorithm (24). Of course this fact is not a disproof of the optimality of the DP algorithm, because one cannot compare two stochastic algorithms by their actual outcomes. However this fact suggests that the myopic algorithm, at least for this case, performs pretty well, despite being suboptimal on an expected value basis. It should be mentioned that the presbyopic allocation in this case is to allocate the sensor to target 2, since the Markov process describing the signal from that target has a higher steady-state probability (0.463) than the other target (0.386). A presbyopic allocation of that nature would have held target 2 27 times.

Another simulation example, this time involving more sensors and targets (m=3, n=8) is presented in Fig. 3. Table 1 presents the Markov transition probabilities (a_{ij}, b_{ij}), the steady-state probabilities $p_\infty(i,j)$ and the a priori (initial) probabilities $p_0(i,j)$ for each sensor-target pair (i,j). In Fig.3, sensors are represented by squares and targets by circles. A link between sensor i and target j means that sensor i is allocated to target j. Successful allocations (target is held) are shown by asterisks (*). The example assumes N=9 time stages, k=1,..9,

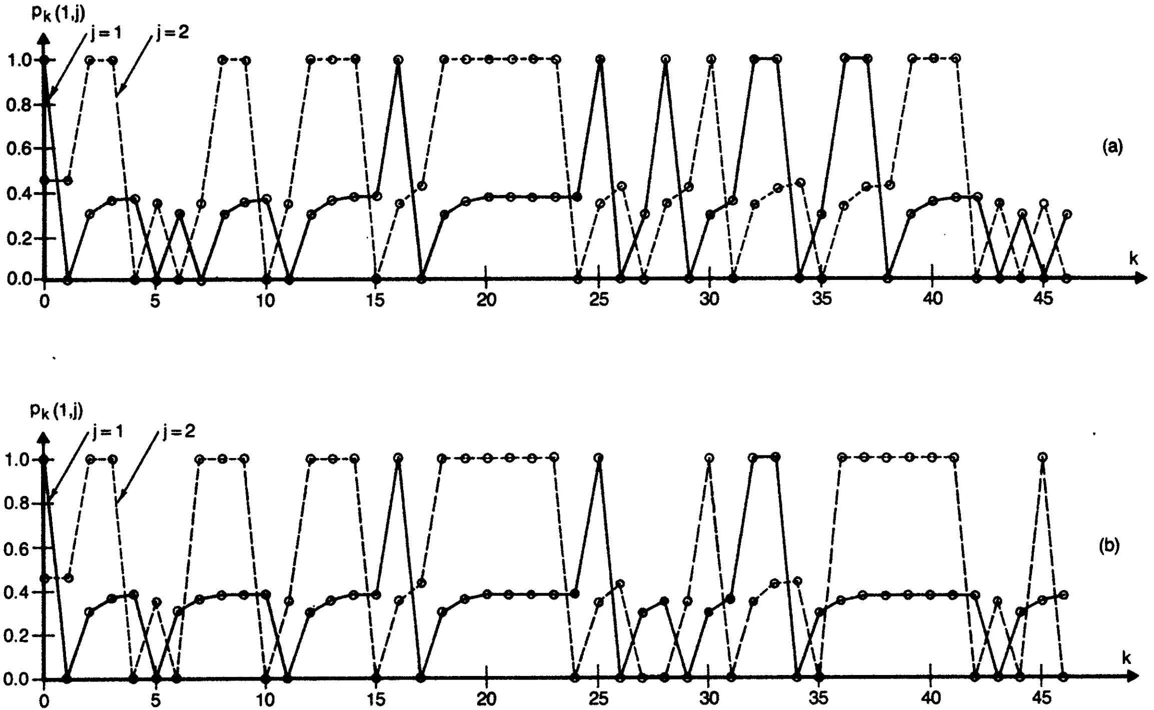

pk (1,j)
j=1
j=2
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
25
30
35
40
45
k
(a)
(b)

Fig. 2

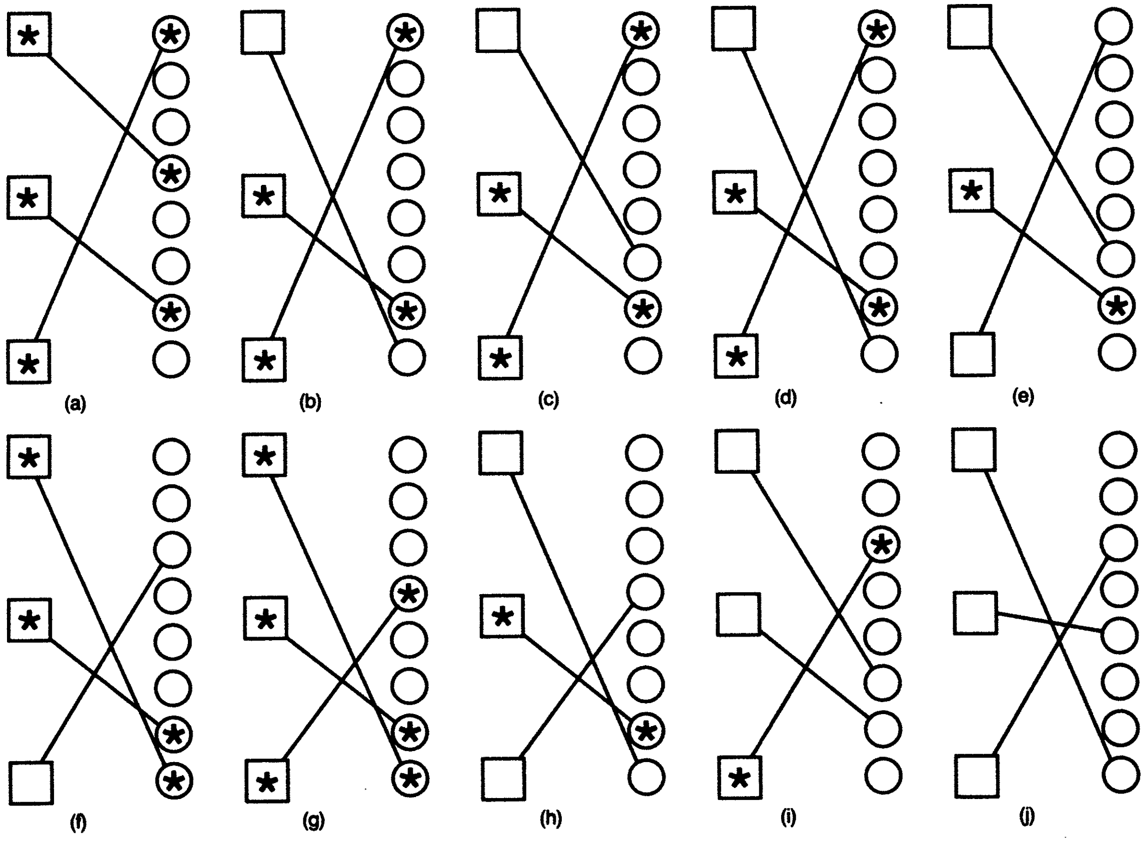

(a)
(b)
(c)
(d)
(e)
(f)
(g)
(h)
(i)
(j)

Fig. 3

with the process initialized at stage k=0. The allocation pattern shown in Figures 3a through 3i corresponds to a myopic policy. The corresponding presbyopic policy is shown in Fig. 3j.

Table 2 shows the expected (E_k) and actual (A_k) number of targets held at each time stage k when a myopic or a presbyopic policy is followed. It is straightforward to see that $E_k = \sum_{j=1}^{n} \sum_{i=1}^{m} p_t(i,j)\, x_t(i,j)$ with t=k for the myopic case and t=∞ for the presbyopic. From Table 2 we can observe that $\sum_{k=1}^{N} E_k$ is about the same for both cases but that the myopic allocation outperforms the presbyopic in terms of actual number of targets held.

TABLE 1: Inputs to the (m=3, n=8) Case

i	j	a_{ij}	b_{ij}	$p_\infty(i,j)$	$p_0(i,j)$
1	1	0.053	0.312	0.383	0.383
	2	0.473	0.331	0.411	0.411
	3	0.444	0.351	0.442	0.442
	4	0.414	0.374	0.475	0.990
	5	0.382	0.396	0.501	0.501
	6	0.350	0.421	0.546	0.546
	7	0.318	0.449	0.585	0.010
	8	0.286	0.480	0.627	0.010
2	1	0.494	0.319	0.392	0.392
	2	0.464	0.338	0.421	0.421
	3	0.433	0.358	0.453	0.453
	4	0.403	0.381	0.486	0.486
	5	0.307	0.458	0.599	0.599
	6	0.340	0.430	0.556	0.556
	7	0.372	0.405	0.521	0.990
	8	0.484	0.325	0.402	0.402
3	1	0.329	0.440	0.572	0.990
	2	0.296	0.469	0.613	0.010
	3	0.262	0.500	0.656	0.010
	4	0.274	0.490	0.641	0.020
	5	0.455	0.345	0.432	0.432
	6	0.424	0.365	0.463	0.463
	7	0.392	0.388	0.497	0.497
	8	0.361	0.413	0.534	0.534

6. FINAL REMARKS

The limited computational experience we had thus far with the algorithms (using simulated data) seems to indicate that the myopic approach produces slightly better allocations than the presbyopic one and that both approaches are comparable, in terms of performance, with the exact stochastic dynamic programming approach and vastly outperform the latter in terms of computational effort. Of course, the extent to which those observations are generalizeable is not known at this point. Further experience with those algorithms is necessary in order to draw general conclusions.

TABLE 2: Expected (E_k) and actual (A_k) number of targets held at each time stage k when a myopic or a presbyopic allocation policy is followed.

	Myopic		Presbyopic	
k	E_k	A_k	E_k	A_k
1	1.880	3	1.882	1
2	1.892	2	1.882	2
3	1.845	2	1.882	1
4	1.891	2	1.882	0
5	1.815	1	1.882	2
6	1.875	2	1.882	2
7	1.983	3	1.882	1
8	2.068	1	1.882	1
9	1.820	1	1.882	2
TOTAL	17.069	17	16.938	12

7. ACKNOWLEDGEMENTS

Work for this paper has been supported by contract N00014-79-C-0238 of the Office of Naval Research. Further details on resource allocation and the modeling of detection can be found in the doctoral dissertation of the second author, Ref. [22]. The second author is now with the Department of Naval Architecture and Marine Engineering of the University of Michigan at Ann Arbor.

8. REFERENCES

[1] Alspach, D.L. (1975). A Gaussian Sum Approach to the Multi-Target Tracking Problem. Automatica, Vol. II, 285-296.

[2] Bar Shalom, Y. (1974). Application of Stochastic Control Theory to Resource Allocation Under Uncertainty. IEE Trans. Autom. Control, Vol. AC-19, No. 1.

[3] Bar Shalom, Y. and E. Tse (1975). Tracking in a Cluttered Environment with Probabilistic Data Association, Automatica, Vol. II, 451-460.

[4] Bar Shalom Y. and A.I. Cohen (1976). Optimal Resource Allocation for an Environmental Surveillance System, IEEE Trans. on Systems, Man and Cybernetics, Vol. SMC-6, No. 6.

[5] Bar Shalom, Y. (1978). Tracking Methods In a Multitarget Environment, IEEE Trans. Autom. Control, Vol. AC-23, 618-626.

[6] Bar Shalom, Y. and G.D. Marcus (1979). Tracking with Measurements of Uncertain Origin and Random Arrival Times, Proceedings, 18th Conference on Decision and Control, Ft. Lauderdale, Fl. Dec. 1979.

[7] Fortmann, T.E. and S. Baron (1977). Octopus, an Experimental Passive Sonar Tracking System. Proc. ONR Passive Target Conference, Monterey, Calif. (SECRET).

[8] Fortmann, T.E. and S. Baron (1978). Problems in Multi-Target Sonar Tracking. BBN report, N.E.S.C. #N00039-78-C-0296.

[9] Fortmann, T.E. et al (1980). Multi-Target Tracking Using Joint Probabilistic Data Assoc. Report for CDC 1980.

[10] Fortmann, T.E. et al (1981). Detection Thresholds for Multi-Target Tracking in Clutter. Report No. 4605, Bolt, Beranek and Newman, Inc.

[11] Friedlander, B. (1980). Multi-Traget Tracking Studies Phase I Final Report. Report No. 5334-01, Systems Control Inc.

[12] Keverian, K.M. and N.R. Sandell (1979). Multiobject Tracking by Adaptive Hypothesis Testing. LIDS-959, MIT Laboratory for Information and Decision Systems.

[13] Reid, D. (1979). An Algorithm for Tracking Multiple Targets. IEEE Trans. Autom. Control, Vol. AC-24, No. 12.

[14] Tenney, R.R. (1981). Optimal Sensor Scheduling for Multiple Hypothesis Testing. Working Paper, MIT.

[15] Psaraftis, H.N., A.N. Perakis and P.N. Mikhalevsky (1981a). New Models on the Ocean Acoustic Detection Process. J Acoust. Soc. Am. 69(6), June 1981, 1724-1734.

[16] Psaraftis, H.N., A.N. Perakis and P.N. Mikhalevsky (1981b). Memory Detection Models for Phase-Random Ocean Acoustic Fluctuations. International Conference on Communications, Denver, Co.

[17] Hamblen, W.R. (1977). A Phase Random Multipath Model for Acoustic Signal Fluctuation in the Ocean and its Comparison with Data. PhD. Thesis, MIT, Dept. of Ocean Engineering.

[18] Mikhalevsky, P.N. and I. Dyer (1978). J. Acoust. Soc. Am. 63, 732-738.

[19] Mikhalevsky, P.N. (1979). J. Acoust. Soc. Am. 66. 751-762.

[20] Mikhalevsky, P.N. (1980). J. Acoust. Soc. Am. 67, 812-815.

[21] Psaraftis, H.N. and A.N. Perakis (1981). A Basic Problem of Resource Allocation in Target Tracking. Working Paper, OE-ONR-81-3, MIT.

[22] Perakis, A.N. (1982). Contributions to the Problem of Resource Allocation in Target Tracking Using New Models in Ocean Acoustic Detection, PhD. Thesis, MIT. Dept. of Ocean Engineering, May 1982.

NONLINEAR DATA FUSION

Dr. David A. Castanon and Dr. Demosthenis Teneketzis

ALPHATECH, Inc., 3 New England Executive Park, Burlington, Massachusetts

Abstract. In this paper, we consider the problem of combining the local conditional distributions of a random variable which have been generated by local observers, having access to their private information. Sufficient statistics for the local distributions are communicated to a coordinator, who attempts to reconstruct the global centralized distribution using only the communicated statistics. We obtain a centralized processing algorithm which recovers exactly the centralized conditional distribution. The results can be applied in designing distributed hypothesis-testing algorithms for event-driven systems.

SECTION 1. INTRODUCTION

Consider the following estimation problem: The state trajectory of a random process is observed by K distinct observers, using noise-corrupted observations. Each observer processes his own observation history, to obtain the local conditional distribution of the state, as a function of time. Assume that sufficient statistics representing each local conditional distribution are communicated to a coordinator at a central location at each point in time. The coordinator's estimation problem consists of constructing the overall conditional distribution of the state, conditioned on knowing all of the observations, while using only the sufficient statistics communicated to him. The above estimation structure is illustrated in Fig. 1.

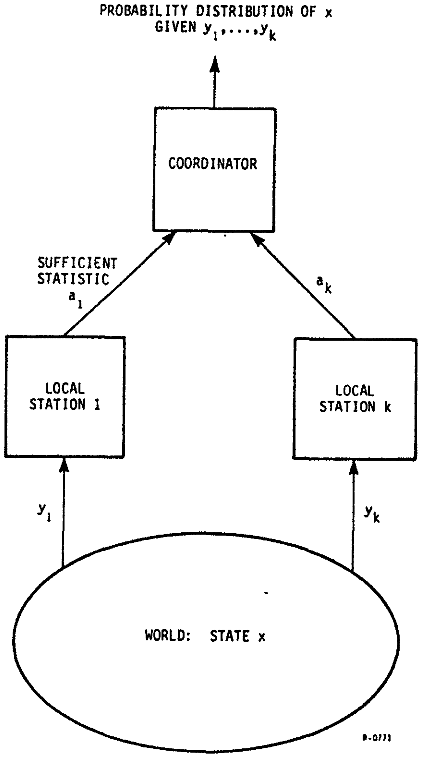


Fig. 1. The Problem of Data Fusion

When the state process is a Gauss-Markov process, and the local observations are linear measurements of the state corrupted by the noise, the solution to the coordinator's problem has been obtained by many authors, notably Speyer [1], Chong [2], and Willsky et al. [3]. In this case, the sufficient statistic is provided by the local conditional mean and covariance. The results of [1], [2] show that the centralized conditional mean and covariance can be obtained using linear operations on the local estimates and covariances. The results of [3] extend these results to consider problems in optimal smoothing, as well as problems where the local models used in producing local estimates differ from the true global model available to the coordinator.

In this paper, we extend the results of [1], [2] and [3] to include general Markov stochastic processes. We deal both with discrete-time and continuous-time Markov processes. For discrete-time Markov processes, we derive the solution of the coordinator's problem using Bayes' rule. The structure of the coordinator's solution is a generalization of the results of [2] to general Markov processes.

In continuous time, we discuss the problem of estimating a Markov process observed through additive white noise. We use recent results in nonlinear filtering to [4], [5] to characterize the evolution of the local and the centralized conditional probability densities of the state. Based on the solution structure for the discrete-time case, we develop the equations for the optimal coordination algorithm.

The results for the discrete and continuous time coordination problem are used in designing optimal hierarchical estimation and hypothesis testing algorithms for a class of event-driven systems. These algorithms provide the basis for designing distributed hypothesis testing algorithms which perform as well as centralized algorithms.

The rest of the paper is organized as follows. In Section 2, we introduce the discrete time formulation of the coordinator's problem; Theorem 1 in that section presents the solution to the coordinator's problem. In Section 3, we introduce the continuous time version of the coordinator's problem, and solve it in Theorem 2. Section 4 illustrates the applications of Section 2 to the problem of event detection in event-driven systems. Section 5 summarizes the results and indicates briefly the directions of future research. The results will be presented without proof for the sake of exposition. The proofs are available in a longer paper [6].

SECTION 2. THE DISCRETE-TIME COORDINATOR'S PROBLEM

Consider the diagram of Fig. 1. The state process is assumed to be a hybrid Markov process $z_t = (x_t \rho_t)$, evolving in discrete time, where x_t takes its values in R^n, and ρ_t is a member of a discrete space S with the discrete topology. The transition probability distribution of z_t is denoted as

$$P_t(A;z_t) \triangleq \text{Prob}\ \{z_{t+1} \in A \mid z_t\} \qquad (2.1)$$

There are K local agents taking observations of the state process. The observations of agent i at stage t are defined as

$$Y_t^i = h_t^i\ (z_t,\ v_t^i) \qquad (2.2)$$

where v_t^i is an R^n-valued random variable, and h is a jointly-measurable function from $R^n \times S \times R^n$ into R^n.

The vector v_t^i represents the local observation noise of agent i at time t. In order to properly define the estimation problem, we must establish the relationship between the various random variables. Let $(\Omega,\ F,\ P)$ be the underlying probability space for the random sequences z_t, v_t^i, $i=1,\ldots n$, $t=0,\ldots T$. We make the following assumptions:

(1) Under P, the sequences z_t, v_t^i, v_t^j are mutually independent, for $i \neq j$, $i,j=1,\ldots K$.

(2) There exists σ-finite measures $\lambda_t^i(dy)$, and jointly measurable, non-negative functions $g_t^i(y_t^i,\ z_t)$ such that

$$\text{Prob}\{y_t^i \in A \mid z_t\} = P\{\omega \mid h_t^i(z_t, v_t(\omega)) \in A\}$$

$$= \int_A g_t^i(y,z_t)\ \lambda_t^i(dy) \qquad (2.3)$$

Assumption 1 states that the local observation noises are mutually independent, and independent from the state. Assumption 2 states that the transition distribution defining y_t^i from z_t has a density with respect to some σ-finite measure. We will also assume that, for each i, the noise v_t^i is a "white noise" sequence.

(3) Under P, $v_{t_1}^i$ is independent of $v_{t_2}^i$ if $t_1 \neq t_2$.

As a final assumption, we will provide some added structure to $P_t(dz_t, z_{t-1})$.

(4) There exists a σ-finite measure $\alpha_t(dz_t)$, and a jointly measurable function $p_t(z_t, z_{t-1})$, such that $P_t(dz_t, z_{t-1}) = p_t(z_t, z_{t-1})\ \alpha(dz_t)$.

Now, consider the estimation problem of each local agent. The purpose of each local agent is to produce the conditional distribution of z_t, given all of the past observations $y_o^i, \ldots y_t^i$. Define the notation Y_t^i as

$$Y_t^i = \{y_o^i, \ldots y_t^i\}$$

$$Y_t = \{Y_t^i, i=1,\ldots K\}$$

Then, agent i's problem is solved recursively as [7]:

$$P\{z_t \in A \mid Y_t^i\} = \frac{J_t^i(A, Y_t^i)}{J_t^i(R^n \times S, Y_t^i)}$$

$$= \int_A p(dz_t,\ Y_t^i) \qquad (2.4)$$

where

$A \subset R^n \times S$ is a Borel set.

$$J^i_t(A,Y^i_t) = \int_{R^n\times S} \left(\int_A g^i_t(y^i_t,z_t)P_t(dz_t,z_{t-1})\right) P(dz_{t-1},Y^i_{t-1}) \qquad (2.5)$$

The derivation of these equations is standard, and can be found in [7] or [8].

Assumptions 2 and 4 are sufficient to guarantee that (2.5) has a density, with respect to the σ-finite measure $\alpha_t(dz_t)$. That is,

$$J^i_t(A,Y^i_t) = \int_A j^i_t(z_t,Y^i_t)\, \alpha_t(dz_t) \qquad (2.6)$$

where j^i_t is a non-negative, jointly measurable function given by

$$j^i_t(z_t,Y^i_t) = g^i_t(y^i_t,z_t) \int_{R^n\times S} p_t(z_t,z_{t-1}) P(dz_{t-1},Y^i_{t-1}) \qquad (2.7)$$

Equations 2.4 - 2.7 are statements of Bayes' rule. Formally, if all of the distributions were continuous with respect to Lebesgue measure, Eq. 2.7 can be rewritten as

$$p(z_t|Y^i_t) = \frac{p(y^i_t|z_t,Y_{t-1})\; p(z_t|Y_{t-1})}{p(y^i_t|Y^i_{t-1})} \qquad (2.8)$$

$$= \frac{p(y^i_t|z_t) \int_{R^n\times S} p(z_t|z_{t-1})p(z_{t-1}|Y^i_{t-1})dz_{t-1}}{\int_{Rn\times S} p(y^i_t|z_t)p(z_t|Y^i_{t-1})\; dz_t} \qquad (2.9)$$

Equations 2.5 and 2.7 make explicit use of assumption 3 to establish that $g^i_t(y^i_t,z_t)$ is also the density of the conditional distribution $P\{y^i_t \in A \mid z_t, Y^i_{t-1}\}$.

Our assumptions (1) - (4) are sufficient to establish that regular conditional probability distributions exist whenever necessary.

The coordinator's problem can be described as follows: At each t, each agent i communicates a sufficient statistic for the measure $j^i_t(dz_t, Y^i_t)$ to the coordinator. The coordinator must use these measures j^i_t to obtain a representation of the overall centralized probability distribution $P\{z_t \in A|\; Y^i_t,\; i-1,\ldots K\}$.

The solution to the coordinator's problem is given in the following Theorem.

Theorem 1: Under assumptions 1-4, the coordination can reconstruct the centralized conditional probability distribution recursively as

$$\Pr\{z_t \in A \mid Y_t\} = \frac{J_t(A, Y_t)}{J_t(R^n\times S,Y_t)} \qquad (2.10)$$

where

$$J_t(A,Y_t) = \int_A \prod_{i=1}^{K} j^i_t(z_t,Y^i_t) \frac{\int_{R^n\times S} p_t(z_t,z_{t-1})\; P(dz_{t-1},Y_{t-1})}{\prod_{i=1}^{K} \int_{R^n\times S} p_t(z_t,z_{t-1})\; P(dz_{t-1},Y^i_{t-1})}\, \alpha_t(dz_t) \qquad (2.11)$$

$$= \int_A \prod_{i=1}^{K} g^i_t(z_t,Y^i_t) \frac{1}{c_t(z_t)}\, \alpha_t(dz_t)$$

which defines $c_t(z)$.

The structure of the coordinator's solution in Theorem 1 is interesting. Basically, the coordinator accounts for the presence of correlations between the local estimators, due to the fact that they all observe the same state process, by compensating the product of the local conditional densities with the ratio of the centralized conditional predicted density and the product of the local conditional predicted densities. Formally,

$$\frac{p(z_t|Y_t)}{\prod_{i=1}^{K} p(z_t|Y^i_t)} = K_t \frac{p(z_t|Y_{t-1})}{\prod_{i=1}^{K} p(z_t|Y^i_{t-1})}$$

where K_t is a proportionality constant, to account for the normalization of the probability densities.

SECTION 3. THE CONTINUOUS-TIME COORDINATOR'S PROBLEMS

Assume that the state process can be described by the stochastic differential equation

$$dx_t = f(t,x_t,\rho_t)dt + \sigma(t,x_t,\rho_t)dw_t \qquad (3.1)$$

where w_t is a standard Brownian motion with values in R^n, and where ρ_t is an element of a finite set S = {1,...N} whose transitions are described in terms of the infinitesimal rates

$$P\{\rho_{t+\Delta}=i\,|\,\rho_t=j,x_t=x\} = \lambda_{ji}(x)\Delta+o(\Delta) \qquad (3.2)$$

Under appropriate assumptions, (3.1) and (3.2) define the evolution of a strong Markov process $(x_t,\rho_t) = z_t$ with values in $R^n\times S$. For our purposes, we assume the following:

(5) $\sigma(t,x,\rho)$, $b(t,x,\rho)$ and $\lambda_{ji}(x)$ are continuous, bounded functions on $R^+ \times R^n \times S$.

(6) Let $a(t,x,\rho) = \sigma(t,x,\rho)\sigma^T(t,x,\rho)$. Then, $a(t,x,\rho) \geq \alpha I$ for all (t,x,ρ) in $R^+ \times R^n \times S$, some $\alpha > 0$.

(7)
$$\|a(t,x,\rho) - a(t,y,\rho)\|^2 + \|b(t,x,\rho) - b(t,y,\rho)\|^2 \leq C_1(1 + \|x-y\|^2)$$
$$\|\alpha(t,x,\rho)\|^2 + \|b(t,x,\rho)\|^2 \leq C_2(1 + \|x\|^2)$$

for C_1, $C_2 > 0$, for all $(t,\rho)\varepsilon R^+\times S$, $x,y\varepsilon R^n$.

Assumptions (5)-(7) guarantee the existence of a strong Markov process (x_t, ρ_t) which is Feller continuous for any x_o, ρ_o [9]. In addition, we assume that the initial distribution of x_o, ρ_o is known. Suppose that each local station i=1,...K has measurements of the state process z_t, described by the stochastic differential equation

$$d\,y_t^i = h^i(t,x_t,\rho_t)dt + d\,v_t^i \qquad (3.3)$$

where v^i are mutually independent standard Brownian notions which are independent of w_t. We assume additionally that

(8) $\lambda_{ij}(x) \geq \varepsilon > 0$ for all $1 \leq i, j \leq K$.

(9) The functions f, σ and h^i, i=1,...K are smooth enough so that conditional probability densities of x_t given the local information Y_t^i exist for each local station i.

Conditions which guarantee the existence and smoothness of the conditional probability density function can be found in [4] and [5]. Typically, the functions h^i will be assumed to be uniformly bounded with bounded x derivatives.

Under these assumptions, the solution to each agent's problem can be obtained using the equations for optimal nonlinear filtering [4], [5]. As before, denote the information available at time t as

$$Y_t^i = \{y_s^i,\ o \leq s \leq t\}$$

$$Y_t = \{Y_t^i,\ \dots\ Y_t^K\}$$

Define A_t as the differential operator on functions on $R^n\times S$ into R as

$$A_t\, v(x,\rho) = \sum_{i=1}^{n} f_i(t,x,\rho)\,\frac{\partial}{\partial x_i}\,v(x,\rho)$$

$$+\ \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}(t,x,\rho)\,\frac{\partial^2}{\partial x_i\partial x_j}\,v(x,\rho)$$

$$+\ \sum_{i=1}^{K} \lambda_{\rho i}(x)\ (v(x,i) - v(x,\rho)) \qquad (3.4)$$

The solution to each local agent's problem is described by Zakai's equation [10] for the unnormalized conditional density of the state z_t given the observations Y_t^i. Let $q_t^i(z)$ denote this density for the local station i. Then,

$$dq_t^i(z) = A_t^*(z)\, q_t^i(z)\, dt + h_t^{iT}(z)\, q_t^i(z)\, dy_t^i \qquad (3.5)$$

where A_t^* is the formal adjoint of the operator A defined in (3.4). The differentials used in (3.5) are the Ito differentials; in Stratonovich form [11], using symmetric differentials, Eq. 3.5 becomes

$$\bar{d}\, q_t^i(z) = (A_t^*(z) - \frac{1}{2}\, h_t^{iT}(z)\, h_t^i(z))\, q_t^i(z)\, dt$$

$$+\ h_t^{iT}(z)\, \bar{d}\, y_t^i \qquad (3.6)$$

where we have adopted the notation $\bar{d}$ of [11] to indicate the Stratonovich symmetric differential.

We formulate the coordinator's problem as follows: At each time t, the coordinator receives, from each location, a multiple of $q_t^i(z)$, or a sufficient statistic which enables him to reconstruct $q_t^i(z)$ exactly. Denote the coordinator's received message as

$$a_t^i(z) = k_t^i\, q_t^i(z)$$

where k_t^i is the scale constant at time t. The solution of the coordinator's problem is summarized in Theorem 2.

Theorem 2: Under assumptions 5-9, the coordinator can reconstruct the centralized conditional probability distribution recursively as

$$P(z_t \varepsilon A \mid Y_t) = \frac{\int_A q_t(z)\, dz}{\int_{R^n \times S} q_t(z)\, dz}$$

where

$$q_t(z) = \frac{\prod_{i=1}^{k} a_t^i(z)}{C_t(z)}$$

and C_t satisfies

$$\frac{1}{C_t}\frac{dC_t(z)}{dt} = \sum_{i=1}^{K} \frac{A_t^* a_t^i(z)}{a_t^i(z)} - \frac{C_t(z)}{\prod_{i=1}^{K} a_t^i(z)} A_t^* \frac{\prod_{i=1}^{K} a_t^i(z)}{C_t(z)} \qquad (3.7)$$

$$C_o(z) = P_o(z)^{n-1}$$

where $P_o(z)$ is the initial density of z_o.

A direct analogy between Theorems 1 and 2 can be drawn as follows: $\frac{A_t^* q_t^i(z)}{q_t^i(z)}$ correspond to predictors in the discrete time case, as in the heuristic formula

$$\ln (P^i_{t+dt}(z|Y_t)) = \ln P^i_t(z|Y_t) + \frac{A_t^* q_t^i(z)}{q_t^i(z)}\, dt \qquad (3.8)$$

Furthermore

$$\frac{-A_t^* q_t(z)}{q_t(z)}\, dt = \ln P_t(z|Y_t) - \ln (P_{t+dt}(z|Y_t)) \qquad (3.9)$$

Substituting into (3.7) gives (formally)

$$\ln C_{t+dt}(z) - \ln C_t(z) = \ln \frac{\prod_{i=1}^{K} P^i_{t+dt}(z|Y_t)}{P_{t+dt}(z|Y_t)} - \ln C_t(z) - \ln K_t \qquad (3.10)$$

where K_t is a constant factor due to the scalings in the definition of $C_t(z)$. Hence,

$$C_{t+dt}(z) = K_t \frac{\prod_{i=1}^{K} P^i_{t+dt}(z|Y_t)}{P_{t+dt}(z|Y_t)} \qquad (3.11)$$

which is the ratio of the product of the decentralized predictors to the centralized predictor. Thus, C_t is the same compensator as was used in Theorem 1.

SECTION 4. DISTRIBUTED EVENT DETECTION: THE DISCRETE TIME CASE

The results of Sections 2 and 3 are of limited use unless a finite-dimensional sufficient statistic can be determined for each local conditional distribution, so that only finite dimensional values are transmitted to the coordinator.

In the previous works [1] -[3], the z_t process was assumed Gauss-Markov, with linear observations, so that the local conditional means were the on-line sufficient statistics to be communicated to the coordinator. In this section, we extend these results to a special kind of hybrid system, representing a problem of event detection.

Assume that, at time 0, an event H occurs, which is one of a finite set $E = \{H_1, \ldots H_N\}$ of possible events. The probability of each event is described a priori by P(H). The event influences the evolution of a Gaus-Markov discrete time R^n-valued process, described as

$$x_{t+1} = A_t(H)\, x_t + B_t(H)\, w_t \qquad (4.1)$$

This state process is being observed by K independent stations, whose measurements are described as

$$y_t = c_t^i(H)\, x_t + v_t^i \qquad (4.2)$$

The sequences $\{w_t\}$, $\{v_t^i\}$, $\{v_t^j\}$ and the initial state x_o are mutually independent, Gaussian random variables with distribution $w_t \sim N(o,I)$.

$v^i_t \sim N(o, R^i_t)$

$x_0 \sim N(\bar{x}_0, M_0)$ with $M_0>0$ (4.3)

Define the overall state at time t as (H, x_t). The local conditional density of (H, x_t) given Y^i_t can be summarized by an on-line finite-dimensional sufficient statistic, which is

$$S^i_t = \left\{ \begin{array}{l} P(H=H_1|Y^i_t) \\ \cdot \\ \cdot \\ \cdot \\ P(H=H_N|Y^i_t) \\ \hat{x}^{i1}_t \\ \cdot \\ \cdot \\ \cdot \\ \hat{x}^{iN}_t \end{array} \right\} \quad (4.4)$$

where $P(H=H_l|Y^i_t) \triangleq P^{il}_t$ is the conditional probability that $H=H_l$, given Y^i, and $\hat{x}^{ij}_t = E\{x_t|Y^i_t, H=H_j\}$.

The local conditional density of (H, x_t) given Y^i_t can be written as

$$p(x_t, H_j|Y^i_t) = p^{ij}_t \frac{1}{|\det \Sigma^{ij}_t|(2\Pi)^{n/2}}{}^{-\frac{1}{2}(x_t - \hat{x}^{ij}_t)^T(\Sigma^{ij}_t)^{-1}(x_t - \hat{x}^{ij}_t)} \quad (4.5)$$

where $\Sigma^{ij}_t = E\{x_t - \hat{x}^{ij}_t)(x_t - \hat{x}^{ij}_t)^T|Y^i_t, H=H_j\}$ can be computed a priori, independent of Y^i_t, from the parameters of the problem.

The system (4.1) - (4.3) satisfies all of the assumptions of Section 2. Furthermore, the centralized conditional density has a sufficient statistic S_t, defined as

$$S_t = \left\{ \begin{array}{l} p^1_t \\ \cdot \\ \cdot \\ \cdot \\ p^n_t \\ \hat{x}^1_t \\ \cdot \\ \cdot \\ \cdot \\ \hat{x}^n_t \end{array} \right\} \quad (4.6)$$

where P^j_t, $\hat{x}^j_t$ are defined as before, with Y_t replacing Y^i_t. Hence, the coordinator's task is to convert the inputs $\{S^i_t, i=1,...K\}$ which he receives, together with his previous sufficient statistic S_{t-1}, to obtain the sufficient statistic S_t. The solution to the coordinator's problem is described in Theorems 3 and 4.

Theorem 3: The centralized estimate $\hat{x}^k_t$ of x_t, given Y_t and assuming $H = H_k$, is given by

$$\hat{x}^k_t = \sum_{i=1}^{K} \{\Sigma^k_t((\Sigma^{ik}_t)^{-1}\hat{x}^{ik}_t - (M^{ik}_t)^{-1}\bar{x}^{ik}_t)\} + \Sigma^k_t(M^k_t)^{-1}\bar{x}^k_t \quad (4.7)$$

where

$$\bar{x}^k_{t+1} = A_t(H_k)\hat{x}^k_t \quad (4.8)$$

$$M^{ik}_{t+1} = A_t(H_k)\,\Sigma^{ik}_t\,A^T_t(H_k) + B_t(H_k)\,B^T_t(H_k) \quad (4.9)$$

$$M^k_{t+1} = A_t(H_k)\,\Sigma^k_t\,A^T_t(H_k) + B_t(H_k)\,B^T_t(H_k) \quad (4.10)$$

$$\Sigma^{ik}_t(H_k) = ((M^{ik}_t)^{-1} + C^{iT}_t(H_k)(R^i_t)^{-1}C^i_t(H_k))^{-1} \quad (4.11)$$

$$\Sigma^k_t = (\sum_{i=1}^{K}((\Sigma^{ik}_t)^{-1} - (M^{ik}_t)^{-1}) + (M^k_t)^{-1})^{-1} \quad (4.12)$$

$$\bar{x}^{ik}_{t+1} = A_t(H_k)\,\hat{x}^{ik}_t \quad (4.13)$$

The proof of this theorem can be found in [2]. Note that Eq. 4.7 reconstructs the centralized conditional mean from the local conditional means, the local predicted mean, and the centralized predicted mean. The matrices in Eqs. 4.9 - 4.12 can be determined off-line as parameters of the problem. The results of Theorem 3 enable the coordinator to reconstruct part of the sufficient statistic (4.6) from the information available in the transmissions S^i_t, i=1,...K.

The remaining part can be constructed using the results of Theorem 4.

Theorem 4: The centralized conditional probability of hypothesis k given Y_t, P_t^k, is given by

$$P_t^k = \frac{Q_t^k}{\sum_{i=1}^{N} Q_t^i} \tag{4.14}$$

where

$$Q_t^k = \prod_{i=1}^{K} \frac{P_t^{ik}\, G(\hat{x}_t^{ik}, \Sigma_t^{ik})}{P_{t-1}^{ik}\, G(\bar{x}_t^{ik}, M_t^{ik})} \; \frac{P_{t-1}^{k}\, G(\bar{x}_t^{k}, M_t^{k})}{G(\hat{x}_t^{k}, \Sigma_t^{k})} \tag{4.15}$$

and $G(x,\Sigma)$ is a Gaussian kernel with mean x, covariance Σ.

Theorem 4 enables us to constant the centralized conditional probabilities that each hypothesis H_k is true, given the local sufficient statistics S_t^i, i=1,...K. Note that this computation involves reconstruction of the local and centralized Gaussian densities described in Theorem 3.

SECTION 5. CONCLUSION

The algorithm presented in Sections 2 and 3 solves the problem of data fusion for nonlinear systems under the conditions that, at each instant of time, each local station transmits a sufficient statistic of its conditional density of the state, given its observation record. It is implicitly assumed that such a sufficient statistic is the natural output of each local station, which attempts to reconstruct the state process. The conditions which we have imposed permit exact reconstruction of the overall centralized conditional distribution of the state process, given the local conditional distributions, using the algorithms of Theorems 1 and 2.

In most practical applications, the rate of communications between local stations and the coordinator will be substantially lower than the rate at which the local stations acquire measurements of the state. In these cases, the local information must be compressed, and exact reconstruction of the centralized conditional distribution of the state may not be possible at the coordinator's level. We are currently investigating the design of algorithms to solve the data fusion problem with restrictions on the frequency of communications.

REFERENCES

1. Speyer, J.L., "Computation and transmission requirements for a decentralized linear-quadratic-Gaussian control problem," IEEE Trans. Automat. Contr., vol. AC-24 pp. 266-269, Apr. 1979.
2. Chong, C.Y., "Hierarchical estimation," presented at the 2nd MIT/ONR Workshop Distributed Inform. Decision Syst. Motivated Naval Command-Contr.-Commun. (C^3) Problems, Naval Postgraduate School, Monterey, CA, July 1979.
3. Willsky, A.S., M. Bello, D.A. Castanon, B.C. Levy, and G. Verghese, "Combining and updating of local estimates and regional maps along sets of one-dimensional tracks," IEEE Trans. Auto. Control, Vol. AC-27, No. 4, August 1982.
4. Pardoux, E., "Stochastic partial differential equations and filtering of diffusion processes," Stochastics, 2, 1979, pp. 127-168.
5. Lipster, R.S. and A.N. Shiryaev, Statistics of Random Processes I, New York: Springer-Verlag, 1977.
6. Castanon, D.A., D. Teneketzis, "Nonlinear Data Fusion," in preparation (1982).
7. Striebel, C.A., Optimal Control of Discrete Time Stochastic Systems, New York: Springer-Verlag, 1975.
8. Loeve, M., Probability Theory, VanNostrand, New York, 1963.
9. Strook, D.W., S.R.S. Varadhan, Multidimensional Diffusion Processes, New York: Springer-Verlag, 1979.
10. Zakai, M., "On the Optimal Filtering of Diffusion Processes," Z. Wahr. Ver. Geb., 11, 1969.

OPTIMAL SMOOTHING AND ESTIMATION FOR HYBRID STATE PROCESSESS[1]

by

F. Bruneau[2]

R.R. Tenney[3]

Laboratory for Information and Decision Systems
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

ABSTRACT

Consider the estimation and smoothing problem for a hierarchical Markov process. The supremal state evolves autonomously; infemal dynamics and observations may be statistically dependent on the supremal state. This class of processes has more structure than a general Markov process; the implications of this structure are developed here. Of special interest is the case of hybrid systems, where the supremal state is discrete and the infemal dynamics are linear and Gaussian. This structure commonly appears in diverse applications, including failure detection, maneuvering target tracking, and digital communications on analog channels. It is also the structure for which the most useful conclusions can be drawn.

I. OPTIMAL SMOOTHING AND ESTIMATION FOR HYBRID STATE PROCESSES

The special structure considered here involves a Markov process with state space $\underline{X}$ which can be decomposed into subspaces $\underline{X}_1 \times \underline{X}_2$, and where the dynamics on $\underline{X}_1$ are independent of $\underline{X}_2$, but not vice-versa. The observation space $\underline{Y}$ can be decomposed compatibly. This structure lies at the heart of several important applications, particularly in hybrid systems where $\underline{X}_1$ is discrete (modeling failure modes, maneuver modes, or digital symbols) and $\underline{X}_2$ continuous (modeling system dynamics, target trajectories, or channel dynamics, respectively). These problems are usually dominated by the entire discrete state sequence. Many ad hoc solutions to these types of problems have appeared in the literature [1-3], where approximations are required in order to overcome the exponential growth of the set of discrete state trajectories as the time horizon of the problem advances. Now that combinational problems are not necessarily computationally unassailable, it is worthwhile understanding the extent to which these problems can be solved <u>exactly</u>. While the result may still be unimplementable, inclusion of computation-reducing features which do not affect performance (and which do exist) certainly provides a starting point for other modifications.

This paper develops optimal methods for approaching the filtering and smoothing problems for systems with the above structure. The contributions are of two types: specific techniques for reducing the complexity of hybrid system estimation algorithms, and a general structure for approaching this class of problems. The techniques and approach seem quite helpful in designing algorithms for VLSI implementation, but do not entirely solve the problem. As an example will show, the specific techniques developed here may reduce the combinatorial growth of a problem from exponential to linear (in time); this is helpful, but still not practical, and approximations must also be introduced. Thus a prime purpose of this work is to delimit the power of exact techniques, and create a framework for future performance analysis of approximate techniques.

The development begins with a formal problem statement, followed by the derivations of optimal filtering and smoothing techniques in a general setting. These are then specialized to the linear-Gaussian and hybrid linear-Gaussian cases.

II. PROBLEM STATEMENT

A. Models

Let the state space of a Markov process be $\underline{X}_1 = \underline{X}_1 \times \underline{X}_2$. $\underline{X}_1$ is the state space of the <u>supremal subsystem</u> which evolves autonomously; $\underline{X}_2$ that of the <u>infemal subsystem</u> which is dependent upon the value of the

---

[1] Support from ONR contract N00014-77-0532C (NR 041-519) is gratefully acknowledge.

[2] 106 Rue Charles Lafitte, 92200 Neuilly, France.

[3] MIT/LIDS Room 35-213, Cambridge, MA, 02139. Address all inquiries to this author.

supremal state $x_1(t)$. Formally, we make[4]:

Assumption 1: The state transition probabilities factor as

$$p(x_1(t+1),x_2(t+1)|x_1(t),x_2(t)) =$$
$$p(x_1(t+1)|x_1(t))p(x_2(t+1)|x_1(t),x_2(t))$$

The process is observed via the space $\underline{Y} = \underline{Y}_1 \times \underline{Y}_2$, where $\underline{Y}_1$ contains observations of the supremal state only, and $\underline{Y}_2$ of the joint state. Again, make

Assumption 2: The observation probabilities factor as

$$p(y_1(t),y_2(t)|x_1(t),x_2(t))=$$
$$p(y_1(t)|x_1(t))p(y_2(t)|x_1(t),x_2(t))$$

We will be interested in the maximum a posteriori (MAP) estimates of the state (filtering), or entire state trajectory (smoothing), conditioned on a sequence of observations received from the system. Introducing the notation

$$X_i(t)\in \underline{X}_i^t \qquad Y_i(t)\in \underline{Y}_i^t$$

for sequences of states and observations (over a time interval $\tau\in\{1,\dots,t\}$), the problem is

Assumption 3: Find

a) for the filtering problem the state $x^*(t)$ which maximizes $p(x(t)|Y(t))$

b) for the smoothing problem, the state trajectory $X^*(t)$ maximizing $p(X(t)|Y(t))$.

This is the general problem. Two special cases which are of interest are the linear-Gaussian, and the discrete/linear-Gaussian (hybrid) structures. In the former, assume that $\underline{X}_i = R^{n_i}$, and that the system dynamics are linear with additive white Gaussian driving noise. (2.1) becomes

Assumption 1L: The hierarchical dynamics[5] are:

$$\vec{x}_1(t+1)=\underline{A}_{11}\vec{x}_1(t)+\vec{w}_1(t) \qquad \vec{w}_1(t)\sim N(\vec{0},\underline{Q}_1)$$
$$\vec{x}_2(t+1)=\underline{A}_{21}\vec{x}_1(t)+\underline{A}_{22}\vec{x}_2(t)+\vec{w}_2(t)\,\vec{w}_2(t)\sim N(\vec{0},\underline{Q}_2)$$

where $\underline{Q}_1$ and $\underline{Q}_2$ are positive definite, and $\vec{w}_1$ and $\vec{w}_2$ are independent of one another.

Similarly, the observations lie in $Y_1 = R^{m_1}$, and

Assumption 2L: The observation equations are

$$\vec{y}_1(t)=\underline{C}_{11}\vec{x}_1(t)+\vec{v}_1(t) \qquad \vec{v}_1(t)\ N(\vec{0},\underline{R}_1)$$
$$\vec{y}_2(t)=\underline{C}_{21}\vec{x}_1(t)+\underline{C}_{22}\vec{x}_2(t)+\vec{v}_2(t) \qquad \vec{v}_2(t)\ N(\vec{0},\underline{R}_2)$$

where $\underline{R}_1$ and $\underline{R}_2$ are positive definite and v_1 and v_2 are jointly independent.

Thus the conditional distributions for the linear case $p(x_1(t+1)|x_1(t))$, etc. in (2-1) and (2-2) are all multivariable Gaussian densities with means and covariances specified by (2-4)-(2-7).

For hybrid models, a combination of discrete and continuous dynamics exist. The supremal system is discrete, specifying some structural mode, and the infemal is assumed linear-Gaussian, with descriptive matrices dependent upon the value of the supremal state. Thus $\underline{X}_1= \{x_1^1,\dots,x_1^{n_1}\}$, $\underline{X}_2 = R^{n_2}$, and

Assumption 1H: The dynamics are specified by

$$p(x_1(t+1)|x_1(t))$$
$$x_2(t+1)=\underline{A}(x_1(t))x_2(t)+w(t)$$
$$w(t)\sim N(0,Q(x_1)))$$

with $\underline{Q}$ positive definite.[6]

Finally, $Y_1=\{y_1,\dots,y_1^{m_1}\}$, $Y_2 = R^{m_2}$, and

Assumption 2H: The observations for a hybrid system are specified by

$$p(y_1(t)|x_1(t))$$
$$y_2(t)=\underline{C}(x_1(t))x_2(t)+v(t)$$
$$v(t)\ N(0,\underline{R}(x_1(t)))$$

with $\underline{R}$ positive definite.[6]

These are the three classes of models treated in sections III-V, respectively.

---

[4] For the general case derivations will be done formally. This is exact when $\underline{X}$ is discrete, and when all invoked distributions exist and are well defined.

[5] Extension to time-varying system and noise matrices is straightforward and not considered here for notational clarity. A number of other assumptions may be relaxed; the purpose here is to develop some new structure in the simplest setting possible.

[6] Nonzero, x_1-dependent means may be treated by state augmentation and the dependence of $\underline{A}$ or $\underline{C}$ on $x_1(t)$.

III. THE GENERAL CASE

This section develops the concepts, notation, and basic techniques for optimal filtering and smoothing under assumptions 1-3.

The filtering solution exhibits no special structure in this case.

A. Smoothing: Compact

We will consider two approaches to the smoothing problem, one the usual optimal algorithm, and the other an expanded version which better permits exploitation of the hierarchical structure at a cost of increased computation.

The suitability of the hierarchical structure to the smoothing problem is suggested by the fact that

$$\max_{X_1,X_2} p(X_1,X_2|Y_1,Y_2) = \frac{1}{p(Y_1,Y_2)} \times \qquad (*)$$

$$\max_{X_1}\{p(Y_1|X_1)p(X_1)\cdot\max_{X_2} p(Y_2|X_1,X_2)p(X_2\ X_1)\}\}$$

This is a direct result of assumptions 1 and 2. Note that it is not necessary to compute $p(Y_1,Y_2)$ at all: (*) suggests an algorithm by which the best X_2, is found for each X_1; and then the best X_1 is found. Unfortunately, $X_1(t)$ and $X_2(t)$ are elements of rather large sets.

However, this has not yet considered the Markov structure of the problem, which is essential to recursive smoothing techniques. Consider the smoothing solution on $\underline{X}_1 \times \underline{X}_2$; we will be interested in determining if (*) affects its structure.

Definition: A survivor function [7] $s(x(t)|Y(t))$ is defined by

$$s(x(t)|Y(t))=\max_{X(t-1)} p(Y(t)|x(t),X(t-1))p(x(t),X(t-1))$$

Technically, s is a function on $\underline{X} \times \underline{Y}^t$; since we will only be interested in evaluating it along a particular realization of the output process Y(t), it is convenient to view it as a function of x(t). It indicates the unnormalized probability of the most likely state trajectory X(t) which terminates in x(t), conditioned on the observation sequence Y(t). Note that the maximizing X(t-1) here may not be unique, but one of them may be selected and stored for each x(t). This permits reconstruction of the entire MAP state sequence by finding x(t) which maximizes $s(x(t)|Y(t))$, and then determining the X(t-1) thus associated with it.

The implications of Markov structure are that s is recursively computable.

Lemma 1: $s(x(t)|Y(t))$ may be computed via

$$s(x(t+1)|Y(t))=\max_{x(t)} p(x(t+1)|x(t))s(x(t)|Y(t))$$

$$s(x(t+1)|Y(t+1))=p(y(t+1)|x(t+1))s(x(t+1)|Y(t))$$

Proof: Bayes' theorem, interchange of max operations with functions not of the same variable, and the Markov assumptions:

$$p(y(t)|X(t))=\prod_{s=1}^{t} p(y(s)|x(s))$$

$$p(X(t))=\prod_{s=1}^{t-1} p(x(s+1)|x(s))\ p(x(1))$$

If s is replaced with $-\ell n(s)$, a monotonic operation, and the resulting function is minimized, the Viterbi algorithm [7] emerges.

Computationally, the Viterbi algorithm is relatively simple, requiring only $O(N_1N_2)$ operations per time step (for discrete $\underline{X}$). Memory for storing the preceeding trajectory associated with each x(t) is the dominant factor in its implementation. As in the filtering problem, however, the hierarchical structure of the Markov process does nothing, in general, to simplify the algorithm further. Again, the case where $x_1(t)=x_2(t)$ for all t generates an $s(x|Y)$ which is diagonal on $\underline{X}_1 \times \underline{X}_2$, and demonstrates the lack of decomposition.

B. Smoothing: Expanded

An algorithm for the smoothing problem can be constructed which does exploit the hierarchical structure, but at a great increase in computational complexity. As such, it is not useful for general problems of the class considered here, but it will be the key to the structure of the hybrid smoothing problem.

Definition: A conditional survivor function $s(x_2|X_1,Y_2)$ is defined as

$$s(x_2(t)|X_1(t),Y_2(t)) = \max_{X_2(t-1)} p(Y_2(t)|X_1(t),X_2(t))p(X_2(t)|X_1(t))$$

This function is an intermediary in the solution of the smoothing problem, as the second maximization in (*) can be rewritten as

$$\max_{X_2(t)} \{p(Y_2(t)|X_1(t),X_2(t))p(X_2(t)|X_1(t))\}=$$

$$\max_{x_2(t)} s(x_2(t)|X_1(t),Y_2(t))$$

These equations suggest an algorithm whereby $s(x_2|X_1,Y_2)$ is computed, based only on Y_2, for each X_1. The result may be summarized in the function

$$r(X_1(t)|Y_2(t)) \stackrel{\Delta}{=} \max_{x_2(t)} s(x_2(t)|X_1(t),Y_2(t))$$

The outer maximization in (*) is then over the product of $p(x_1|Y_1)p(X_1)$, which is computable just from the structure of the sumpremal system, and $r(X_1|Y_2)$, derived from the infemal structure only.

This algorithm does capitalize on the hierarchical structure, but leaves two questions to be answered. First, can the $s(x_2|X_1,Y_2)$ be computed recursively? Second, is there some recursive structure which can be exploited in the outer maximization, over X_1, without reducing the solution to a Viterbi algorithm on $\underline{X}_1$ x X_2? The answer to the latter is particularly critical, as the size of $\underline{X}_1^t$ grows exponentially with time.

The answer to both questions is yes. Consider the computation of $s(x_2|X_1,X_2)$ first.

Lemma 2: $s(x_2\ X_1,Y_2)$ may be computed as

a) predict:

$$s(x_2(t+1)|X_1(t),Y_2(t)) = \max_{x_2(t)} p(x_2(t+1)|x_1(t),x_2(t)) \cdot s(x_2(t)|X_1(t),Y_2(t))$$

b) update:

$$s(x_2(t+1)|X_1(t+1),Y_2(t+1)) = p(y_2(t+1)|x_1(t+1),x_2(t+1)) \, s(x_2(t+1)|X_1(t),Y_2(t))$$

Proof: Identical to Lemma 1, with conditioning on X_1.

The structure of these computations is straightforward. For each supremal trajectory $X_1(t)$, these implement a Viterbi calculation for the survivor function on x_2. Note that the explicit conditioning on X_1 removes the coupling between the statistics of x_2 and the supremal observations Y_1; X_1 provides a more complete statistical specification of the evolution of x_2 than does Y_1; it would be helpful if that X_1 still under consideration.

The tree of X_1 sequences may be pruned, in a way which guarantees that the MAP trajectory will never be eliminated yet preserves the hierarchical structure. This first requires:

Lemma 3: $p(Y_1(t)|X_1(t))p(X_1(t))$ may be computed recursively:

$$[p(Y_1(t+1)|X_1(t+1))p(X_1(t+1)]=p(y_1(t+1)|x_1(t+1) \; p(x_1(t+1)|x_1(t))[p(Y_1(t)|X_1(t))p(x_1(t))]$$

Proof: Elementary manipulations and the Markov properties.

Lemma 3 provides for the computation of the term other than $r(X_1|Y_2)$ - the term which captures the supremal dynamics through $p(x_1(t+1)|x_1(t))$, and the supremal observation through $p(y_1(t+1)|x_1(t+1))$.

One more notion is needed.

Definition: The sources of a state $x_1(t)$ are all trajectories in X_1^t terminating in x_1.

Theorem 1: Let $(X_1^*(t), x_2^*(t))$ be the MAP trajectory for observations $Y_1(t),Y_2(t)$. Let τ be any time preceeding t. If, at time τ

$$p(Y_1(\tau)|\tilde{X}_1(\tau))p(\tilde{X}_1(\tau))s(x_2(\tau)|\tilde{X}_1(\tau),Y_2(\tau)) \le p(Y_1(\tau)|X_1(\tau))p(X_1(\tau))s(x_2(\tau)|X_1(\tau),Y_2(\tau))$$

$$\max_{\{X_1(\tau)\}}$$

for each $x_2(\tau)$, where $\{X_1(\tau)\}$ contains all sources of $\tilde{x}_1(\tau)$ except $\tilde{X}_1(\tau)$ itself, then $\tilde{X}_1(\tau)$ will not be a subsequence of $X_1^*(t)$.

Proof: Consider the (compact) smoothing algorithm, and the $s(x_1(\tau),x_2(\tau)|Y_1(\tau),Y_2(\tau))$ computed by it. Each $(x_1(\tau),x_2(\tau))$ has a sequence $(X_1(\tau),X_2(\tau)$ associated with it which constitutes an optimal trajectory estimate through $(x_1(\tau),x_2(\tau))$. If $\tilde{X}_1(\tau)$ never appears as the first component of one of these associated sequences, it will not appear as a subsequence of any longer trajectory. $\tilde{X}_1(\tau)$ can only appear in association with states of the form $(\tilde{x}_1(\tau),x_2(\tau))$.

(3.17 assures that there is no $x_2(\tau)$ for which $\tilde{X}_1(\tau)$ is the most likely source of $\tilde{x}_1(\tau)$, hence $\tilde{X}_1(\tau)$ may be eliminated.

Theorem 1 establishes a looser requirement for eliminating trajectories than the compact smoothing algorithm. The Viterbi algorithm will eliminate trajectories at each point $(x_1(\tau), x_2(\tau))$, leaving only one candidate terminating there. (3-17) suggest eliminating $\tilde{X}_1(\tau)$ only if there is no $x_2(\tau)$ at all with which $\tilde{x}_1(\tau)$ may be paired and which preserves $\tilde{X}_1(\tau)$ as a candidate. An even looser criterion is given by

Corollary 1a: $\tilde{X}_1(\tau)$ will not be a subsequence of the optimal estimate if there exists some $X_1(\tau) \neq \tilde{X}_1(\tau)$, both sources of $x_1(\tau)$, where

$$p(X_1(\tau)|Y_1(\tau))s(x_2|\tilde{X}_1(\tau),Y_2(\tau)) \leq$$

$$p(x_1(\tau)|Y_1(\tau))=s(x_2|X_1(\tau)\ Y_2(\tau))$$

for every x_2.

Proof: (3-18) implies (3-17). The converse is not true as the maximizing $X_1(\tau)$ in (3-17) may vary with x_2.

Thus we have established two pruning rules for the x_1 trajectories. Both require functional dominance between two scaled versions of $s(x_2|X_1,Y_2)$ to hold for a trajectory X_1 to be eliminated. Both are weaker than the optimal pruning rules on $\underline{X}_1 \times \underline{X}_2$ implied by the Viterbi algorithm, as the latter are pointwise dominance relations. Thus the strength of the pruning technique has been sacrificed; this can only be advantageous if either $p(X_1|Y_1)$ or $s(x_1|X_1,Y_2)$ has a particularly convenient form compared to $s(x_1,x_2|Y_1,Y_2)$. We will see that this is the case for hybrid state models.

IV. · LINEAR-GAUSSIAN CASE

Before moving to the hybrid case, the relation between linear filtering and smoothing algorithms and the quantities introduced above need to be established. While the linear case exhibits no special solution structures as a result of assumptions 1b and 2b, the development here is necessary for section V.

A. Filtering

The solution to the joint filtering problem in $\underline{X}$ is well known for the linear-Gaussian case: the Kalman filter [8]. The statistics

$$x(t)=E\{x(t)|Y(t)\} \quad \underline{p}(t)=\mathrm{cov}\{x(t)Y(t)\}$$

may be recursively computed as

$$x(t)=\underline{A}\ x(t-1)+\underline{K}(t)(y(t)-\underline{C}\ \underline{A}\ x(t-1))$$

$$\underline{P}(t)=[I-K(t)\underline{C}][\underline{A}P(t-1)\underline{A}^T+Q]$$

$$[\underline{I}-\underline{K}(t)\underline{C}]^T+\underline{K}(t)\underline{R}\ \underline{K}^T(t)$$

$$\underline{K}(t)=\underline{P}(t-1)\underline{C}^T[\underline{C}P(t-1)\underline{C}^T+R]^{-1} \quad .$$

While assumptions 1b and 2b imply a block triangular or diagonal form in $\underline{A}$, $\underline{C}$, $\underline{R}$ and $\underline{Q}$, this is not reflected in the propagation of $\underline{P}(t)$, and hence in the structure of the algorithm. The reason for a lack of separation is (4-4); the update gains are not block triangular as both $y_1(t)$ and $y_2(t)$ convey information about both components of the state, just as in section IIIA. Thus the linear-Gaussian assumption does not allow extra structure to become apparent.

B. Smoothing

We will consider only the compact smoothing problem here, as the set $\underline{X}_1^t$ is an entire $N_1 t$ dimensional vector space which cannot be profitably dealt with on a pointwise basis. Thus we will specialize Lemma 1 to this case.

Theorem 2: Under assumptions 1L and 2L, and with $\underline{A}$, $\underline{C}$, $\underline{Q}$ and $\underline{R}$ the matrices which can be partitioned to provide A_{11}, A_{21}, etc. $s(\vec{x}(t)\ Y(t))$ is of the general form

$$s_o(t)e^{\frac{1}{2}(\vec{x}(t)-\hat{\vec{x}}(t))^T\underline{P}^{-1}(t)(\vec{x}(t)-\hat{\vec{x}}(t))}$$

with the parameters $\hat{\vec{x}}$ and $\underline{P}$ computable as above and with $s_o(t)$ given by:

$$s_o(t)=(2\pi)^{-N/2}(2\ )^{-M/2}\det(\underline{Q})^{-1/2}(\underline{R})^{-1/2}s_o(t-1).$$

$$e^{-\frac{1}{2}(\vec{y}(t)-\underline{C}\ \underline{A}\hat{\vec{x}}(t-1))^T\underline{S}^{-1}(t-1)(\vec{Y}(t)-\underline{C}\ \underline{A}\hat{\vec{x}}(t-1))}$$

$$\underline{S}(t)=\underline{C}[\underline{A}\ \underline{P}(t-1)\underline{A}^T+Q]\underline{C}^T+\underline{R} \quad .$$

Proof: See [11].

V. HYBRID CASE

Now we turn to the hybrid system case, given by assumptions 1H and 2H. The set of supremal trajectories X is discrete, so they can be viewed as being arranged in a tree as in Section III. The conditional survivor function $S(x_2|X_1,Y_2)$ will be that of a particular linear-Gaussian system with time-varying dynamics specified by X, so the results of theorem 2 translate to it. Thus the smoothing solution takes the form of a bank of Kalman filters, one for each X_1, with some supremal logic which prunes elements of X_1 using the tests of theorem 1. While this scheme is dominated by the combinatorial size of X_1, we will see that this same structure dominates both the filtering solution and the straightforward Viterbi algorithm for hybrid systems. Only the expanded smoothing approach of section IIIC allows any practical reduction in the size of X_1 on-line.

A. Filtering

The filtering problem for a hybrid system was first addressed many years ago [10]. The exact solution is found from a decomposition.

$$p(x_1(t),x_2(t)|Y_1(t),Y_2(t)) =$$

$$= \frac{1}{p(Y_1(t),Y_2(t))}{}_{X_1(t-1)} p(y_1(t)|X_1(t))p(X_1(t)).$$

$$p(Y_2(t)|X_1(t))p(x_2(t)|X_1(t),Y_2(t)).$$

Unlike the general case, and the linear case, the structure of the optimal state estimator forces one to consider expansions over $X_1(t)$. This is because the conditional distribution $p(x_2(t)|X_1(t),Y_2(t))$ is conveniently parametrized by its mean and covariance, but sums of such distributions can only be expressed in terms of the parameters of the components.

B. Smoothing

The smoothing problem has a structure wherein pruning is a natural operation. While the ideal smoother requires a survivor function which has many components to it, each being a weighted Gaussian shape, the combination of components is by a max operator, rather than a sum. Thus some components may in fact be completely dominated by others, and dropped without affecting the section of the trajectory estimate. This is the idea behind optimal pruning of X_1 trajectories.

Consider (3-3) and (3-12):

$$\max_{X_1,X_2} p(X_1,X_2|Y_1,Y_2 = \frac{}{p(Y_1,Y_2)} \max_{X_1}\{p(Y_1|X_1)p(X_1)\cdot$$
$$\max_{x_2}\{s(x_2|X_1,T_2)\}\}$$

From Lemma 2 and theorem 2, $s(x_2(t)|X_1(t),Y_2(t))$ will have a weighted Gaussian shape; hence the outer maximization is over a set of Gaussian shapes weighted by both supremal and infemal components. It is conceivable that some terms in this set may be eliminated by the criterion stated in Theorem 1.

First establish:

Lemma 4: In the hybrid case, $s(x_2|X_1,Y_2)$ may be computed with:

a) predict:

$$\hat{\vec{x}}(t+1|t) = \underline{A}(x_1(t))\hat{\vec{x}}(t|t)$$

$$\underline{P}(t+1|t)=\underline{A}(x_1(t))\underline{P}(t\ t)\underline{A}^T(x_1(t))+\underline{Q}(x_1(t))$$

$$s_o(t+1|t)=(2\pi)^{-N_2/2}\det(\underline{Q}(x_1(t)))^{-1/2}s_o(t|t)$$

b) update

$$\hat{\vec{x}}(t+1|t+1)=\hat{\vec{x}}(t+1|t)+\underline{K}(t+1)$$
$$[\vec{y}(t+1)-\underline{C}(x_1(t+1))\hat{\vec{x}}(t+1|t)]$$

$$\underline{P}(t+1|t+1)=[\underline{I}-\underline{K}(t+1)\underline{C}(x_1(t+1))]$$
$$\underline{P}(t+1|t)[\underline{I}-\underline{K}(t+1)\underline{C}(x_1(t+1))] +$$
$$+ K(t+1)\underline{R}(x_1(t+1))\ K^T(t+1)$$

$$s_o(t+1|t+1)=(2\pi)^{-M_2/2}\det\underline{R}(x_1(t+1))^{-1/2}s_o(t+1|t)$$
$$e^{-\frac{1}{2}[\vec{y}(t+1)-\underline{C}x_1(t+1))\hat{\vec{x}}(t+1|t)]^T\underline{S}^{-1}(t+1)[\vec{y}(t+1)-\underline{C}(x_1(t+1))\hat{\vec{x}}(t+1|t)]}$$

$$S_o(t+1)=\underline{C}(x_1(t+1))\underline{P}(t+1|t)\underline{C}^T(x_1(t+1)) +$$
$$\underline{R}(x_1(t+1))$$

where

$$\hat{\vec{x}}(t|\tau) = \hat{\vec{x}}(t|X_1(\tau),Y_2(\tau))$$

etc.

Proof: Apply Theorem 2 to the recursion of Lemma 2, conditioning on X_1. This structure of $s(x_2|Y_1,Y_2)$ indicates that a strict Viterbi algorithm on X necessarily involves a parametrization which is based on trajectories X_1. Thus the compact smoothing algorithm of section IIIB is no simpler than the expanded of IIIC in this hybrid state case.

Definition: A quality function $q(x_2|X_1,Y_1,Y_2)$

$$q(x_2,X_1|Y_1,Y_2)=p(Y_1|X_1)p(X_1)s(x_2|X_1,Y_2)$$

In the hybrid case, $q(x_2X_1|Y_1,Y_2)$ is a scaled Gaussian with mode and quadratic weights given by Lemma 4, and with scale factor

$$q_o(X_1|Y_1,Y_2)=s_o(X_1,Y_2)p(Y_1|X_1)p(X_1)$$

where the latter two terms may be recursively computed via Lemma 2.

The crux of the expanded smoothing algorithm in the hybrid case is:

Theorem 3: A supremal trajectory $\tilde{X}_1(\tau)$ will never be a subsequence of an optimal trajectory estimate $(X_1^*(t),X_2^*(t))$, $t \geq \tau$, if there exists another $X_1(\tau) \neq \tilde{X}_1(\tau)$ which is a source of $x_1(\tau)$ and for which

$$q(\vec{x}_2(\tau), X_1(\tau) \mid Y_1(\tau), Y_2(\rightarrow) \leq q(\vec{x}_2(\tau), X_1(\tau) \mid Y_1(\tau), Y_2(\tau))$$

for all values of $\vec{x}_2(\tau)$. This inequality holds iff

$$\tilde{P}^{-1}(\tau|\tau) - P^{-1}(\tau|\tau) \geq 0 \qquad \text{(a)}$$

$$\frac{1}{2}(\tilde{\vec{x}}_2(\tau|\tau) - \vec{x}_2(\tau|\tau))^T [\tilde{P}(\tau|\tau) - P(\tau|\tau)]^{-1} (\vec{x}_2(\tau|\tau) - x_2(\tau|\tau)) \geq \ell n\ q_o(\tilde{X}_1(\tau) \mid Y_1(\tau), Y_2(\tau)) - \ell n\ q_o(X_1(\tau) \mid Y_1(\tau), Y_2(\tau)) \qquad \text{(b)}$$

Proof: (5-13) is a restatement of corollary 1a. The equivalence is shown in [11].

The interpretation of these conditions is interesting. Figure 1a illustrates a case where the $\tilde{q}$ associated with $\tilde{X}_1$ allows it to be eliminated in favor of X_1. (5-14) requires that the conditional Fisher information matrix of a pruned trajectory be greater than that of the one that dominates it; Figure 1b shows that violation of this inequality will lead to $\tilde{q}$ dominating q on the tails of the distributions. Thus trajectories with good conditional information may be eliminated in favor of those with poorer information, but not vice-versa; this imparts a natural conservatism to the pruning. For cases which satisfy (a) and for a given $\vec{x}_2$, (b) determines an ellipsoidal region wherein x_2 may lead to pruning $\vec{X}_2$. Note that (a) ensures that the left hand side of (b) will always be nonpositive, hence if

$$\frac{q_o(\tilde{X}_1\ Y_1, Y_2)}{q_o(X_1\ Y_1, Y_2)} > 1 \qquad \text{(c)}$$

then this ellipsoid will be empty. (Fig. 1c). (Note that (c) can be interpreted as a likelihood ratio test on the hypotheses that $\tilde{X}_1$ or X_1 is the true trajectory). Even if (c) is satisfied, if the offset between the conditional means is too large, no elimination can take place (Figure 1d).

Since theorem 3 is based on corollary 1a, it is not as complete as possible. There may be cases where $\tilde{X}_1$ is dominated by neither X_1 nor X_1' alone, but is dominated by the max of their respective q functions (Figure 2) (provided $x_1(\tau) = x_1'(\tau)$). While the general inequality of theorem 1 may be applied: prune $\tilde{X}_1$ if

$$x_2\ q(x_2\ \tilde{X}_1, Y_1, Y_2) \leq \max_{X_1 \in \text{sources}(x_1)} q(x_2 | X_1, Y_1, Y_2)$$

the reduction of this test to simple algebraic tests such as those above is rather cumbersome.

Thus the hierarchical structure of the hybrid state dynamics, coupled with the simple parameterization of the conditional survivor function, leads to a hierarchically structured algorithm for the smoothing problem. The infemal level consists of a Kalman filter computing the mode and quadratic spread of the survivor function, and a scale factor calculation based on the Kalman filter residuals and applicable noise covariances. The supremal logic computes conditional probabilities on X_1 based on Y_1, and then prunes away some possibilities based on a Viterbi-like criterion posed in terms of functional, rather than pointwise, dominance.

VII. CONCLUSIONS

In conclusion, this work has presented a new perspective on filtering and smoothing for hierarchical Markov processes, particularly hybrid state systems. The results fall into two categories. The negative results are that the hierarchical structure does not contribute to simplification of the solution to the <u>state</u> estimation problem, nor to the <u>trajectory</u> estimation problem for discrete state, or linear-Gaussian, problems. The positive results are related to the hybrid case, where both state and trajectory estimation are dominated by a structure involving combinations of weighted Gaussian terms. While both can then be realized by separate computations of the weights and parameters of the Gaussian shapes, only the smoothing problem affords us the opportunity to eliminate some of the components entirely. This simplification of the combinatorial aspect of the problem suggests adoption of the trajectory estimation approach to hybrid systems, particularly in light of the relationship between the parameters of the Gaussian components in the two cases; they are computed by the same Kalman filters.

The results of the adoption of the trajectory estimation viewpoint is a pruning rule which is optimal in a well defined sense: the elimination of a trajectory is guaranteed to never increase the probability of error in estimating the discrete state trajectory. An example showed that this rule alone can be effective, but that some other selection mechanism is required in order to bound the number of survivors at a finite level.

Computationally, the structure of the algorithm described in Section V is ideal for VLSI implementation. The infemal calculations, involving Kalman filters and residuals computations, are completely separate from one another and would benefit from parallel execution. The interconnection between them is provided by the (simple) supremal computation involving the discrete observation, and the pruning mechanism. The latter involves simple exchange and tests of the results of the separate infemal calculations, and thus is a relatively loosely coupled mechanism.

REFERENCES

1. A.S. Willsky, "A Survey of Design Methods for Failure Detection in Dynamic Systems," Automatica, Vol. 12, pp. 6-1-611, Pergammon Press, 1976.
2. Y. Bar-Shalom, "Tracking Methods in a Multitarget Environment," IEEE Trans. AC, Aug. 1978, pp. 618-626.
3. D.G. Laniotis, "Joint Detection, Estimation, and System Identification," Information and Control, Vol. 19, pp. 75-92, Aug. 1971.
4. R.R. Tenney, R.S. Hebbert and N.R. Sandell, Jr., "A Tracking Filter for Maneuvering Sources," IEEE Trans. A.C., April 1977, pp. 246-251.
5. H.N. Psaraftis, A.N. Perakis, and P.N. Mikhalevsky, "New Models on the Ocean Acoustic Detection Process," J. Accoust. Soc. Am., Vol. 69, No. 6, June 1981.
6. F.B. Bruneau, "State Estimation of a Hybrid Markov Process with Application to Multitarget Tracking," LIDS-TH-1172, M.I.T., Cambridge, MA. June 1982.
7. G.D. Forney, "The Viterbi Algorithm," IEEE Proceedings, March 1973.
8. A.Gelb, Ed., Applied Optimal Estimation, M.I.T. Press, Cambridge, MA., 1974.
9. B.C. Levy, D.A. Castanon, G.C. Verghese, and A.S. Willsky, "A Scattering Framework for Decentralized Estimation Problems," Proc. 21st. CDC, San Diego, Dec. 1981.
10. P.J. Buxbaum and R.A. Haddad, "Recursive Optimal Estimation for a Class of Non-Gaussian Processes," Proc. Symp. on Comp. Proc. in Comm., Polytechnic Institute of Brooklyn, 1969.
11. R.R. Tenney, "Optimal Smoothing and Estimation for Hybrid State Processes," LIDS-P- , Oct. 1982.

SPREAD SPECTRUM MULTIPLE ACCESS ISSUES IN THE HF INTRA TASK FORCE COMMUNICATION NETWORK

Jeffrey E. Wieselthier

Naval Research Laboratory
Washington, D.C. 20375

Abstract. Issues raised by the use of frequency hopping (FH) spread spectrum multiple access signaling in the HF Intra Task Force (ITF) Communication Network are discussed. Of particular importance is the question of the number of asynchronous FH signals that can share a wideband channel using code division multiple access (CDMA) techniques while maintaining acceptable performance. The answer to this question depends on the modulation/coding scheme, channel characteristics, and receiver implementation. We present a new signaling scheme that, for the case of a noiseless channel, provides considerable improvement in channel throughput.

INTRODUCTION

A High Frequency (HF) Intra Task Force (ITF) Network is being designed as a robust, survivable, anti-jam (AJ) communication network for the interconnection of mobile task force elements. This network will consist of various platforms (including ships, aircraft, and submarines) with markedly different characteristics, and is expected to support the traffic requirements of many diverse scenarios. The use of the HF groundwave medium (2 - 30 MHz) is dictated by its Extended Line of Sight (ELOS) communication range as well as by its natural survivability properties in post nuclear detonation environments.

The proposed network organization, which we call "Linked Clusters," has a hybrid structure that mixes distributed and centralized control by grouping together sets of platforms that are within one-hop distance from a central platform into a centrally controlled cluster, and allowing distributed operation among "cluster heads." These clusters are established by a fully distributed algorithm [1,2].

The HF ITF Network will use spread spectrum signaling techniques in order to provide protection from jamming and interception of messages. The use of spread spectrum signaling leads naturally to the use of Code Division Multiple Access (CDMA) techniques, which can be used to provide both multiple access capability and jamming resistance. The use of spread spectrum signaling has had considerable impact on the HF ITF Network, not only from the standpoint of waveform design but also in regard to network organization and control. The primary purpose of this paper is to examine the major issues raised by the use of spread spectrum multiple access signaling in this network, as well as to present some new and preliminary results related to the channel throughput achievable using such signaling.

We begin this paper with a brief discussion of HF ITF Network operational requirements and constraints as well as a description of the Linked Cluster Architecture. We then discuss the need to use spread spectrum signaling; frequency hopping (FH) has been chosen as the spectrum spreading mechanism [3]. Finally, we discuss the issues that arise from the use of FH-CDMA signaling in our network, with emphasis on the question of the number of FH signals that can share a wideband channel, while maintaining acceptable performance levels.

HF ITF NETWORK BACKGROUND

We now review some of the major operational requirements and environmental constraints imposed upon the HF ITF Network in its role as the primary ELOS communication system in the intra task force environment. These topics are discussed in greater detail in [4].

A task force consists of up to one hundred mobile platforms that travel together as a unit, usually located within a circle 500 km in diameter. The network is characterized by a variable topology that results from changes in the HF radio connectivity of the plaforms; these connectivities change as a result of varying radio wave propagation conditions, noise levels, hostile jamming, interference from other members of the task force and other sources, platform destruction, and platform mobility.

The network must handle both voice and data traffic of several priorities with varying security requirements at acceptable error rates. Both point-to-point and broadcast modes of operation must be supported. An internetting capability with other military communication networks must be developed.

Among the most important requirements of the HF ITF Network are survivability and robustness; the network must degrade gracefully under stress conditions. Network degradation will usually be caused by loss of nodes or degradation (or loss) of links. The major threat to network links is hostile jamming; however, link quality is also affected by changing propagation conditions, platform mobility, and other-user interference. Furthermore, the increased traffic requirements during periods of crisis can cause network overloads that result in degraded performance.

THE LINKED CLUSTER ARCHITECTURE

The issue of survivability has had a profound influence on the network design. The use of decentralized network control reduces the vulnerability associated with a single central controller. Furthermore, the proposed architecture is based on the use of fully distributed algorithms that enable the task force platforms to self-organize into a reliable network structure and to continually monitor the changing connectivities for the maintenance of such a structure. A disadvantage of a completely distributed control structure, however, is that significant communication resources are needed to maintain consistent data bases at each platform; this problem is especially significant in the HF band where data rates are often limited to at most 2400 bps. We have therefore proposed a hybrid structure known as the Linked Cluster Architecture. This structure, illustrated in Fig. 1, consists of clusters of platforms within communication range of local controllers known as cluster heads. The architectural profile of the network at any given moment consists of clusters that are linked to each other via gateways. We note that when used with typical shipboard antennas HF is a broadcast (i.e., nominally omnidirectional rather than highly directional as is UHF) medium, and therefore all platforms within communication range can simultaneously monitor the transmissions of e.g., their cluster head.

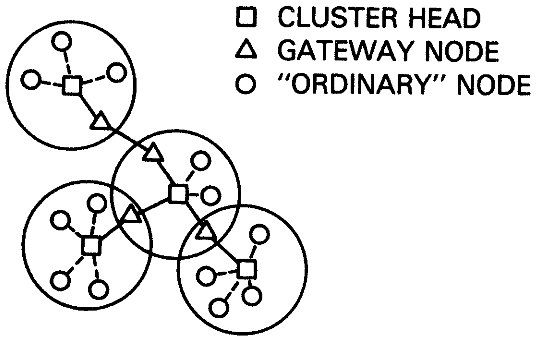


Fig. 1 Example of Linked Cluster Architecture.

The network organization algorithms make use of connectivity information exchanged between neighboring platforms. The algorithms consist of two TDMA frames, after which each platform is able to determine whether it should assume the role of cluster head or gateway, or to remain an ordinary node [1,2]. The algorithms may be executed periodically in order to form new network structures that are based on current connectivities. We emphasize that although a hierarchical network structure is implemented, the network organization algorithms themselves are in fact totally distributed.

The HF groundwave medium is characterized by a communication range that varies as a function of frequency; typically, communication range decreases as frequency increases. The Linked Cluster Architecture takes advantage of this apparent shortcoming of the HF medium by partitioning the HF band into a number of sub-bands, each with a bandwidth of a few MHz over which the groundwave communication range is approximately constant. The organization algorithm is run consecutively for each sub-band, thus producing a set of overlaid connectivity maps that give rise to a set of simultaneously operating networks. The HF ITF Network consists of this set of individual networks that are defined in separate sub-bands. Network management schemes are presently under development to coordinate the operation of each of the individual networks into an effective overall ITF Network structure. We note that the network organizational algorithms are run independently in each sub-band. Thus, while one of the Linked Cluster Networks is reorganizing, the others maintain communication using their most recently derived network structure. This capability of preserving a communication structure during the reorganization process adds considerably to the robustness of the HF ITF Network.

SPREAD SPECTRUM SIGNALING: FREQUENCY HOPPING CODE DIVISION MULTIPLE ACCESS (FH-CDMA)

A networking structure inherently provides resistance against jamming by providing relays as needed along communication paths. The use of one or more relays located between the source and destination platforms results in shorter links, and therefore improved signal to interference levels. Furthermore, an adaptive routing capability can often permit networks to send traffic along paths that avoid the most heavily jammed portions of the network.

In particular, the HF ITF Network will reorganize itself in response to hostile jamming, designating a new set of clusters and routes when necessary [5]. The ability to maintain a sufficiently high degree of connectivity is of course limited by the AJ performance of the individual network links. Only limited AJ performance can be obtained with narrowband signaling, and thus we have proposed spread spectrum signaling for the HF ITF Network. In [3] we concluded that noncoherent frequency hopping (FH) with frequency shift keying (FSK) is the most practical spreading/modulation choice for use in this network. In addition, we presented results on the AJ performance of such signaling with convolutional coding and diversity.

In FH systems the transmitter hops from one frequency slot to another, transmitting a narrowband signal at each hop. Selective addressing is achieved by the transmitter via the use of the intended receiver's unique FH pattern, and interception is difficult without knowledge of the exact FH pattern. Resistance against jamming is achieved because the jammer cannot put large amounts of energy throughout the entire frequency band, and because it cannot predict the FH pattern. In order to provide a sufficient degree of AJ and LPI capability FH systems require a wide bandwidth channel over which to hop, perhaps the entire bandwidth of one of the Linked Cluster networks, i.e., about 2 - 5 MHz. A FH system will be able to tolerate the loss of a number of hops (which would correspond to fractions of packets) caused by jamming or any other disturbance if appropriate forward error correction coding and diversity are used.

In a FH-CDMA system the code corresponds to the FH pattern. A wideband channel can be shared among a number of simultaneous users, each of which employs a different code. In such systems it is possible for two or more users (using different hopping patterns) to transmit simultaneously at the same frequency. The loss of a number of hops caused by such frequency "hits" (i.e., collisions of fractions of packets) can again be tolerated if appropriate coding techniques are used.

In this paper we address the major issues associated with FH-CDMA systems and their networking implications, and in particular the implications of using FH-CDMA in conjunction with the Linked Cluster Architecture. The major problem areas associated with the use of FH-CDMA are:

1) Synchronization and hopping rate considerations.
2) The generation of FH patterns.
3) Interference in multiple user FH systems.
4) Multiple access control schemes for FH channels.
5) The assignment and distribution of codes.
6) Contention among signals using the same code.

We now summarize some of the major issues associated with these areas, with particular emphasis on item 3, including some new and preliminary results. A more detailed discussion of these topics is presented in [3].

1) Synchronization and Hopping Rate Considerations

Other-user (i.e., other task force member) interference can be avoided if the users are coordinated so that at most one platform transmits in any frequency slot at any given time. It is in principle possible to achieve such an orthogonality in FH systems by coordinating the FH patterns of all users so that no two are scheduled to transmit simultaneously in the same frequency slot. In practice, however, the degree of synchronization required to achieve such coordination is generally not feasible at high hopping rates because of timing uncertainties that result from propagation delays as well as the inability to maintain a perfect timing reference at each platform. Therefore, the best that one can normally hope to obtain is a quasi-orthogonality of hopping patterns (and therefore codes) rather than a true orthogonality.

Although network-wide synchronization at the hop level is not feasible, such synchronization at the packet level can and should be maintained. The guard time requirement to achieve such synchronization is of the order of 3 ms in the HF ITF environment [3]. This is a small fraction of the anticipated packet length (tens of ms) and therefore represents a small degree of added overhead. While synchronization at the packet level does not reduce the occurrence of frequency hits, it does, however, facilitate the use of network management schemes that depend on the allocation of network resources on a time division basis.

2) The Generation of FH Patterns

The methods used to generate secure FH patterns are of course classified, and the network designer will have little or no control over their choice. It is therefore not realistic to expect the availability of an orthogonal set of hopping patterns even if timing uncertainties could be resolved. The most important property of a FH pattern in a secure military application is that it appears to be random; knowledge of the frequency at any set of hops should provide no information as to the future hopping sequence. We assume that random FH patterns are being used, and that network-wide synchronization at the hop level is not possible. We also assume that a very large family of such FH patterns exists, so that several distinct hopping patterns may be assigned to each platform. The intended receiver must of course know the hopping pattern of the transmitter, and must be able to synchronize to it.

3) Interference in Multiple User FH Systems

The most significant difference between FH and narrowband signaling is that in the FH case there is some degree of contention even when dedicated links are used. In the case of dedicated links only a single transmitter attempts to communicate with any particular receiver at any given time, and so there is no contention for access to the intended receiver. However, as a result of the lack of orthogonality among FH patterns, frequency hits may occur because of interference from other signals (intended for other receivers) that share the same wideband channel. The question therefore arises of how many FH signals (that use random independently generated hopping patterns) can simultaneously share the same wideband channel without resulting in significant performance degradation. There is no easy answer to this question; we must consider the modulation/coding scheme, channel characteristics, and receiver implementation. We note that the overall channel capacity would be higher if the channel were divided (e.g., via FDMA) among a number narrowband signals; however, the need for protection from jamming necessitates the use of spread spectrum signaling, as discussed earlier. After a review of the model of [6] we present a new multi-user channel model that permits considerable performance improvement for the case of noiseless channels.

Binary Signaling

We consider a slow FH system (i.e., several bits are transmitted per hop) in which Reed-Solomon coding of rate approximately 1/2 is used to correct the burst errors caused by frequency hits. (Note, however, that no decision has as yet been made as to whether block or convolutional coding will actually be used in the HF ITF Network). Fig. 2 illustrates the probability of packet error as a function of the number of users that are transmitting simultaneously over a channel with 100 frequency slots. The channel is assumed to be noiseless in this case, and so the only source of errors is other-user interference, although a noisy channel model can also be considered. We consider two packet sizes chosen so that the data of a packet can be encoded as a single Reed-Solomon code word. The RS-(31,15) code corresponds to a packet length of 155 bits; the packet is divided into 31 five bit bytes, one of which is transmitted at each hop. This code is capable of correcting up to eight byte errors per code word (packet). The RS-(255,127) code corresponds to a packet length of 2040 bits, which are divided into 255 eight bit bytes. This code is capable of correcting up to 64 byte errors. Virtually all such packet errors are detectable, often permitting the subsequent retransmission of packets that are received with uncorrectable errors. The probability of an undetected packet error is less than 2×10^{-5} for the RS-(31,15) code and less than 10^{-89} for the RS-(255,127) code [7].

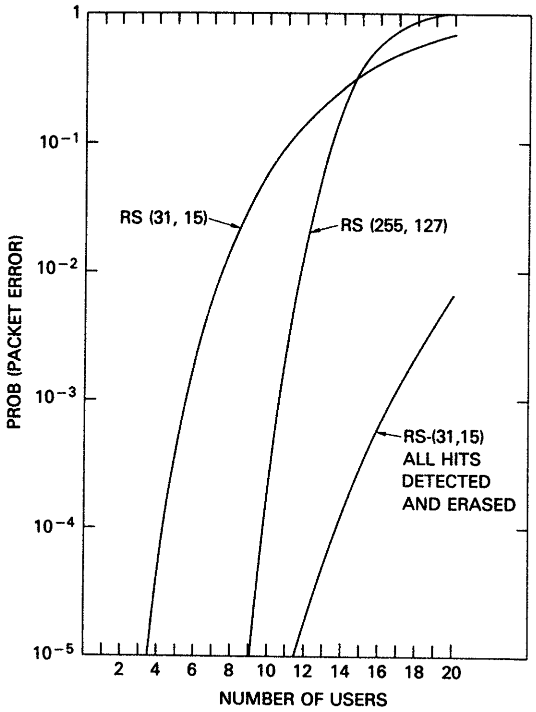


Fig. 2 Packet error probability for a noiseless asynchronous multiple user FH channel; rate 1/2 RS coding; 100 frequency slots.

Three curves are shown in Fig. 2. The two upper curves were generated under the assumption that frequency hits are not detectable, and that they all result in byte errors. This is a pessimistic assumption, because relatively strong signals will certainly have lower error probability. The bottom curve was generated under the assumption that frequency hits are detectable, and that the corresponding bytes can be erased. The packet error probability using the RS-(255,127) code with detectable hits is extremely low, and falls below the range of the plots. The ability to detect frequency hits and erase the corresponding bytes therefore results in a considerable increase in the number of simultaneous users that the FH channel can support. The tradeoffs between the increased complexity of such a system and its benefits must be addressed in the future.

The throughput of a FH channel of this type can be defined as the expected number of correct packets that can be delivered per time slot, where a time slot is equal to a packet duration (and is therefore different for the different codes and packet sizes used). Throughput is thus simply the product of the number of users and the probability of correct packet delivery for any given user. Under this model packets that are incorrectly received are subsequently retransmitted. We actually consider the throughput per frequency slot, expressed in terms of packets/time slot per frequency slot. Channel throughput for the RS codes discussed above is shown in Fig. 3 for the noiseless channel, as well as for a channel in which the noise-induced byte error probability (in the absence of other-user interference) is 0.1. Also shown in Fig. 3 is the noiseless case in which hits are recognized and erased.

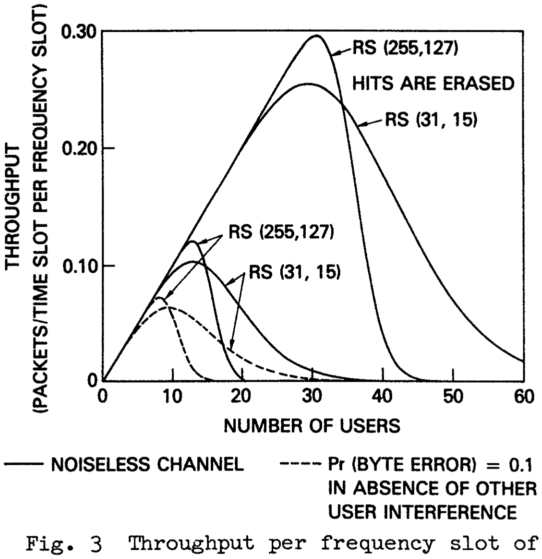


Fig. 3 Throughput per frequency slot of a multiple user FH channel; rate 1/2 RS coding; 100 frequency slots.

We have illustrated the tremendous improvement (more than a doubling of throughput) that can be obtained if hits are detected and the resulting bytes erased. The detection of hits would be straightforward in applications involving noiseless channels. In the case of binary FSK the presence of energy in both the mark and space tone positions (impossible for a valid signal) would indicate a hit. Future studies will address a more realistic channel model that considers the effects of channel noise and fading.

M-ary Signaling: A New Model

The benefits obtained through the detection of hits and the subsequent erasure of affected bytes raise the question of whether further improvement is possible. If one could not only detect hits, but also determine which of the received energy belongs to the desired signal and which to the interfering signal(s), he would then be able to make a correct byte decision thereby eliminating the need for an erasure. Unfortunately, it is in general not possible to differentiate between energy received from one source and that received from another, except perhaps in some cases where their signal amplitudes are markedly different. We shall demonstrate, however, that the use of M-ary (rather than binary) FSK signaling permits the implementation of such a scheme, at least for the case of a noiseless channel.

We assume noncoherent M-ary FSK signaling with one M-ary tone transmitted per hop. Each frequency slot thus contains M tone positions. Each M-ary symbol represents K bits, where $M = 2^K$, and therefore may be treated as a K bit Reed-Solomon symbol; e.g., a RS-(31,15) codeword (corresponding to one of our packets) would consist of 31 32-ary symbols. It is assumed that each receiver is synchronized to its desired signal, but that no synchronization at the hop level can be maintained among the signals that simultaneously share the wideband channel. Random FH patterns are again assumed.

As in the binary case, a frequency hit occurs when one or more other users transmit in the same frequency slot as the desired signal. Since a valid signal can consist of only a single tone, the presence of two or more tones in any hop duration indicates that something is wrong (i.e., a hit has occurred). As in the binary case such bytes can be erased.

We claim, however, that it is often possible in the M-ary signaling, noiseless channel case to discriminate between the desired signal and the undesired signals. If perfect synchronization is maintained with the desired signal's hopping pattern it would be possible to discriminate against tones that are present for less than a certain fraction, which we denote by ρ, of the hop duration. There is no constraint on the amplitude of the interfering signals.

Fig. 4 illustrates 8-ary FSK signaling and several types of frequency hits, denoted a-d. The effects of these hits are as follows:

a) A hit in the same tone position as the desired signal does not bother us.

b) A single hit in a tone position different from the desired signal will bother us only if the overlap is greater than ρ.

c) Hits from two or more other users in the same tone position (but different from the desired signal) at opposite ends of the hop will bother us only if their combined overlap is greater than ρ.

d) Hits in different tone positions, each with overlap less than ρ, do not bother us even if their combined overlap is greater than ρ.

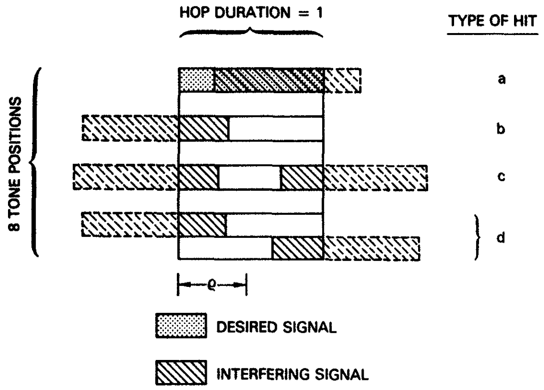


Fig. 4 8-ary FSK example illustrating four types of frequency hits.

A more detailed description of this model, including a derivation of system performance as well as performance curves for several M-ary alphabet sizes, is presented in [8].

The implementation of a system of this type would require accurate synchronization and a fairly sophisticated receiver. A possible implementation would examine the time derivative of the matched filter outputs of each of the tone positions throughout the hop duration. A time derivative value of zero would indicate the absence of a signal, independent of the total energy accumulated thus far. A signal will be declared present as long as the time derivative is sufficiently large.

The ability to ignore interfering signals that are present for less than a certain fraction of the hop duration is based on the fact that the M-ary signal is constant throughout the hop duration. In contrast, in the binary case discussed earlier several bits are transmitted serially; if one or more bits are obscured by a hit there is no way to recover the lost information, thus necessitating an erasure.

The achievable throughput per frequency slot using the model presented here is illustrated in Fig. 5 for the case of 100 frequency slots, a noiseless channel, and 32-ary signaling. RS-(31,15) coding is used; each codeword (packet) can tolerate 16 hop erasures. The case of $\rho = 0$ corresponds (almost exactly) to the curves representing detectable and erasable hits shown in Fig. 3. As ρ approaches 1 the total channel throughput approaches the number of users; the throughput per frequency slot can actually be greater than one packet per time slot! It is difficult to estimate values of ρ that may be achievable in a practical system. Realistic values would depend on hopping rates and hardware implementation.

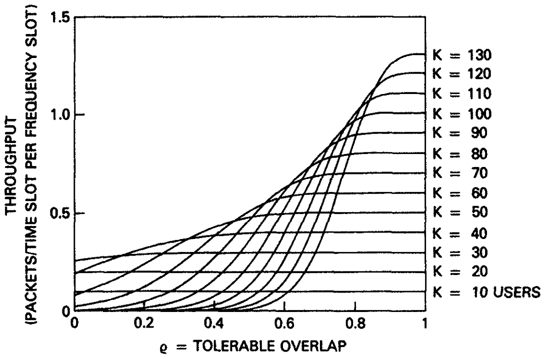


Fig. 5 Throughput per frequency slot of a noiseless multiple user FH channel in which partial hits can be ignored; RS-(31,15) coding; 100 frequency slots.

Future studies will address the relationships among M-ary alphabet size, hopping rate, frequency slot bandwidth, code rates, and achievable data rate. Furthermore, we will attempt to extend our model to more realistic channels that are affected by noise and fading.

4) Multiple Access Control Schemes for FH Channels

An area of great interest in recent years has been the development of control schemes for random access channels that ensure stability as well as satisfactory delay and throughput performance. In the past only schemes that are suitable for time-domain channels have been considered. It is not clear how to apply most of these schemes to a FH channel.

Many of the control schemes that have been proposed for time-domain multiple access channels require that the users monitor the channel to determine whether there was a single (and therfore successful) transmission, two or more transmissions (all of which are usually assumed to be unsuccessful), or no transmissions. The situation in a FH-CDMA channel is different because several signals can in fact be successfully transmitted simultaneously. Another characteristic of FH-CDMA is that it is not feasible to monitor the success or failure of each of the individual transmissions of all other users because to do so would require the use of a separate frequency hopped receiver corresponding to the FH pattern of every user in the population.

Instead of attempting to monitor individual transmissions it is better to observe the overall FH channel. During each slot every user can use a random hopping pattern to hop a receiver over the frequency band. An estimate of overall channel activity can be obtained by counting the number of dwell times that the channel is in use in the monitored frequency slot. Hajek has demonstrated that feedback information of this type can be used to implement a control policy suitable for FH random access systems [6]. If such an estimate of channel activity is not available it is necessary to use multiple access schemes that require a minimum of feedback channel information, e.g., the Acknowledgment Based Retransmission Control (ABRC) policies [9].

5) The Assignment and Distribution of Codes

The basic code assignment considerations include the allocation of frequency hopping codes to users and the decision of whether to use a code associated with the transmitter or with the receiver. The four basic types of links in the network will be point-to-point links, broadcast links, random access links, and common links.

Point-to-Point Links: When point-to-point (i.e., single source, single destination) links are used the FH code of either the transmitting or receiving platform may be used, as long as consistency is maintained.

Broadcast Links: In a broadcast link a single transmitter sends a common message to two or more platforms. It is therefore appropriate to use a FH pattern associated with the transmitting platform, and monitored by all potential receiving platforms. For example, each cluster head would use a distinct FH pattern to brodcast messages to all members of its cluster.

Random Access Links: In a random access link many platforms attempt to communicate with a single platform on a contention basis. A FH pattern associated with the receiver is therefore essential, because the receiver will not know a priori which platforms are attempting to communicate with it. For example, it may be necessary for cluster members to communicate with their head on a random access basis.

Common Links: A common link, monitored by all network platforms, can be used for either broadcast or random access applications. A single FH pattern would be associated with such a link. For example, such a common code would be used during the execution of the network organization algorithms.

Links of each of these types will share a wideband FH channel simultaneously. The relationships among these classes of links and the Linked Cluster Architecture are discussed in [2,3].

An important question is the procedure used to assign the frequency hopping patterns to the network platforms. The simplest approach is simply to associate a priori a pair of unique codes with each platform in the network. There would then be no need to develop a distributed scheme for the assignment of FH patterns to platforms, although of course the need for channel allocation schemes to share resources on a time division basis still exists. Such a scheme has in fact been developed for the HF ITF Network [10]. It is of course necessary for each platform to know the codes associated with the other network platforms. It may be possible to disseminate this information to all task force members before deployment of the task force. If not, then some mechanism will be needed to broadcast this information throughout the network; distributed techniques for secure key distribution would then have to be developed. Security will be a major issue in the distribution of FH patterns, because it is essential that they are not known by an adversary.

6) Contention Among Platforms Using the Same Code

Contention among signals that use the same code may arise when two or more platforms attempt to transmit to the same receiver simultaneously. We are assuming that no two receivers will use the same code, except to monitor a network-wide (or cluster-wide) random access channel. The simplest case to consider is that of only two signals. The two basic situations that may arise are those in which the delay at the receiver between the two signals is:

a) less than the hop duration,

or,

b) greater than the hop duration.

In case (a) there is a collision that is characteristic of time-domain systems, and typically both signals will be lost, unless one is considerably greater in amplitude than the other. In case (b) it will often be possible for the receiver to acquire and maintain synchronization with the first signal to arrive. Once a signal has captured the receiver, a delayed version of the same signal (generated by multipath propagation) or another signal that uses a delayed version of the same hopping pattern will ideally not be troublesome (although in practice synchronization can be disrupted by fading or jamming, especially when platform mobility results in varying propagation delays). Further effort is needed to evaluate the performance of contention-based FH-CDMA systems that use a common code, and to assess the implications of their use in the HF ITF Network.

CONCLUSION

The use of frequency hopping (FH) spread spectrum signaling in the HF ITF Network, necessitated by AJ considerations, permits the use of Code Division Multiple Access (CDMA) techniques. In this paper we have discussed many of the issues raised by the use of FH-CDMA. Of particular importance are the questions of the throughput achievable on wideband channels, as well as the problem of channel access schemes that are suitable for FH-CDMA signaling.

We have demonstrated that the number of FH signals that can share a wideband channel is related to the modulation/coding scheme, number of frequency slots, channel characteristics, and receiver implementation. In particular, we have presented a new scheme that greatly increases the achievable throughput for the idealized case of a noiseless channel in which the only interference is caused by other users. While the HF ITF environment is far from ideal, these studies have provided insight into desirable signaling schemes and receiver structures.

One can not design network control procedures without considering the impact of the use of FH-CDMA. Of importance are not only achievable channel throughput, but also the question of channel access control schemes that can be implemented using FH-CDMA signaling. Future studies will continue to investigate the intimate relationships between FH-CDMA and overall HF ITF Network design.

REFERENCES

1. D.J. Baker and A. Ephremides, "The Architectural Organization of a Mobile Radio Network via a Distributed Algorithm," IEEE Trans. Commun., COM-29, 1694-1701, November 1981.
2. D.J. Baker, A. Ephremides, and J.E. Wieselthier, "An Architecture for the HF Intra-Task Force (ITF) Communication Network," NRL Report 8638.
3. J.E. Wieselthier, D.J. Baker, A. Ephremides, and D.N. McGregor, "Preliminary System Concept for an HF Intra Task Force Communication Network," NRL Report 8637.
4. J.E. Wieselthier, D.J. Baker, and A. Ephremides, "Survey of Problems in the Design of an HF Intra Task Force Communication Network," NRL Report 8501, October 1981 (AD-B060-647L).
5. D.J. Baker, J.E. Wieselthier, A. Ephremides, and D.N. McGregor, "Distributed Network Reconfiguration in Response to Jamming at HF", Proc. MILCOM, October 1982.
6. B. Hajek, "Recursive Retransmission Control -- Application to a Frequency-Hopped Spread-Spectrum System," Proc. Sixteenth Ann. Conf. Inf. Sci. and Syst., Princeton University, March 1982.
7. E.R. Berlekamp, "The Technology of Error-Correcting Codes," (Appendix B), Proc. IEEE, 68, 564-592, May 1980.
8. J.E. Wieselthier and A. Ephremides, "Frequency Hopping Spread Spectrum Multiple Access with M-ary FSK Signaling," submitted for publication.
9. B. Hajek, "Acknowledgment Based Random Access Transmission Control: An Equilibrium Analysis," Proc. IEEE Int. Conf. Commun., June 1982.
10. D.J. Baker, A. Ephremides, and J. Wieselthier, "A Distributed Algorithm for Scheduling the Activation of Links in a Self-Organizing, Mobile, Radio Network," Proc. IEEE Int. Conf. Commun., June 1982.

TECCNET: A Software Testbed for Use in C3 System Research[1]

Elizabeth R. Ducot

Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, Massachusetts

Abstract. TECCNET (Testbed for Evaluating Command and Control NETworks) is a small, expandable software system created to support C3 system research. It has been designed to facilitate the modeling and analysis of the complex interactions between the distributed command and control network elements, the algorithms and procedures that characterize the information flow networks of which these elements are a part, and the environment within which they must function. The TECCNET system provides a software laboratory, with a flexible interactive structure, in which basic system research functions are performed. These functions include: the on-line description of the problem of interest and the definition of a model to be studied, the generation of an appropriate scenario, and the execution of the desired experiment. A progression of experiments using TECCNET has been planned to serve a dual purpose. For the near term, the focus will be on the areas of distributed information network management and decentralized estimation. This will allow necessary building blocks to be created, contributing toward the development of an Information Intermediary that is intended to resolve conflicts between the needs of C3 system users and the capabilities of the C3 networks.

INTRODUCTION

The need to provide dynamic limitations on the flow of information in a Command, Control, and Communications (C3) system has become increasingly apparent. Indeed, this need, coupled with the requirement that the C3 systems operate effectively under a variety of adverse conditions, has provided the motivation for much of the recent C3 system research. TECCNET (Testbed for Evaluating Command and Control NETworks) is an experimental software laboratory, designed in response to this need in a way that will support a number of complimentary research activities. Before describing the structure and characteristics of TECCNET, it is appropriate to summarize the point of view taken in creating this research support system.

In this work, the C3 system is visualized as an information flow network. This description encompasses not only the communications systems that transmit data and messages, but also the processing and storage systems that acquire, translate, manipulate, and disseminate information. The performance of this network may be described (at least conceptually) in terms of its ability to deliver, at designated points, the desired information so that upon arrival, it is timely, accurate, complete, and easy to use.

The underlying C3 system problem that motivated the design of TECCNET is extremely complex. The system elements comprising the information flow network are highly distributed, have diverse physical characteristics, and are often governed by ill-defined operational constraints and procedures. The technologies that affect the elements of this system are changing rapidly; advances in electronic weaponry, sensors, and computers, for example, combine with changes in the way information is used by the commander to increase both the flow of information in the network and the time pressures associated with its delivery.

Even under somewhat benign conditions, the task of supporting this flow of information is a formidable one. However, when the tactical situation intensifies, the load on the system increases substantially, just when the external stress on the network induced by a hostile atmosphere is at its peak. Competition for the same resources to move, process, store, and display information also intensifies -- frequently with disastrous results (i.e., excessive message delays, system and user information overloads, etc.). Thus, the C3 system, viewed in terms of how well it provides the

---

[1] This work was supported initially by the Air Force Office of Scientific Research (Contract Number AFOSR-80-0229). Support for the continued development of the system has been granted by the Office of Naval Research (Contract Number ONR/N00014-77-C-0532).

information support expected by the decision-maker, is perceived to degrade exactly when it is most important that it operate well: when battle information is flowing and the time available for decision making is short. It follows then that there is need to modify the information flow to match it in real-time to the facilities and the time available for processing.

The research problems formulated from the preceding statements share a common premise: in order to develop techniques for controlling the flow of information effectively in a C3 context, one must be able to express and exploit the relationships between the activities of the user and the conditions of the network. This premise is evident in research efforts addressing all levels of the information flow system -- the control of the underlying communication network, the way information is generated and used, and ultimately the interface between the human users and the systems (Ducot, 1980 and 1982). As a result, one of the first goals in the development of TECCNET was to encourage the integration, within a common framework, of the relevant ideas drawn from diverse research areas (i.e., distributed data base, sensor and network management, information processing and presentation, etc.).

Along with the common premise indicated above, a common concern has been expressed over the potential cost (both in terms of system and user resources) of implementing proposed information flow control schemes. Hence the second hope in creating the TECCNET system: that, through experimentation, a better understanding could be developed of the complex interactions between the distributed command and control network elements, the various models, algorithms and procedures that characterize the information flow, the needs of the users of the C3 systems, and the environment within which these systems must function.

Ultimately, however, the objective in creating TECCNET is to promote the development of new models for representing the C3 information processes and new concepts for dealing with the resulting information flow. An important goal, therefore, is to provide the type of environment that will foster a broad range of research activities and will facilitate the testing of proposed algorithms and procedures. In the next section, the design and characteristics of the TECCNET system are described briefly, followed by a discussion of the initial plans for its use as a research tool.

TECCNET: THE SYSTEM IN OUTLINE

In order that TECCNET meet the goals outlined above, a number of design objectives were established. Based on these objectives, a skeleton system was implemented at MIT beginning in 1981. Development of the system continues within this same framework:

1) The testbed should be SMALL, with a controlled plan for expansion, so that it will remain a manageable tool for project research.

2) The software should be structured so that it can operate in a multi-user environment and meet the needs of users with different levels of software and system expertise. Moreover, the system should be interactive and provide considerable on-line documentation.

3) System interface and support software should be developed to facilitate both modeling and testing activities.

4) Default models and representations of the system should be available in order to reduce the effort required to initiate simple experiments.

5) The modeling tools created for the TECCNET system should make it easy for the user to represent the asynchronous interactions and complex protocols that are characteristic of the models and algorithms to be explored.

The preceding statements encompass a broad range of system capabilities--capabilities outside those customarily associated with software for algorithmic and system research. As a result, the final design of TECCNET and the open-ended plans for its development resemble procedures for the design of computer operating systems (Corbato, Saltzer, and Clingen, 1972) as much as they do those applied to the type simulation system software originally anticipated (Oren, Shub, and Roth, 1980).

This dual nature of the software is apparent in the organization of TECCNET, depicted in Figure 1. Three basic research functions, designated 1) the Model Generator, 2) the Scenario and Input Generator, and 3) the Information Network Simulator, are supported within an interactive structure. A brief description of the funtional elements follows. The reader is referred to a discussion of the software system (Ducot, 1982) for additional detail and sample TECCNET interactions.

The Model Generator

The Model Generator is the first of TECCNET components. It permits the user to specify the modeling environment, and, in some sense, to build the simulation on-line. A view of the C3 information flow network is defined by combining models of the local processing nodes, constraints on the movement of messages, protocols governing the information flow, and the algorithms for managing the network. In general, these specifications are tightly coupled, bound together by the need for consistency in the modeling assumptions. Sets of simulation specifications, along with descriptions of any built-in assumptions, may be stored in the system as defaults -- defaults which can then be manipulated by the user.

For example, one such set (currently in place within TECCNET) permits analysis at the level of the nodes and links comprising a store and forward data communication network. In this case, the queueing model of the processing elements used is extremely simple. Processing of data packets is assumed to take "zero" time compared to packet queueing and transmission delays. The link buffers at the nodes are not modeled explicitly; their capacities are reflected in an effective link capacity that is a fraction of the physical limit of the line. Moreover, the transmission and receipt of packets are assumed to be perfect processes. Certain characteristics of the message traffic (exclusive of volume) also have been specified as defaults. For simplicity, data packets are assumed to have the same average length, and only one conversation may be active between a pair of nodes at any given time. A first-in-first-out (FIFO) service discipline is used for the treatment of data packets, with preemptive logic for both the control packets and acknowledgements having a higher priority in the system.

The structure and content of the data packets are not given for this simple default model; data comprise background traffic within the network. Control messages, on the other hand, require explicit treatment of both structure and content, as these messages are used as signals to drive the TECCNET algorithms. An initial network management algorithm (an outgrowth of an original distributed routing scheme developed by Gallager (1977)) which utilizes the preceding modeling assumptions, is included as part of this modeling environment.

A library structure has been developed to house modeling specifications and algorithmic building blocks. Descriptive information is associated with each entry in the library in order that the user may select among default models and procedures. Additional specifications can be added to modify an existing modeling environment, or to specify a new one as desired by the user.

Scenario Generator

The Input Generator is the data-intensive component of the TECCNET System in which the user defines the conditions to be simulated. For the sample view of the network indicated above, three steps are required: 1) the specification of the network topology, capacity of the links etc., 2) the association of nodes with particular processing models and descriptions of the traffic between them, and 3) the representation of the environment. As the modeling of the information flow network elements becomes more sophisticated, additional inputs representing different types of decision variables will be developed.

Inputs are solicited from the user at the terminal in free form. These data are organized into permanent files and are catalogued with descriptive comments that later may be displayed on-line. The files may be shared between users, each of whom is given a private working copy that functions as a data base during his input session. Sample sessions, indicating how the data bases are created and manipulated by TECCNET, are presented in (Ducot, 1982).

Scenario inputs (describing the condition of the information flow network) are distinguished from those that define the experiment (such as number of iterations, convergence criteria, cost function parameters, type of statistics to be collected, etc.) and are stored separately. This distinction is best appreciated by the user who attempts to combine "canned" scenarios and model specifications for use in multiple experiments. The scenario building process may occur in small segments at different TECCNET sessions until a complete scenario has been obtained and stored in the system.

Network Simulator

Once a model and scenario of interest have been established, the execution phase of the experiment can be initiated. Discrete event simulation techniques form the basis of the execution software. This permits the integration of many procedure-driven models and the represention of asynchronous operation of the elements of a distributed system.

Three types of events are modeled, designated for purposes of discussion "external", "spontaneous", and "responsive". External events are derived from the environment, and refer to situations arising outside the network. These events are not modeled within the system in detail; they are represented only as time-dependent effects applied to one or more of the capabilities of the system elements.

Spontaneous events simulate actions that are based solely on internal logic operating at the nodes of the information flow network. Thus, these internal events correspond to the decoupled actions of a cooperating member of a distributed system. These events may take many forms depending on the modeling environment that has been specified. The most straightforward spontaneous event is a scheduling event. An example, drawn from the current distributed communication algorithm, is the command from an individual node signalling the initialization of a routing/flow control update cycle that will change the flow within the network. A slightly more elaborate form of the scheduling event is a conditional one, in which some quantity (observable at the node) is monitored until a threshold is reached, at which time the event is scheduled. Internal clocks (synchronized or unsynchronized) are maintained at the nodes to determine the activation time for the spontaneous events.

Execution of a spontaneous event may initiate a sequence of responsive events. Communication with other nodes in the system is required to generate one or more of the events in the sequence. Since responsive events are triggered only by the receipt of an appropriate control message, they are used to model the various forms of <u>cooperative</u> actions among the distributed network elements.

The event generator as currently implemented is only partially interactive. The user is on-line while the simulation is running: he may view the results, request that output at various levels be displayed or suppressed, and decide whether or not to continue the experiment. In the future, the user of TECCNET will be permitted a dual role: that of researcher who is observing the experiment, on the one hand, and that of C3 information network customer who is changing inputs and requests in real time, on the other. This additional capability is reflected in the presence of an interactive node in Figure 1.

Conversational Interface

The Conversational Interface provides the link between the user and the body of the TECCNET system. Communication is interactive, with commands and responses entered and displayed at the user's terminal. The forms of the interaction can be controlled by the user, and the display level ("verbose" or "terse") may be set by him according to his familiarity with the system.

This interface software is basically command driven; a feature which gives a user considerable flexibility in his use of the system. This is a "user active" style generally preferred by the designers of interactive systems, reflecting the fact that experienced users can learn to bypass detailed explanations and move efficiently through the system. Occassionly however, when complex descriptions or order-dependent responses must be solicited from the user, the user active mode is suppressed by the Conversational Interface, and a more restrictive question and answer (or "user passive") format is employed.

Whenever the user message (****USER::) appears, TECCNET is awaiting input from the user. A specially designed command-line interpreter monitors the user's entry to distinguish the following: 1) signals for movement within TECCNET (motion commands), 2) requests for information (help commands) and 3) specific data entries (responses to system prompts and questions).

Motion commands allow the transfer between the basic user activities. For example, the command "model" places the user in a position to define his modeling environment. Unlike motion commands, help commands (which provide on-line documentation and clarification) have no positional properties and may be issued without limit at any time. When a help command is received by TECCNET, the information requested is displayed at the terminal, at which time the user may continue his session as though no interrupt had occurred. If a specific question had been asked of the user, the question is repeated at the end of the help message.

These same help messages can also be used to provide on-line training for the user. This requires that an underlying sequence for command use be evident during the interaction. TECCNET messages contain the suggestion of a next logical command at the end of each response; suggestions which may be used by the user to guide him through the system description. As an example, a partial sequence of commands and responses, used as an introduction to the system, is depicted in Figure 2.

USE OF TECCNET

The preceding presentation of the TECCNET modeling system has provided a brief indication of how the software has been structured to support C3 system research. Since the major goal in the development of TECCNET is to promote the evolution of new approaches to information flow control, the plans for using the TECCNET system as a research tool are of immediate interest. A significant payoff is anticipated if the experiments can be structured into experimental building blocks; each of which contributes a portion of the insight and experience necessary to proceed to successively higher levels of abstraction in viewing the information flow network.

One of the chief beneficiaries of such an approach is expected to be the effort to develop the Information Intermediary (Ducot, 1982) which addresses the information related interactions at the highest system level -- the interface between the user of the information and the C3 network itself. The intent of this Intermediary is to assist the human user of the C3 system; aiding him to reformulate his requests for information and to change his use of the information flow network in light of network conditions. In order to introduce notions of flow control for information (as opposed to data) into the network, the Intermediary must have access to a specially developed local status model of the system; a model that integrates dynamic network, data base, and user information and requires the flow of control information between network elements. In other words, this model must reflect the interactions between three types of information management procedures: 1)strategies that induce changes in routing and control of data flow, given network parameters, 2)criteria for modifying decisions governing the generation and injection of information into the system, given these same or related status indicators, and 3)changes in approaches to information retrieval, given the behavior of the network.

In considering candidate approaches on the basis of their compatiblilty and potential contribution to such a model, three desirable features were identified. The first is the feasibility of representing proposed technique for detecting flow conditions and for exerting necessary control, in a way that can be implemented via a distributed algorithm. The second is the ability to formulate the control actions and decisions to be exercised at the nodes based on <u>limited</u> local information. And third is the possibility of sharing common status information and network parameters among different types of management algorithms.

These characteristics were considered in determining the first step in the TECCNET utilization plan; development of a modeling baseline from which a broad class of techniques for managing the communication network could be studied. The initial algorithm included in the TECCNET system is representative of a procedure (type 1) that induces changes in data flow given network conditions. This approach (extended from the original formulation (Gallager, 1977) by Golestaani (1980)) treats flow control and routing together, leading to a flow control algorithm that is expressed in terms of the following conflicting objectives: to reduce congestion in the network while at the same time minimizing the amount of offered traffic that is rejected by that network. A convex optimization problem is formulated in which short-term average information on network utilization is used to allocate both maximum data rates for user sessions (viewed as source/destination pairs) and the optimum routes through the network for information flowing within it (Gallager and Golestaani, 1980). From the point of view of potential contributions to the model required by the Information Intermediary, the appeal of this initial approach lies in the formulation of the distributed algorithm, the type of marginal delay information communicated, and the structure of priority functions that represent the cost of rejecting flow between individual node pairs.

The second of the TECCNET building blocks presumes the existence of both the real-time status information (of the type described above) and the distributed algorithm by which it is communicated. The experiments being considered as part of this second phase (Ozbek, ongoing), represent the first attempt in the TECCNET framework to associate the criteria for generating and injecting information into the network with the network parameters themselves.

The decision variables are drawn from a formulation of a decentralized estimation problem in which explicit use is made of the fact that communication from sensors to estimators is <u>not</u> instaneous. The normal incentives to obtain high quality estimates by transmitting complete information frequently between nodes, are recognized as being far from optimal. As part of this research effort, a number of tradeoffs dealing with the generation and scheduling of information reporting can be addressed. Of immediate interest are those describing: 1) the frequency of reports (relating raw data reporting frequency, traffic volume, delay, and the use of sensor information) and 2) the quality of reports (frequent compressed reports, partially processed at intermediate nodes, versus the less frequent receipt of nearly raw data.

With the experience gained in creating these two building blocks (type 1 and type 2 procedures), it is hoped that the next stage in the TECCNET utilization plan, the incorporation of information retrieval strategies (type 3), may be initiated in the not too distant future.

CONCLUSIONS

In the preceding sections, the design, and intended use of a new research tool, created especially to support C3 system research, was presented. The potential contributions to a variety of ongoing research activities were considered in the development of the initial version of TECCNET, now operational at MIT. Preliminary experience with the TECCNET software suggests that the design objectives, outlined at the beginning of this paper, are being met. The interactive format and modular structure of the system appear appropriate to the needs of users with different levels of software and system expertise who will be participating in this activity in the future. The modeling tools incorporated in the system provide the capability for representing the asynchronous interactions and complex protocols inherent in the models and algorithms likely to be explored. Development of the system is continuing. Additional default modeling environments will be included to allow the pursuit of several lines of inquiry in parallel, each of which is expected to contribute a different perspective to the overall development of information flow control techniques. It is anticipated that extensive use of the TECCNET system will lead as a by-product to modifications and improvements in the system. As these enhancements are made, it is hoped that the scope of the information flow modeling activities will continue to broaden.

REFERENCES

Corbato, F.J., J.H. Saltzer, and C.T. Clingen, (1972) "Multics--The First Seven Years", AFIPS Conference Proceedings 40, 1972, SJCC AFIPS Press, Montvale NJ,pp.571-583.

Ducot, E.R. (1980) "Some Thoughts on Information Flow Control in C3 Systems" Volume 5, Proceedings of the Third MIT/ONR Workshop on Distributed Information and Decision Systems, LIDS-R-1024, Laboratory for Information and Decision Systems, MIT, Cambridge MA, December 1980.

Ducot, E.R. (1982) TECCNET: A Testbed for Evaluating Command and Control NETworks, LIDS-R-1227, Laboratory for Information and Decision Systems, MIT, Cambridge MA, August 1982, 63pp.

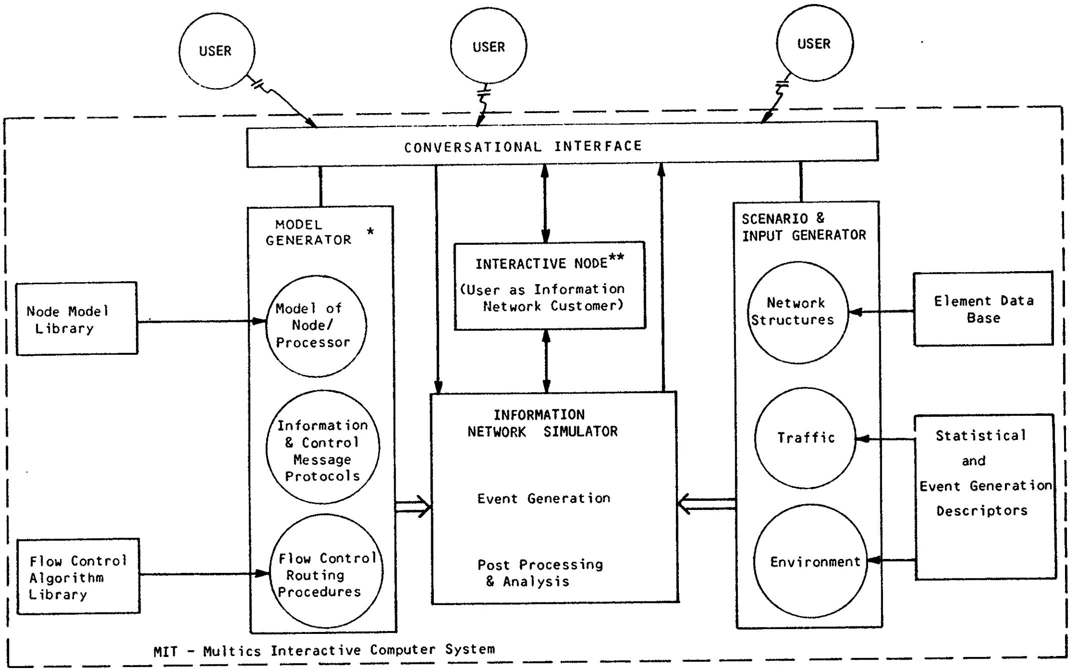
Gallager, R.G., (1977) "A Minimum Delay Routing Algorithm Using Distributed Computation", IEEE Trans. on Communications, Jan. 1977, pp.73-85.

Gallager, R.G. and S.J. Golestaani, (1980) "Flow Control and Routing Algorithms for Data Networks", Proceedings of ICCC'80, Atlanta GA, October 1980.

Golestaani, S.J. A Unified Theory of Flow Control and Routing in Data Communication Networks, PhD Thesis LIDS-TH-963, Laboratory for Information and Decision Systems, MIT, Cambridge, MA January 1980, 92 pages.

Oren, T.I., C.M. Shub , and P.F. Roth, (1980) "Simulation with Discrete Models: A State-of-the-Art View", Volume 2, Proceedings of 1980 Winter Simulation Conference, Orlando FL, Dec. 1980. 258 pages.

Ozbek, A. (ongoing) Unpublished memoranda in preparation for S.M. Thesis, Laboratory for Information and Decision Systems, MIT (scheduled for completion 1983).



*Initial version allows selection of only one process model (simple input/output store & forward node) and one flow control algorithm with baseline message set.

**Planned for future development.

Fig. 1 Structure of TECCNET

```
++++TECCNET:

    Welcome to the Information Flow Network Testbed (TECCNET).
 For information on how to use the system, type:
          help
 followed by a carriage return.  Otherwise, on receiving the
 user cue (****USER::) you may type any TECCNET command.

****USER::
help

++++TECCNET:
     TECCNET is an interactive testbed which is intended to support
 the analysis of a number of information flow related issues.  It is
 structured to provide user support in three areas:

          1) the specification of his modeling environment, selection
             of local node models, message protocols and flow control
             strategies.  (program segment:  'model')

          2) the generation of simulation input and scenarios that define
             the network / traffic conditions. (program segment: 'scenario')

          3) the on-line execution of the simulation experiment (program
             segment:      'run').

    For information on how to interact with the TECCNET system, the
 inexpert user should type:        use

****USER::
use

++++TECCNET:
    You converse with TECCNET by entering commands and responses from your
 terminal.  When the user cue (****USER::) appears, it is the indication
 that you may begin typing.  To signal the computer that you have completed
 your entry you MUST strike the carriage return (CR) key.  The computer will
 not respond until you do.
     The commands permit you to move freely within the TECCNET system,
 to exercise the various programs, or to receive explanations and
 assistance in program selection and data preparation.  Occasionally,
 a specific response is required.  In these instances, you will be
 prompted from the terminal before the user cue is given.
     Depressing the BREAK or ATTN key at any time will interrupt TECCNET
 and return you to a point where you may again enter commands.
   The inexpert user should now type:
                    commands

****USER::
commands

++++TECCNET:
    The basic commands for controlling and interacting with the TECCNET
 system are organized into three groups:

          1) movement within the system:
             model: to change modeling environment
             scenario: to create scenario data
             run:    to execute the experiment
             stop:   to leave the system
          2) on-line documentation:
             help:  brief description of TECCNET
             use:   outline of interaction modes
             help model, help scenario, help run: descriptions of
                    the TECCNET submodels
             use model,use scenario, use run: instructions for their use
             help network, help traffic: description of input parameters
          3) user support:
             help error: how to correct errors in typing
             help comment: how to leave comments about the system
             comment: comment mechanism
             terse: request for brief interactive reporting
             verbose: request for complete interactive reporting
```

Fig. 2 Introduction to TECCNET: A Tutorial

OPTIMAL FILE ALLOCATION PROBLEMS FOR DISTRIBUTED DATA BASES IN UNRELIABLE COMPUTER NETWORKS*

by

Moses Ma
Michael Athans
Laboratory for Information and Decision Systems
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

ABSTRACT

This paper deals with the problem of optimally locating files, and their optimum number of redundant copies in a vulnerable communication network. It is assumed that each node and link of the communication network can fail independently. The optimization problem maximizes the probability that a commander can access the subset of files that he needs while minimizing the network-wide costs related to storage, query and update communication costs. The problem reduces to a linear zero-one integer programming one; several theorems that reduce its complexity of solution are presented.

1.1 INTRODUCTION

This paper focuses on the problem of optimal redundant file allocation for a very vulnerable distributed data base system. This file allocation is different than previous file allocation problems because it considers the following new items:

1. Vulnerability of the nodes and links due to enemy actions, e.g. jamming
2. The importance of the users
3. The importance of particular files to particular users.

First we shall discuss the motivation for the problem of optimal file allocation in a vulnerable environment. Second we shall explain the problem and its constraints. Third we shall discuss the possible trade-off in costs. Fourth a brief literature survey will be presented. Fifth the actual formulation will be explained. Sixth the various theorems that have been developed for this formulation will be stated and explained in words; however, we do not include the theorem proofs. Seventh the conclusions and suggestions for further research will be presented.

1.2 MOTIVATION

The motivation behind our problem is in the C^3 (Command, Control, and Communications) context. In this context we are considering a Naval Battle Group (BG) composed of aircraft carriers, cruisers, destroyers, aircraft, etc. The BG gathers information through its organic sensors. This information must somehow be stored and maintained to the utmost correctness because the BG must coordinate its actions. The system may be considered as a distributed data base system. The ships and planes can be considered as nodes, and the communication channels between the ships and planes can be considered as links. The data can be considered as files stored in the computers of the ships and planes.

Considering the BG as a set of nodes, links and data files, we have defined for ourselves a distributed data base network. Since in warfare, ships can be destroyed and communication links jammed, our network is vulnerable. Therefore we must consider how to maintain a consistent and complete data base.

If we also consider the individual warfare commanders and the data files they need, the problem becomes more complex. We can also rank the importance of each commander and the importance of each data file to each commander and include this in our optimization problem.

1.3 PROBLEM

The problem is therefore as follows-we are given the following:

1. A set of M data files
2. A set of N nodes to store the data files
3. The probabilities of any node or link being destroyed from which the probabilities of any particular commander at one node can access any particular file at another node.
4. A set of L commanders.
5. The costs of assigning a particular file to any particular node.
6. The query rates for any particular file emanating from any particular node. The query rate is the rate at which files are requested.
7. The update rates for any particular file emanating from any particular node. The update rate is the rate at which files are updated (changed).

We desire the following:

1. To locate single or multiple copies of

---

*This research was supported by the Office of Naval Research under contract ONR/N00014-77-C-0532 (NR 041-519).

the M files at the N nodes such that the files will be accessible to the commanders who need the files.
2. To locate the files at nodes that will provide the least amount of cost. The cost can be related query communication costs, update communication costs and file storage costs.

1.4 TRADE-OFF OF COSTS

There is a trade-off in costs if we consider the following costs:

1. query communication costs
2. update communication costs
3. storage costs
4. the cost to the BG if a particular commander does not have access to the particular file he desires.

To minimize any one of the four costs we can do the following:

1. To reduce the query communication costs, we can store more redundant copies of each file so that each query can find its file with less communication cost. This is true since we shall assume that each query goes to the nearest node containing the file.
2. To reduce the update communication costs, we can store fewer redundant copies of each file so that each update will update fewer files and incur less communication costs. This is true since we assume each update goes to all the nodes containing the file.
3. To reduce the storage costs, we can store fewer redundant copies of each file so the cost will decrease.
4. To reduce the cost of non-accessibility of particular files to particular commanders, we can store more redundant copies of that particular file so that it has a higher probability of being able to be accessed by the particular commander.

The bottom line is: we cannot increase and decrease the number of redundant copies of a particular file. We would like to find the optimal number of redundant copies for all the files and where to store them.

2. LITERATURE SURVEY

The file allocation problem was first investigated by Chu [1]; a global optimization was considered, consisting in obtaining the minimum overall operating costs subject to two kinds of constraints; first, the expected time to access each file had to be less than a given delay, and secondly the amount of storage needed at each computer had not to exceed the available storage capacity. The number of copies of each file was assumed to be fixed. A generalized model was defined, in which storage and transmission costs were associated to file allocations; channel queues were modeled in order to introduce the constraint on the delay. The resulting linearized integer program was characterized by a very great number of variables even for application of limited dimensions; its solution was extremely hard from a conceptual viewpoint.

Casey [2],[7] considered the problem of allocating single files separately, but the number of copies of each file was not assumed to be fixed. Communication costs and storage costs of allocations were analyzed in order to determine the optimal set of nodes on which the file was to be allocated. The difference between retrieval and update transactions was stressed; while retrieval transactions are routed to only one copy of the file, update transactions are routed to all the copies, in order to preserve consistency of redundant information. Under the assumption of taking equal cost rates for retrieval and updates, theorems were given for limiting the number of replicated copies of the file on the basis of the update/query ratio; obviously the convenience of taking replicated copies decreases while the update/query ratio increases. Although the file allocation problem was analyzed for each file separately, Eswaran [3] proved that Casey's formulation was NP complete and, therefore, suggested to investigate heuristic approaches.

Morgan and Levin [4], examined both the allocation of files and transactions within a generalized, ARPA-like network. They adopted the user's viewpoint, assuming to be under the jurisdiction of a network management providing services at the market price. Because of this characterization of application environment, storage capacity constraints were not included; the provision of sufficient storage was considered a task of the network management. Therefore, by introducing some other simplifying assumption, the authors demonstrated that the multiple file allocation problem could be decomposed into independent (single) file allocation problems; they also developed an heuristic solution technique.

Finally two contributions to the file allocation problem have been very recently presented. Ramamoorthy and Wah [5] analyzed a relational Distributed Data Base; they observed that the general approach of query processing optimization consists in the minimization of communication costs. These communication costs are mostly due to data moves which are necessary for providing the logical correlation, expressed by the query, between files stored on different nodes. A logical operation which is particularly critical is the join operation between remote files; a join between two files can be performed only if the two files are co-located at the same node. Therefore the authors developed a model in which redundant files are introduced in order to avoid distributed joins, on the basis of the frequency of queries.

3. NOTATION

b_i = the available memory size of the i^{th} computer in terms of file [illegible]
x_{ij} = the file j is stored at node i
N_i = node i

$\Gamma(N_i)$ = set of nodes that are directly linked to node i

$\Gamma\ (N_i) = \Gamma\Gamma\ldots\Gamma(N_i)$

$R(N_i)$ = set of nodes that are accessible to node i by a path

$R(N_i) = \Gamma(N_i) \cup \Gamma^2(N_i) \cup \ldots$

α_i = the importance of commander i

β_{ji} = the value of file j to commander i

$P_{ij}(I)$ = the probability that file j is accessible to the commander at node i given an assignment I

$$A_i(j) \triangleq \begin{cases} 1 & \text{if } \exists k \text{ s.t. } j \text{ at } N_k \in R(N_i) \\ 0 & \text{if } \exists k \text{ s.t. } j \text{ at } N_k \in R(N_i) \\ 0 & \text{if } \forall k \text{ s.t. } j \text{ at } N_k, N_k \text{ destroyed} \end{cases}$$

r_j = the number of redundant copies of file j stored in the system

λ_{ji} = the volume of query traffic emanating from node j for file 1

ψ_{ji} = the volume of update traffic emanating from node j for file 1

d_{jk} = the cost of a unit of communication from node j to node k

σ_{kj} = the cost of locating a copy of a file j at the k^{th} node

T_{ij} = the maximum allowable query traffic time of the j^{th} file to the i^{th} node

a_{ijk} = the expected time for the i^{th} node to query the j^{th} file from the k^{th} node

I_ℓ = the set of node indexes representing a given assignment of file ℓ.

4. FORMULATION

Since $x_{i\ell}$ is a zero-one variable, the sum over all nodes i must be equal to the number of redundant copies of file ℓ. Therefore we have:

$$\sum_{i=1}^{N} x_{i\ell} = r_\ell \tag{4.1}$$

We define

$$I_\ell(R(N_i)) = \beta_{\ell i} A_i(\ell) \tag{4.2}$$

which denotes the accessibility of file ℓ to the commander at node i weighted by the importance of file ℓ to the commander at node i.

The initial formulation is as follows:

$$\min_{I_\ell} \sum_{\ell=1}^{M} = \min_{I_\ell} \cdot \sum_{\ell=1}^{M} \left[\sum_{j=1}^{N} \sum_{k=1}^{N} \psi_{j\ell} d_{jk} x_{\ell k} + \sum_{j=1}^{N} \min_{k \in I_\ell} \lambda_{j\ell} d_{jk} + \sum_{k=1}^{N} \sigma_{k\ell} x_{\ell k} - E\left[\sum_{k=1}^{L} \alpha_i I_\ell(R(N_i)) \right] \right] \tag{4.3}$$

such that

$$x_{11} + x_{12} + \ldots + x_{1M} \leq b_1;$$
$$x_{21} + x_{22} + \ldots + x_{2M} \leq b_2;$$
$$\vdots$$
$$x_{N1} + x_{N2} + \ldots + x_{NM} \leq b_N .$$

and

$$(1-x_{i1}) x_{k1} a_{i1k} \leq T_{i1};$$
$$(1-x_{i2}) x_{k2} a_{i2k} \leq T_{i2};$$
$$\vdots$$
$$(1-x_{iM}) x_{kM} a_{iMk} \leq T_{iM} .$$
$$x \geq 0$$

where the first term in the minimization corresponds to the cost of updating file ℓ at node k which was requested by node j, where each node k is a node that has file ℓ. The second term denotes the cost of querying file j at node k which was requested by node j, where node k is the closest node containing file ℓ. The third term represents the cost of storing file ℓ at node k. The last term denotes the cost associated with the expected accessibility of file ℓ to the commander at node i weighted by the importance of commander i.

The first set of constraints state that the number of files stored at any node must be less than the capacity at each node. The second constraint states that the expected time to retrieve a query is less than a certain threshold quantity. The last constraint states that all the zero-one variables are nonnegative.

If we now examine the last term in the minimization, we can simplify the expression. The expected value may be brought inside the summation. Since the importance of the commander i and the importance of file ℓ to commander i are not probabilistic, we can simply take the expected value of the accessibility. However, the expected value of the accessibility is simply the probability that commander i can access file ℓ given the allocation of redundant copies of file ℓ in the network. We have:

$$\begin{aligned} E\left[\sum_{i=1}^{L} \alpha_i I_\ell(R(N_i))\right] &= E\left[\sum_{i=1}^{L} \alpha_i \beta_{\ell i} A_i(\ell)\right]; \\ &= \sum_{i=1}^{L} \alpha_i \beta_{\ell i} E\ A_i(\ell) \quad ; \\ & \sum_{i=1}^{L} \alpha_i \beta_{\ell i} P_{i\ell}(I_\ell) \ . \end{aligned} \tag{4.4}$$

where

$$P_{i\ell}(I_\ell) = \begin{cases} 1 & \text{if } Pr(\exists k \text{ s.t. } \ell \text{ at } N_k \in R(N_i)) \\ 0 & \text{if } Pr(\exists k \text{ x.t. } \ell \text{ at } N_k \in R(N_i)) \\ 0 & \text{if } Pr(\forall k \text{ s.t. } \ell \text{ at } N_k, N_k \text{ destroyed}) \end{cases} \tag{4.5}$$

Where $P_{i\ell}(I_\ell)$ for one file ℓ is by definition:

$$P_{i\ell}(I_\ell) = \sum_{j=1}^{N} \prod_{\substack{k=1\\k\neq j}}^{N} (1-x_k)x_jP_{ij}$$

$$+ \sum_{h=1}^{N} \sum_{\substack{j=1\\j>k}}^{N} \prod_{\substack{k=1\\k\neq j\\k\neq h}}^{N} (1-x_k)x_jx_h[p_{ij}+(1-p_{ij})P_{ih}]\,..$$

$$+ \sum_{a=1}^{N} \sum_{\substack{b=1\\b>a}}^{N} \cdots \sum_{\substack{j=1\\j>h\\ \vdots\\j>b\\j>a}}^{N} \prod_{\substack{k=1\\k\neq j\\k\neq h\\ \vdots\\k\neq b\\k\neq a}}^{N} \qquad (4.6)$$

$$x_k[P_{ij}+(1-P_{ij})P_{ih}+(1-P_{ij})(1-P_{ih})P_{ig}\ldots]$$

which simplifies to the following:

$$= \sum_{j=1}^{N} x_jP_{ij} - \ldots$$

$$+ (-1)^{N+1} \sum_{j=1}^{N} \sum_{\substack{k=1\\k>j}}^{N} \prod_{\substack{m=1\\m\neq j\\m\neq k}}^{N} x_mP_{im}$$

$$+ (-1)^{N} \sum_{j=1}^{N} \prod_{\substack{m=1\\m\neq j}}^{N} x_mP_{im} + (-1)^{N+1} \prod_{m=1}^{N} x_mP_{im} \qquad (4.7)$$

where P_{ij} is the probability of accessibility between nodes i and j.

Substituting this back into the initial formulation we now have:

$$\min_{I_\ell} \sum_{\ell=1}^{M} C(I_\ell) =$$

$$\min_{I_\ell} \sum_{\ell=1}^{M} \left[\sum_{j=1}^{N} \sum_{k=1}^{N} \psi_{j\ell}d_{jk}x_{\ell k} + \min_{k\in I_\ell} \lambda_{j\ell}d_{jk} + \right.$$

$$\left. + \sum_{k=1}^{N} \sigma_{k\ell}x_{\ell k} - \sum_{i=1}^{L} \alpha_i\beta_{\ell i}P_{i\ell}(I_\ell) \right] \qquad (4.8)$$

such that

$$\sum_{j=1}^{M} x_{ij} \leq b_i \;; \quad 1\leq i\leq N$$

$$(1-x_{ij})x_{kj}a_{ijk} \leq T_{ij}; \quad 1\leq j\leq M \quad i\neq k$$

$$x\geq 0$$

We know that

$$\sum_{k=1}^{N} (\text{anything})x_{\ell k} = \sum_{k\in I_\ell} (\text{anything}) \qquad (4.9)$$

So substituting into the previous formulation we have:

$$\min_{I_\ell} \sum_{\ell=1}^{M} C(I_\ell) =$$

$$\min_{I_\ell} \sum_{\ell=1}^{M} \left[\sum_{j=1}^{N} \sum_{k\in I_\ell} \psi_{j\ell}d_{jk} + \min_{k\in I_\ell} \lambda_{j\ell}d_{jk} + \right.$$

$$\left. + \sum_{k\in I_\ell} \sigma_{k\ell} - \sum_{i=1}^{L} \alpha_i\beta_{\ell i}P_{i\ell}(I_\ell) \right] \qquad (4.10)$$

such that

$$\sum_{j=1}^{M} x_{ij} \leq b_i \;; \quad 1\leq i\leq N$$

$$(1-x_{ij})x_{kj}a_{ijk} \leq T_{ij}; \quad 1\leq j\leq M \quad i\neq k$$

If we remove the constraints, we are minimizing over disjoint sets, so we have

$$\min_{I_\ell} \sum_{\ell=1}^{M} C(I_\ell) = \sum_{\ell=1}^{M} \min_{I_\ell} C(I_\ell) \qquad (4.11)$$

Let's now try to $\min C(I_\ell)$ for a particular ℓ. The following of theorems will set bounds on how to allocate the files and determine when not to allocate files.

5. THEOREMS

First let's look at allocating one file ℓ optimally by placing redundant copies at different nodes, so without loss of generality let $I=I_\ell$.

<u>Theorem I</u>: If $\psi=\rho\lambda_j$ for j=1,2,...,n then an r-node assignment cannot be less costly than the optimal one-node assignment if

1)
$$\rho \geq \frac{1}{r-1}, \qquad (5.1)$$

and

2)
$$\gamma \leq \frac{(\rho r-\ell-1)(1+\rho)}{\rho r} \sum_{j=1}^{N} \lambda_j d_{j1} + \frac{(1+\rho)}{r} \sum_{k=2}^{r} \delta_k$$

$$+ \frac{(1+\rho)}{\ell r} \sum_{j=1}^{N} \min_{k} \lambda_j d_{jk} + \frac{(r-1)}{r} \sigma_1 + \frac{1}{\rho r} \sum_{k=2}^{r} \sigma_k$$

Theorem I states that if we have allocated r-1 copies of a file and the two inequalities hold, then by allocating the rth file, our total cost will not be less than just allocating one file optimally. This will allow us to reduce the possible solution space in which the integer program must search. Now we can exclude all allocations with more than r-1 files from possible file allocations before execution of the integer program.

<u>Theorem II</u>: If for some integer $r\leq n$,

1)
$$\rho > \frac{1}{r-1}, \qquad (5.3)$$

and

2)

$$\rho<\frac{(\rho r-\ell-1)(1+\rho)}{\ell r}\sum_{j=1}^{N}\lambda_j d_{j1}+\frac{(1+\rho)}{r}\sum_{k=2}^{r}\delta_k$$

$$\frac{(1+\rho)}{\rho r}\sum_{j=1}^{N}\min_k \lambda_j d_{jk}+\frac{(r-1)}{r}\sigma_1+\frac{1}{\rho r}\sum_{k=2}^{r}\sigma_k \tag{5.4}$$

then any r-node file assignment is more costly than an optimal one-node assignment.

Theorem II states that if we have allocated r-1 copies of a file and certain conditions hold, then by allocating the rth file, our total cost will be more than allocating one file optimally. This will allow us to reduce the possible solution space in which the integer program must search. Now we can exclude all allocations with more than r-1 files from possible file allocations before execution of the integer program.

Define u_k as follows:

$$u_k = \sigma_k + \gamma_k + \sum_{j=1}^{N}\psi_j d_{jk}\ , \tag{5.5}$$

where $\gamma_k = \gamma_{(I\cup k)} - \gamma_{(I)}$ (5.6)

Then the cost function for any given assignment I is given by:

$$C(I) = \sum_{k\in I} u_k + \sum_{j=1}^{N}\lambda_j \min_{k\in I} d_{jk}\ . \tag{5.7}$$

Theorem III: If

$$C(I)\leq C(I\sim[k]) \quad \text{for } k=1,2 \tag{5.8}$$

then

$$C(I\sim[k])\leq C(I\sim[1,2]) \quad \text{for } k=1,2 \tag{5.9}$$

Theorem III states that our cost graph if the given vertex has a cost less than the cost of two vertices leading to it, then the cost of the predecessor of the two vertices is greater than either of the two vertices.

Theorem IV: Given an index set $X\subseteq I$, containing r elements with the following property:

$$C(I)\leq C(I\sim[x]) \quad \text{for each } x\in X \tag{5.10}$$

Then for every sequence $R^{(1)},R^{(2)},..,R^{(r)}$, which are subsets of X, such that $R^{(k)}$ has k elements and $R^{(k)}\subset R^{(k+1)}$, the following is true:

$$C(1)\leq C(I\sim R^{(1)})\leq C(I\sim R^{(2)})\leq C(I\sim R^{(3)})\leq ..C(I\sim R^{(r)}) \tag{5.11}$$

Theorem IV states that if a given vertex has a cost less than the cost of any vertex along the path leading to it, then the sequence of costs encountered along any one of these paths decrease monotonically. Thus in order to find the optimal allocation policy, it is sufficient to follow every path of the cost graph until the cost increases and no further. This will given a locally optimum allocation of which the global optimum is one of them.

This allow us to reduce the solution space of the integer program. Once we find a local optimum then we know that any more file allocation is not required. Hence the integer program will not have to search for solutions in that part of the solution space.

Theorem V: All optimal allocations will include site i if

$$\lambda_i \min_{j\neq i} d_{ij} > Z_i, \tag{5.12}$$

where

$$Z_i = \sigma_i + \gamma_i + \sum_{j=1}^{N}\psi_j d_{ji}\ . \tag{5.13}$$

Theorem VI: No optimal allocation including more than one site will include site i if the following is true:

$$Z_i > \sum_{j=1}^{N}\lambda_j(\max_k d_{jk}-d_{ji}). \tag{5.14}$$

Theorem V states that if the cost of having a local file copy is smaller than the smallest possible cost of sending queries elsewhere, then a local copy should unquestionably be included in the optimal allocation. This will require certain nodes to have files allocated there. Therefore the solution space required to be searched by the integer program will be reduced. The integer program can ignore any possible file allocations that excludes the files that are unquestionably allocated by Theorem V.

Theorem VI states the other extreme of Theorem V. If it costs more to maintain a local copy than we could possible save by having one, then we do not want one. This theorem will allow the integer program to ignore allocating files in locations that are definitely too costly and therefore reduce the possible solution space for the integer programming solution. The integer program can ignore file allocations which include file allocations that are ruled out by Theorem VI.

Define the following:

$$m_i = \lambda_i \min_{j\neq i} d_{ij}, \tag{5.15}$$

and

$$M_i = \sum_{j=1}^{N}\lambda_j(\max d_{jk}-d_{ji})\ . \tag{5.16}$$

Then for each i the real line is partitioned by m_i and M_i into three regions.

If Z_i falls into region I then it should unquestionably be included. If Z_i falls into region III it will be excluded unless all Z_i fall in region III, then just include the largest one. If Z_i falls in region II then it must be further considered.

These theorems are useful because now the region in which the integer program must search for solutions is reduced. We can force files to be allocated in region I and not be allocated in region III.

Theorem VII: By choosing $d_{jk}=1$, $\sigma_{kl}=\sigma_1$ and a completely connected network, then the cost function reduces to the following:

$$C(I_\ell)=\sum_{\ell=1}^{M}\left[\sum_{k=1}^{N}\sigma_\ell x_{k\ell}+\sum_{i=1}^{N}\sum_{k=1}^{N}\psi_{i\ell}x_{k\ell}+\sum_{k=1}^{N}\lambda_{k\ell}(1-x_{k\ell})-\sum_{i=1}^{L}\alpha_i\beta_{\ell i}p_{i\ell}(I_\ell)\right]. \qquad (5.17)$$

The decision rule for the initial file assignment is $x_{ij}=1$ if:

$$\lambda_{ij}+\psi_{ij}\geq\psi_{kj}+\lambda_{kj} \quad 1\leq k\leq N \quad k\neq i \qquad (5.18)$$

This theorem just states the initial file allocation for this special type of network.

Theorem VIII: Given a node k and a file j, then to store a copy of j in k leads to a reduction of the overall costs if the following holds:

$$\lambda_{kj}+\psi_{kj}>\sum_{i=1}^{N}\psi_{ij}+\alpha_j-\gamma_j\;. \qquad (5.19)$$

For allocating a new copy later, the theorem states that if allocation of a new copy leads to a cost decrement for the host node which is greater than the overcost due to the necessity of updating the additional copy and storage cost, then we should store the file there.

This theorem is useful because the solution to the file allocation problem does not require integer programming and therefore is not NP complete. It enables simple calculations to determine file allocation.

Theorem IX: Define the following allocations: $I'=I\cup[k]$ and $I''=I'\cup[i]$. If site i satisfies

$$Z_i>\sum_{j=1}^{N}\lambda_j\max[(d_{jk}-d_{ji}),0], \qquad (5.20)$$

for some site k in the network, then $C(I'')>C(I')$.

Theorem IX states that if site i is sufficiently costly, then by adding site i to an allocation which already includes k increases the total cost.

Theorem X: Define the following allocations $I'''=I\cup[i]$. If sites i and k satisfy

$$Z_i-Z_k>\sum_{j=1}^{N}\lambda_j\max[(d_{jk}-d_{ji}),0], \qquad (5.21)$$

then

$$C(I''')>C(I') \qquad (5.22)$$

Theorem X states that if:

$$Z_i-Z_k>\sum_{j=1}^{N}\lambda_j\max[d_{jk}-d_{ji},0] \qquad (5.23)$$

is satisfied, then replacing site k by site i in an allocation will increase the cost.

Theorem XI: A site i cannot be included in any optimal allocation if there exists another site k in the network such that

$$Z_i-Z_k>\sum_{j=1}^{N}\lambda_j\max[d_{jk}-d_{ji},0]. \qquad (5.24)$$

Theorem XI states that instead of determining that no more than one of some group of geographically close sites can be included in an optimal solution, Theorem XI states that certain sites may be excluded from being optimal allocations by the existence of better nearby sites. This theorem is useful because it allows us to reduce the possible solution space of the integer program. The integer program can ignore file allocations that allocate separate copies of the same file at geographically close nodes.

Theorem IX, theorem X, and theorem XI are extensions of work done by Grapa and Belford [6].

Theorem XII: The following are equivalent:

$$P_i(I)=\sum_{j=1}^{N}\prod_{\substack{k=1\\k\neq j}}^{N}(1-x_k)x_jp_{ij}$$

$$+\sum_{h=1}^{N}\sum_{\substack{j=1\\j>h}}^{N}\prod_{\substack{k=1\\k\neq j\\k\neq h}}^{N}(1-x_k)x_jx_h[p_{ij}+(1-p_{ij})p_{ih}]\ldots$$

$$+\sum_{a=1}^{N}\sum_{\substack{b=1\\b>a}}^{N}\cdots\sum_{\substack{j=1\\j>h\\ \vdots\\ j>b\\ j>a}}^{N}\prod_{\substack{k=1\\k\neq j\\k\neq h\\ \vdots\\ k\neq b\\ k\neq a}}^{N}x_k[p_{ij}+(1-p_{ij})p_{ih}+(1-p_{ij})(1-p_{ih})p_{ig}\,..]$$

and

$$= \sum_{j=1}^{N} x_j p_{ij} - \dots$$

$$+ (-1)^{N+1} \sum_{j=1}^{N} \sum_{\substack{k=1\\ k>j}}^{N} \prod_{\substack{m=1\\ m\neq j\\ m=k}}^{N} x_m p_{im}$$

$$(-1)^{N} \sum_{j=1}^{N} \prod_{\substack{m=1\\ m\neq j}}^{N} x_m p_{im} + (-1)^{N+1} \prod_{m=1}^{N} x_m p_{im} \tag{5.26}$$

where p_{ij} is the probability of accessibility between nodes i and j.

This theorem essentially states a generalization of the well known probability law:

$$P(ABC) = P(A) + P(B) + P(C) - P(AB) - P(BC) - P(AC) + P(ABC) . \tag{5.27}$$

6. CONCLUSIONS

We have formulated the file allocation problem in a C^3 context where vulnerability is an issue. The formulation considers:

1. The probability of commander accessing files;
2. The importance of commanders;
3. The importance of particular files to particular commanders.

The theorems have provided ways to cut down on the possible file allocations (solution space) in which the integer program has to search. Therefore, we reduce the amount of time required to solve for a solution using integer programming.

We have extended and proved twelve theorem, all applicable to the new formulation.

In the C^3 context, we may not need inger programming to solve for a solution if we make the following assumptions:

1. Connected network where all nodes are connected to each other;
2. Cost of communication is same between all nodes;
3. Cost of storing a file is the same.

In the area of further research, we plan to explore the effects of where the data sources are located on the file allocation problem. This would be applicable in a C^3 context, where sensor data may come from only a fixed set of nodes. The data must also pass through a processing node. The location of where the processing node should be is also an optimization problem which can be incorporated into our formulation.

REFERENCES

1. W.W. Chu, "Optimal File Allocation in a Multiple Computer System." IEEE Transactions on Computer, Vol. C-18, No. 10 (1969).
2. R.G. Casey, "Allocation of Copies of a File in an Information Network," AFIPS, SJCC (1972).
3. K.P. Eswaran, "Placement of Records in a File and File Allocation in a Computer Network," Information Processing, North Holland (1974).
4. H.L. Morgan, J.D. Levin, "Optimal Program and Data Locations in Computer Networks," CACM, Vol. 20, No.5 (1977).
5. C.V. Ramamoorthy, B.W. Wah, "The Placement of Relations on a Distributed Relational Data Base," Proc. 1st. Int. Conference on Distributed Computing Systems (1979).
6. E. Grapa and G.G. Belford, "Some Theorems to Aid in solving the File Allocation Problem," CACM, Vol. 20, No. 11, pp. 878-882 (1977).
7. R.G. Casey, "Allocation of Copies of a File in an Information Network," AFIPS, SJCC (1972).

EVALUATING C^3 SYSTEM SURVIVABILITY BASED ON RELIABILITY AND NETWORK ANALYSIS THEORY

Dr. Donald R. Edmonds
The MITRE Corporation
1820 Dolley Madison Boulevard
McLean, Virginia 22102

Abstract. A technique from reliability and network analysis theory is used to analyze C^3 system survivability. The methodology is demonstrated through an illustrative example and shows the effect that mutual backup of C^3 components as a form of redundancy can have on system survivability.

INTRODUCTION

The purpose of this paper is to present a technique from reliability theory to analyze C^3 system survivability. A typical military C^3 system is composed of sensors, communication links, information processors, command centers, weapons and human decisionmakers distributed over many locations or platforms. Moreover, the command and control (C^2) process encompasses such diverse functions as compiling information on threats, force performance and damage; assessing the status of enemy and friendly forces from the information compiled; and directing offensive and defensive operations through the use of weapons. As a result, the choice of a meaningful measure of C^3 survivability and approach to network analysis can present a formidable problem for the analyst.

Frank(1,2) suggests that the choice of technique depends on the criterion used to measure network survivability. The criterion, in turn, must be chosen on the basis of the particular system function being analyzed and the goal of the analyst. Thus, a single system which must perform several functions may require several survivability criteria. Among the many measures of survivability that he suggests for consideration are:

1. The ability of any node to communicate with any other node
2. The length of the shortest surviving path between each pair of nodes or between specified pairs of nodes
3. The average fraction of specified pairs of nodes able to communicate after attack
4. The average or maximum time required to transmit a high priority message from source to destination
5. The expected percentage of the surviving forces receiving the message within a given period of time
6. The probability that a specific unit will receive the transmission within a given period of time.

In spite of the usefulness of several measures associated by function, it is highly desirable to have an overall measure of C^3 system survivability extending over multiple C^2 functions. This has led the author to define C^3 system operational survivability as the probability of successfully executing all tasks required to sense the presence of intruders, assess the threat, and direct offensive and defensive actions through use of weapons. The probability P_s of C^3 operational survivability is a function of the probability of kill of C^3 components (sensors, information processors (staff), command, communication links and weapons) plus the pattern of mutual backup among components.

Steps in the methodology to determine P_s are:

1. Identify the critical C^3 functions
2. Associate the functions with system components
3. Examine the effect of loss of any components upon system operational survivability

a. Identify mutual backup between components

b. Identify cuts

c. Determine minimal cuts

d. Iteratively derive the structure function for the network based on minimal cuts

e. Take the expected value of the structure function.

A technique of network analysis from reliability theory (namely, construction of a structure function) provides a bridge between the loss of individual components and the operation of the entire system.(3) This technique, coupled with the concept of mutual backup among components, provides an approach for the analysis of C^3 systems survivability in which various components can perform the tasks originally assigned to destroyed components.

MATHEMATICAL RELATIONSHIPS

Suppose in a system of N components that X_i, i = 1, 2, ..., N are state random variables such that

$$X_i = \begin{cases} 1, & \text{if component i performs satisfactorily} \\ 0, & \text{if component i fails (i.e., is destroyed).} \end{cases}$$

The state vector $(X_1,X_2,\ldots,X_N)$ of the system can take on 2^N realizations since each X_i, i = 1, 2, ..., N, is equal to either zero or one. The "performance" of the system is measured by a binary random variable known as the structure function $\phi(X_1,X_2,\ldots,X_N)$ where

$$\phi(X_1,X_2,\ldots,X_N) = \begin{cases} 1, & \text{if the system performs satisfactorily} \\ 0, & \text{if the system fails.} \end{cases}$$

The probability of operational survivability, P_s, is given by

$$\begin{aligned} P_s &= P(\phi(X_1,X_2,\ldots,X_N) = 1) \\ &= E(\phi(X_1,X_2,\ldots,X_N)) \qquad \text{eq(1)} \end{aligned}$$

where E is the expected value operator.

The structure function can be written in terms of the elements of minimal cuts. Suppose we denote minimal cuts as Z_j, j = 1, 2, ..., J where, for example, with J = 4 minimal cuts and N = 4 components

$$Z_1 = (X_1,X_2)$$

$$Z_2 = (X_1,X_3)$$

$$Z_3 = (X_2,X_3)$$

$$Z_4 = (X_3,X_4)$$

A system will fail if and only if all the components in at least one of the minimal cuts fails. Likewise, the system will operate if at least one of the components in each minimal cut operates. Hence minimal cuts operate as a series system so that the structure function is given by

$$\begin{aligned} \phi(X_1,X_2,\ldots,X_N) &= \min(\phi_1(Z_1), \phi_2(Z_2), \ldots, \phi_J(Z_J)) \\ &= \phi_1(Z_1)\, \phi_2(Z_2) \ldots \phi_J(Z_J) \qquad \text{eq(2)} \end{aligned}$$

where $\Phi_j(Z_j)$ is the structure function for minimal cut Z_j, j = 1, 2 ..., J. For the example,

$$\phi_1(Z_1) = 1 - (1 - X_1)(1 - X_2)$$

$$\phi_2(Z_2) = 1 - (1 - X_1)(1 - X_3)$$

$$\phi_3(Z_3) = 1 - (1 - X_2)(1 - X_3)$$

$$\phi_4(Z_4) = 1 - (1 - X_3)(1 - X_4)$$

if X_1, X_2, X_3, and X_4 are independent.

Now Z_1, Z_2, ..., Z_J may contain common elements of X_1, X_2, ..., X_N so that they are not independent random vectors. As a result, in general, one cannot treat ϕ_1, ϕ_2, etc. as independent. Nevertheless, if the X_i, i = 1, 2, ..., N are independent random variables and if the probability of X_i is given by

$$X_i = \begin{cases} 1, & \text{with probability } p_i \\ 0, & \text{with probability } q_i = 1 - p_i \end{cases}$$

then the expected value of X_i is

$$\begin{aligned} E(X_i) &= 1\, p_i + 0\, q_i \\ &= p_i \end{aligned}$$

$$E(X_u\, X_v) = p_u\, p_v \text{ for all } u \neq v$$

and

$$E(X_u\, X_v\, X_w) = p_u\, p_v\, p_w \text{ for all } u \neq v \neq w$$

and so forth. Moreover, because $X_i^m = X_i$ for all powers m = 1, 2, ..., all the exponents of any individual X_i in the structure function ϕ can be reduced to the power one. Consequently, one may readily solve for P_s in terms of p_i or q_i, i = 1, 2, ..., N by taking the expected value of the structure function ϕ written in terms of X_i, i = 1, 2, ..., N.

STATE VECTOR REPRESENTATION OF A CUT IN TERMS OF X AND Y AND ASSOCIATED STRUCTURE FUNCTION

Above we defined the state random variable X_i as a binary random variable such that

$$X_i = \begin{cases} 1, & \text{if component } i \text{ performs satisfactorily} \\ 0, & \text{if component } i \text{ fails (i.e., is destroyed).} \end{cases}$$

Alternatively, we can define the state random variable Y_i such that

$$Y_i = \begin{cases} 0, & \text{if component } i \text{ performs satisfactorily} \\ 1, & \text{if component } i \text{ fails (i.e., is destroyed).} \end{cases}$$

The structure function of a system was also defined above as a binary random variable $\phi(X_1,X_2,\ldots, X_N)$ such that

$$\phi(X_1,X_2,\ldots,X_N) = \begin{cases} 1, & \text{if the system performs satisfactorily} \\ 0, & \text{if the system fails.} \end{cases}$$

In this notation whenever $\phi(X_1,X_2,\ldots,X_N)$ is equal to 1, it indicates the system is operating. An alternative structure function $\Upsilon(Y_1,Y_2,\ldots,Y_N)$ can be defined such that whenever it is equal to 1, it would indicate that the system has failed. That is, let

$$\Upsilon(Y_1,Y_2,\ldots,Y_N) = \begin{cases} 1, & \text{if the system fails} \\ 0, & \text{if the system performs satisfactorily} \end{cases}$$

Now the structure functions ϕ and Υ are related as follows

$$\phi(X_1,X_2,\ldots,X_N) = 1 - \Upsilon(Y_1,Y_2,\ldots,Y_N)$$

so that

$$\begin{aligned} P_s &= P(\phi(X_1,X_2,\ldots,X_N) = 1) \\ &= P(\Upsilon(Y_1,Y_2,\ldots,Y_N) = 0) \\ &= 1 - P(\Upsilon(Y_1,Y_2,\ldots,Y_N) = 1) \\ &= 1 - E(\Upsilon(Y_1,Y_2,\ldots,Y_N)). \qquad \text{eq(3)} \end{aligned}$$

Thus, eq(3) can be used instead of eq(1) to compute P_s. In subsequent sections of this paper the state vector $(Y_1,Y_2,\ldots,Y_N)$ and the associated structure function $\Upsilon(Y_1,Y_2,\ldots,Y_N)$ will be used, since the notation Y_i instead of X_i simplifies the computational aspects of deriving a structure function. Accordingly, eq(3) will be used to compute P_s.

COMPUTATIONAL METHODOLOGY FOR DERIVING STRUCTURE FUNCTION

Whereas the theoretical aspects of deriving the structure function are rather simple, the computational aspects require rigorous bookkeeping, especially when the number of components, N, and the number of minimal cuts, J, are sizeable quantities. There is a rather easy paper and pencil method of deriving the structure function iteratively which has been followed in developing computer programs to accomplish the derivation. The steps of the paper and pencil method are provided below. The example presented earlier is used here for illustrative purposes. Given the list of minimal cuts $(Y_1,Y_2)(Y_1,Y_3)(Y_2,Y_3)$ and (Y_3,Y_4) in term of Y_i, where $i = 1, 2, \ldots, N$, the structure function can be iteratively derived as follows:

<u>Step 1.</u> Additively combine the first two minimal cuts in terms of Y_i, $i = 1, 2, \ldots, N$, and subtract their product. Any powers greater than one may be omitted. Using the example, with minimal cuts identified in terms of (Y_1,Y_2) and (Y_1,Y_3), a function G_u is obtained

$$G_u = Y_1Y_2 + Y_1Y_3 - Y_1Y_2Y_3.$$

<u>Step 2.</u> Set $u = 2$.

<u>Step 3.</u> Additively combine the next minimal cut in terms of Y_i, $i = 1, 2, \ldots, N$, with G_u and subtract their product, giving G_{u+1}. Note that $Y_i^m = Y_i$ for all powers $m = 1, 2, \ldots$. Therefore, all the exponents of any individual Y_i in G_{u+1} can be reduced to the power of one. For example, for $u = 2$, introduction of the minimal cut identified as (Y_2,Y_3) gives

$$\begin{aligned} G_{u+1} = G_3 &= Y_2Y_3 + Y_1Y_2 + Y_1Y_3 \\ &\quad - Y_1Y_2Y_3 - \left[Y_1Y_2Y_3 \right. \\ &\quad \left. + Y_1Y_2Y_3 - Y_1Y_2Y_3\right] \\ &= Y_2Y_3 + Y_1Y_2 + Y_1Y_3 \\ &\quad - 2Y_1Y_2Y_3 \end{aligned}$$

<u>Step 4.</u> If $u + 1$ is less than the number of minimal cuts J, increase u by one integer and go to Step 3. Otherwise, go to Step 5.

<u>Step 5.</u> When $u + 1$ equals the number of minimal cuts J, then

$$P_s = 1 - E\left[G_{u+1}\right]$$

For example, for u = 3 so that u + 1 = 4 = J

$$\begin{aligned} G_4 &= Y_3Y_4 + Y_2Y_3 + Y_1Y_2 + Y_1Y_3 \\ &\quad - 2Y_1Y_2Y_3 - \Big[Y_2Y_3Y_4 \\ &\quad + Y_1Y_2Y_3Y_4 + Y_1Y_3Y_4 \\ &\quad - 2Y_1Y_2Y_3Y_4\Big] \\ &= Y_3Y_4 + Y_2Y_3 + Y_1Y_2 + Y_1Y_3 \\ &\quad - 2Y_1Y_2Y_3 - Y_2Y_3Y_4 - Y_1Y_3Y_4 \\ &\quad + Y_1Y_2Y_3Y_4 \end{aligned}$$

and

$$\begin{aligned} P_s &= 1 - E\Big[G_4\Big] \\ &= 1 - q_3q_4 - q_2q_3 - q_1q_2 \\ &\quad - q_1q_3 + 2q_1q_2q_3 + q_2q_3q_4 \\ &\quad + q_1q_2q_4 - q_1q_2q_3q_4 \end{aligned}$$

EXAMPLE

Consider the C^3 system in Figure 1. The objective of this system is to sense the presence of an enemy intruder, assess the information and assign a weapon to kill the intruder. Three distributed sensors S_1, S_2 and S_3 provide information over communication link L_1 to an information processing center I. The information from I is transmitted over communication link L_5 to a command post C. The command post C directs weapons W_1 and W_2 over communication link L_4. As backup to I, the sensors S_1, S_2 and S_3 are connected to C by communications link L_2. It is presumed C can perform the tasks assigned to I if I is destroyed. In addition, as backup to C, I is connected to W_1 and W_2 by communication link L_3. It is also presumed I can do the tasks assigned to C if C is destroyed.

It is further assumed that the areas of sensor coverage of a potential enemy intruder overlap for S_1 and S_2, so that each can mutually back up the other. Likewise, assume that areas of sensor coverage overlap for sensors S_2 and S_3, so that each can mutually back up each other. Lastly, it is assumed that either W_1 or W_2 can be assigned to an enemy intruder.

Figure 2 is a network diagram for the C^3 system in Figure 1. The components of the C^3 system are shown as branches connecting dummy nodes. All paths of successful operation of the C^3 system are represented by starting from the left-most (source) node and reaching the right-most (sink) node through the directed graph. Branches on the horizontal line represent one possible path of successful operation in which all three sensors detect the intruder, the sensor information is transmitted over L_1 to I, I transmits processed information over L_5 to C, and C directs orders to W_1 over L_4. The other paths are additional paths which represent the mutual backup given in the statement of the example.

The network diagram in Figure 2 is used to identify cuts and subsequently minimal cuts from which a structure function is derived. A cut is a set of components that, by failing ensures the failure of the system (i.e., disjoining the source node from the sink node). A minimal cut is defined as a cut involving a minimum set of components that, by failing ensures the failure of the system. There are 96 cuts out of 128 states in this example. Fortunately, as in many other examples, the number of minimal cuts is significantly less than the number of cuts. In this example there are 12 minimal cuts. They are tabulated in Table 1.

TABLE 1
LIST OF MINIMAL CUTS

S_1S_2	$L_2L_3L_5$	$L_1L_4L_5$
S_2S_3	CL_1	IL_4
L_1L_2	CI	L_3L_4
IL_2	CL_3	W_1W_2

The minimal cuts in Table 1 have been obtained by observation. Presently no general algorithm exists to determine minimal cuts. A specialized form of an algorithm is given in Reference 4 in which components back up each other on a one-on-one basis only. Such a situation exists for the sensors in Figure 2.

PARAMETRIC ANALYSIS - WITH MUTUAL BACKUP BETWEEN I AND C

Assuming

$$q_S = q_{S_i} \text{ for } i = 1, 2, 3$$

$$q_L = q_{L_u} \text{ for } u = 1, 2, \ldots, 5$$

$$q_I = q_C$$

and

$$q_W = q_{W_j} \text{ for } j = 1, 2$$

eq(3) for P_s with mutual backup between I and C results in

$$P_s = \left[1 - 2q_S^2 - q_S^3\right]\left[1 - q_{CC}\right]\left[1 - q_W^2\right] \qquad \text{eq}(4)$$

where

$$\begin{aligned} q_{CC} = 2q_C &\left[2q_L - 3q_L^2 - 2q_L^3 + 5q_L^4 - 2q_L^5\right] \\ + q_C^2 &\left[1 - 4q_L + 4q_L^2 + 2q_L^3 - 5q_L^4 \right. \\ &\left. + 2q_L^5\right] + 2q_L^2 + 2q_L^3 - 5q_L^4 + 2q_L^5 \end{aligned}$$

Figure 3 shows how the factor $1 - q_{CC}$ varies parametrically for values of q_L and q_C when mutual backup exists between components I and C. The value q_L represents the probability of jamming. (Curves quite similar to the case of $q_L = 0$ in Figure 3 result for the first factor versus q_S and third factor versus q_W in the relationship for P_s in eq(4).) For values of $q_S = 0.4$, $q_C = 0.4$, $q_L = 0.3$ and $q_W = 0.2$, the value of $P_s = (.74)(.52)(.96) \approx 0.4$. Note that eq(4) can be highly dependent on q_C and q_L.

PARAMETRIC ANALYSIS - WITHOUT MUTUAL BACKUP BETWEEN I AND C

It is of interest to demonstrate how the factor $1 - q_{CC}$ varies parametrically for values of q_L and q_C without mutual backup between I anc C. If the C^3 system lacks mutual backup capability between I anc C, then

$$\begin{aligned} 1 - q_{CC} = &(1 - q_{L_1})(1 - q_I)(1 - q_{L_5}) \\ &(1 - q_C)(1 - q_{L_4}) \qquad \text{eq}(5) \end{aligned}$$

(Note that L_2 and L_3 are superfluous in this case.)

Assuming

$$q_L = q_{L_i} \text{ for } i = 1, 4 \text{ and } 5$$

$$q_I = q_C$$

then $1 - q_{CC}$ in eq(5) reduces to

$$1 - q_{CC} = (1 - q_C)^2(1 - q_L)^3 \qquad \text{eq}(6)$$

Figure 4 shows how the factor $1 - q_{CC}$ in eq(6) varies parametrically with values of q_L and q_C without mutual backup between I and C. Comparison of Figures 3 and 4 indicate that mutual backup between I and C can have a profound effect on enhancing system survivability. For values of $q_S = 0.4$, $q_C = 0.4$, $q_L = 0.3$ and $q_W = 0.2$, the value of $P_s = (.74)(.12)(.96) \approx 0.1$ as compared to 0.4 with mutual backup between I and C computed above.

STATUS OF RESEARCH

Research to date consists of developing a generalized algorithm for deriving a structure function from minimal cuts. This includes computational enhancements to save computer storage and computation time as described in Reference 4. (For example, for a system of 13 components with 77 minimal cuts, it took approximately 14 minutes in CPU time to compute the structure function on an IBM 370 computer and 7 minutes on an IBM 4341.) Future research will be focused on developing a generalized algorithm and computer program to find minimal cuts instead of relying on the use of network diagrams.

SUMMARY

The intent of this paper has been to present a technique from reliability and network analysis theory to analyze C^3 system survivability, demonstrate the methodology through an illustrative example, and show the effect of mutual backup of C^3 components on system survivability as a form of redundancy.

REFERENCES

1. Frank, H., "Survivability Analysis of Command and Control Communications Networks -- Part I," IEEE Trans on Communications, Vol. COM-22, No. 5, May 1974, pp. 589-595.
2. ________, "Survivability Analysis of Command and Control Communications Networks -- Part II, "IEEE Trans on Communications, Vol. COM-22, No. 5, May 1974, pp. 596-605.
3. Hillier, F.S. and Lieberman, G.J., Operations Research. San Francisco: Holden-Day, Inc., 1974 Second Edition. Chapter 13, Reliability Theory, pp. 579-588.
4. Edmonds, D.R., Emami, G. and Hise, W.B. "Computational Methodology for Evaluating ADP Operational Survivability," The MITRE Corporation, MTR-81W00152, September 1981.

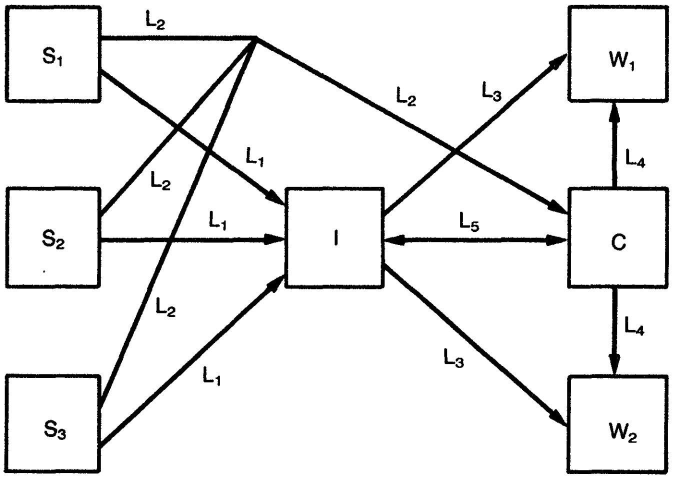


FIGURE 1
EXAMPLE OF C^3 SYSTEM

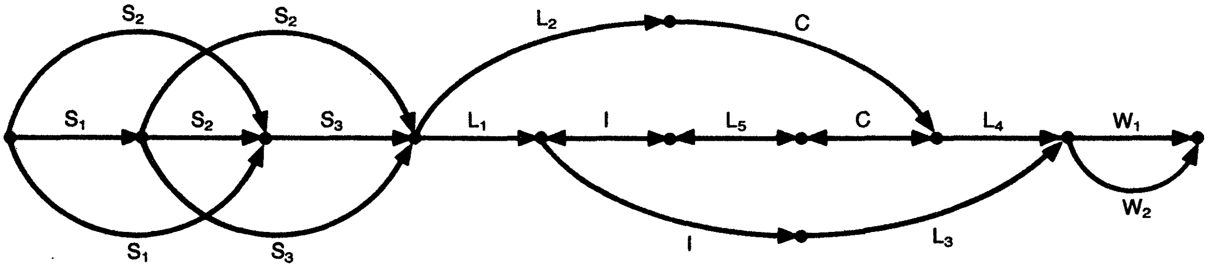


FIGURE 2
NETWORK DIAGRAM

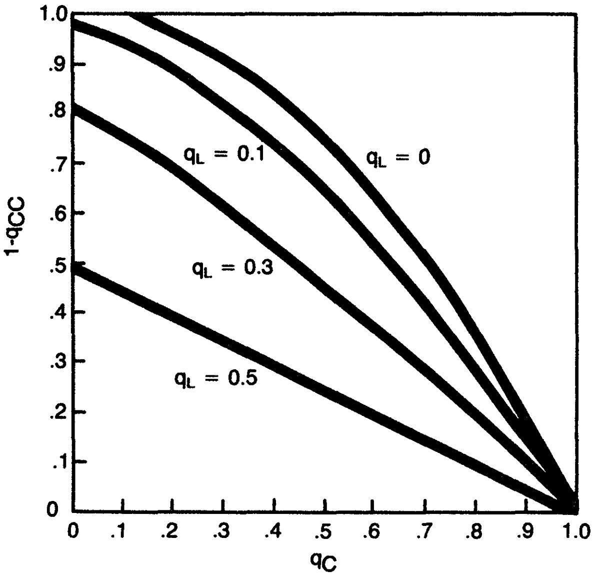


FIGURE 3
1-q_{CC} VERSUS q_C WITH MUTUAL BACKUP BETWEEN I AND C

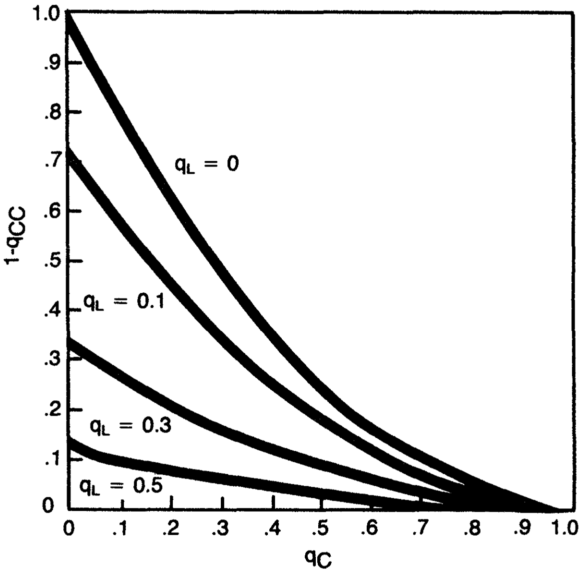


FIGURE 4
1-q_{CC} VERSUS q_{CC} WITHOUT MUTUAL BACKUP BETWEEN I AND C

OPERATING SYSTEMS FOR TACTICAL DISTRIBUTED PROCESSING NETWORKS (OSTDPN)

Carolyn A. Hutchinson

Systems Engineer, Comptek Research, Inc., San Diego, CA

Timothy T. Kraft

Systems Engineer, Comptek Research, Inc., San Diego, CA

Abstract. While the advantages of a flexible data transport mechanism are evident and have been thoroughly investigated, the impact of this technology upon the system's software architecture is not entirely understood for realtime command and control applications. The SHINPADS system control concept of "dedicated but reassignable system resources" or the Litton distributed processing system concept of dynamically reassignable master/slave control represent the current state of the art in tactical system design. However, they do not represent what is considered true distributed processing systems, since they do not incorporate the concept of cooperative autonomy of control (i.e., decentralized control). Involved with decentralization of control are issues central to the concept of distributed data processing and critical to the successful application to realtime systems. This paper will develop and explain some of these issues as they apply to realtime command and control systems.

1. INTRODUCTION

For many combat system designers, distributed processing is visualized as multiple computers attached to a reliable interconnect medium. This vision of distributed hardware resources offers the promise of increased and new capabilities for a combat system including:

- ability to reconfigure around a failed component (flexibility, survivability)
- faster throughput from multiple small computers working together (performance)

There are several current proposed and existing combat system designs which are based on distributed hardware described above. They include the Litton Distributed Processing System [Mauriello, 1981], SHINPADS [Carruthers, 1981] and XHDP [Jensen, 1978]. Most have a software structure which causes a dedicated allocation of functions to resources (possibly supported by dynamic reassignable backup). Generally the control structures are hierarchical, characterized by master/slave (or possibly dynamic master/slave) interactions between resources. In other words, the software has built basically federated systems on top of distributed hardware. This may restrict the ability of the system to provide the advantages of distributed processing.

This project represents the start of the investigation of operating system issues of tactical distributed processing networks, and is an on-going effort sponsored by NAVSEA 61R. The work is performed under the technical direction of NOSC Code 824 which coordinates the work of Comptek with a related effort at Carnegie-Mellon University (CMU).

The initial effort has been to define tactical and operational requirement areas which impact distributed processing, develop reference models which characterize the tactical and technical objectives, and develop architectural concepts for resource management and database control in tactical distributed processing networks [Hutchinson, 1981]. This led to the identification of the key operating system issues which must be resolved before distributed processing technology as a whole can be successfully introduced into tactical systems. The continuing effort to resolve the issues includes experimentation, defined in the OSTDPN transition plan. The first results from the planned experimentation are expected next summer.

This paper introduces this new area of Navy research in distributed processing by defining the scope of research (Section 2), identifying the key issues, and indicating the approach which is currently taken to resolve the issues (Section 3).

2. SCOPE OF OSTDPN

This section defines "operating systems" and "distributed processing" for this investigation. This helps define the scope of OSTDPN and was a necessary first step because there is widespread ambiguity in the Navy as to what constitutes distributed processing, and even less of a common understanding of the role and issues of an operating system for a distributed processing network.

2.1 Scope of Investigation: Operating Systems

For this study it was important that the definition encompass all of the aspects traditionally considered part of an operating system, including: network executive, distributed database control, and local executives. For this reason the following broad definition was adopted: An operating system is that part of major subsystems which controls the resources of the subsystem. The major subsystems for a tactical distributed system are the application subsystem (which includes processors and tactical application processes as resources), the database management subsystem (which has data as a resource), and the data transfer subsystem (which includes the bus or network). Thus the operating system of a distributed tactical system includes process control, database control, and data transfer control. This study focused on process control and database control.

In general, the functions of resource control are similar for all resources, whether the specific resource is a process, processor, memory, IPC path, data or some abstract combination of physical and logical resources. The typical resource control functions include:

- allocation - determine and schedule the use of resources and provide access to the resources (e.g., process to processor binding, provide access to remote data, allocate the bus)
- synchronization or concurrency control - coordinate use of resources when resources are shared or when multiple resources work cooperatively to provide a service (e.g., control concurrent access to shared data, synchronize parallel processing of tasks)
- maintain resource - detect and recover from faults (e.g., reconfiguration).

For OSTDPN, resources and resource control are defined in terms of the object model [Jones, 1979] where a resource (data, process) is an abstract data type which is encapsulated by one or more managers which perform the resource control functions for the resource (allocation, synchronization, maintenance). The managers are themselves typed objects, which allows the system resources to be defined at different layers of abstraction.

2.2 Scope of Investigation: Distributed Processing

For OSTDPN, the mere distribution of computers throughout a ship interconnected within a network does not, in itself, qualify the system as distributed processing. The operating system must also be considered as an essential element of the system.

The OSTDPN study adopted Enslow's criteria [Enslow, 1978] for determining whether a processing network is a distributed processing system.

Enslow's definition of distributed data processing systems has five components:

- A multiplicity of dynamically reassignable resources.
- Physical distribution of these physical and logical resources interacting through a communication network. This implies loose-coupling of processing resources.
- A high-level operating system that unifies and integrates the control of the distributed components.
- System transparency, permitting services to be requested by name only.
- Cooperative autonomy, characterizing interaction of resources - no hierarchy of control.

These criteria are expressed graphically in Fig. 1, which gives the three dimensions characterizing distribution, and outlines the region of hardware, control and database decentralization allowable for distributed systems.

This definition is strict in that all of the elements listed above must be present to be uniquely defined as a distributed processing network. It is especially stringent when applied to tactical processing networks. For example, though ship's combat systems have large numbers of interconnected computers and displays, they are not distributed processing systems because:

- Processes and operators are assigned to specific components and are not dynamically reassignable.
- There is a general lack of system transparency; i.e., data and requests for services are directed to specific physical resources.
- The hierarchical control structures extensively use master/slave control with lack of cooperative autonomy.

The DDG-51 combat system design is considering distributing the combat direction system functions among command and the individual warfare areas; but without high level operating systems to integrate control, the DDG-51 combat system will become a federated network, rather than truly distributed. The proposed Litton distributed processing system which has multiplicity of dynamically reassignable computational resources, physical distribution of processors, system transparency, and a high-level executive would not qualify under Enslow's definition because of its extensive use of a centralized, albeit reassignable, hierarchy of control using dynamic master/slave control processes.

In summary, the careful definition of distributed processing rules out many current and proposed systems with multiple computers on a network and physically partitioned databases because of limitations of the operating system:

- The systems lack an operating system which allows resources to be physically distributed yet conceptually viewed as global resources, available to the whole system.
- The operating systems are implemented with centralized resource control structures.

An underlying assumption of the proposed systems is that the functions of a tactical system are independent enough to be partitioned. This would obviate the need for a global operating system which unifies and integrates control. While this assumption may be valid for current combat systems, studies show this assumption will no longer apply to advanced combat systems, for example, the Advanced Combat Direction System (ACDS) [NAVSEA, 1982]. While current combat systems are functionally separated by warfare area, each supported by a separate database, ACDS requires the integration of all of this information into an integrated multi-source database (IMSDB) to better support command decision-making. ACDS is also required to have improved performance and robustness compared with existing systems. For this reason ACDS Block 2 is a candidate for distributed processing. Thus the IMSDB may be physically partitioned or replicated for robustness and performance, yet it must remain conceptually integrated to support the functional capabilities of ACDS. For the first time in tactical systems there will not be an isomorphism between the conceptual view of the system and the physical system. It becomes the responsibility of the operating system (database control) to provide a conceptual view of a single resource by providing access to remote data, maintaining the consistency copies of tracks which are updated concurrently, and recovering the database in case of failure.

3. OSTDPN ISSUES

The results of the preliminary tradeoff studies of OSTDPN have determined that one of the major issues which must be resolved before distributed processing can be successfully used for a tactical system such as ACDS is decentralized control:

- The decentralization of control is a key difference between more centralized systems and true distributed systems (as indicated in Section 2).
- It is hypothesized that decentralizing control functions is necessary to gain the promised advantages (and possibly incur costs) of distributed systems.

The next subsections will give some of the motivation for decentralizing control by explaining why decentralized control may be necessary to improve the robustness of tactical systems. The feasibility of decentralizing control in a realtime environment is in question, especially because of the impact of decentralized control on performance. The last two subsections present the motivation for an approach to decentralizing two different types of resource control function, resource allocation, and concurrency control.

3.1 Decentralized Control

The current state of the art in resource control offers the following basic alternatives:

- Centralized control - system resources are controlled by one manager.
- Partitioned control - control of system resources can be divided among several managers. This division can be by resources or type of control function (e.g., StarOS, Medusa) or along other dimensions [Jensen, 1980].

The single manager implementing centralized control can optimally manage the system resources to achieve overall system objectives, yet the manager is a single point of failure. If the control of system resources is partitioned among multiple managers, then management can only achieve subsystem goals.

Decentralized control is a new type of operating system control investigated at CMU which is characterized by the following:

- There are multiple managers per resource.
- The managers negotiate and come to a consensus as to resource use.
- The managers cooperate to manage resources according to "systemwide" criteria.

This inculcates the concept of cooperative autonomy discussed in Section 2.

In order for the cooperating managers to participate in an activity requiring negotiation and consensus, it is clearly necessary that the managers communicate and share information. In fact, the major problem in decentralizing control is caused by the lack of visibility into system state due to incomplete and inaccurate information. To understand why this is key, assume that managers in a distributed system could have any information about peer managers and shared resources instantly and completely. Supported by this type of perfect status, the manager could develop a "mutual model" of the other managers' state, resource use needs and strategies. The availability of perfect information, assumed above, means that the manager can develop mutual models which are accurate to any arbitrary degree. The extreme case would be if the managers replicated each other's algorithms. Thus, the results of the decisions and actions of the distributed managers can be exactly the same as the results with one centralized manager. Without "perfect status" it might not be possible to build a model which is good enough to achieve those results, no matter how sophisticated the modeling is.

3.2 Motivation for Decentralized Control

One of the primary attractions of decentralized control to tactical systems is that it offers the possibility to improve system robustness. Robustness is the system's capability to adapt to unexpected changes in the processing environment, such as processor failure or load/mode changes. Fig. 2 shows centralized, partitioned, and decentralized control in the concept diagram of the object model. From the diagram it can be seen that centralized control is vulnerable with respect to the single manager, which is a single point of failure and potential bottleneck. Partitioned control, on the other hand, is vulnerable with respect to resource failure. For example, assume in the diagram that the manager of each subsystem needs two resources to provide a service. If one resource fails, its manager will not have access to resources encapsulated by the other manager, and so the entire subsystem will be disabled by the failure of one resource. Only when control is decentralized is the system robust with respect to either manager or resource failure:

- If a resource fails, then both managers will still have access to the remaining resources.
- A manager fails, then the other manager can still provide access to the services of all of the resources. Also, the peer managers should be able to recognize the situation as part of their common protocol. In other words, the other managers can respond to the failure of a given manager as part of their normal operations.

Currently, a common approach to improving the reliability of centralized management is to provide a backup of the central manager with dynamic reconfiguration in case of failure. For example, in the Litton distributed processing system there is a backup for the global executive which monitors the primary executive and takes over if the primary executive fails. Since this could be an alternative to decentralized control, it is worthwhile to examine the approach more carefully to understand the need for decentralized control. This concept is shown in Fig. 3 where the primary manager allocates, synchronizes and maintains the system resources, while the activities of the higher level (backup) manager are to monitor the primary manager and take over in case of failure.

There are several potential difficulties with this approach. Because the primary manager is centralized, the state of the resources it controls are known only to the primary manager. This information would be lost with the failure of that manager, even though control was dynamically switched to a backup after failure occurred. The loss of this state information at the primary executive would most likely disrupt normal system operation for a period of time while the system recovers.

A more disastrous possibility is that the loss of the information on the status of the resources (which may be cooperating processes) will leave the system in an inconsistent state, from which recovery is not possible.

Another concern is that the backup manager is itself centralized, which raises the following questions:

- What happens if the backup manager fails first (before the primary manager which it is monitoring)?

- What happens if the backup attempts to take over from a healthy manager?

These limitations are caused by a single backup which doesn't have the status of the resources. Some possible solutions to the problems include:

- Having two or more backups which monitor the primary and decide when to "take over."

- Having the "backup(s)" maintain some information on the status of the resources as well as the manager. This is the concept of a "hot spare" implemented in some tightly-coupled systems.

However, both of these "solutions" raise the difficult issues of decentralized control.

- If there are two or more backup managers, then there is the issue of how they cooperate. Do the peer backups agree by consensus that a failure has occurred and do they negotiate to determine which backup should take over the duties of the primary manager? Or is there a hierarchy of backups characterized by master/slave relationships?

- If the backup manager(s) attempts to maintain the same state information on the resources as the primary manager, then there is the problem of how to keep this state information consistent, when it is assumed that the status messages will be inaccurate and incomplete in loosely-coupled systems. As indicated in Section 3.1, this is a key technical issue in decentralizing control.

Thus, to overcome the vulnerability of centralized control using a backup strategy, it may be necessary for the multiple backups to adopt some aspects of decentralized control. However, decentralized backup managers would be "passive" compared to the decentralized resource managers (shown in Fig. 3) in the sense that the backups do not perform the control functions directly for the resources.

The preceding paragraphs have given some of the motivation behind the interest in decentralized control as a means for improving system robustness. If decentralized control is needed for system robustness, it is necessary to understand the impact of decentralized control on other important system capabilities, such as performance. It seems intuitive that decentralizing control will have a negative impact on system performance because of the increased overhead of negotiation among the peer managers. Because performance (throughput, response time) is critical to realtime systems, this is an issue of decentralized control which must be addressed up front.

The following paragraphs will discuss this issue for two types of control functions, resource allocation and concurrency control. The paragraphs indicate an approach which, if successful, will reduce the overhead of decentralized control.

3.3 Resource Allocation

A classic example of resource allocation is process to processor binding. In this case, there are n processors in a network which are the resources to be allocated. Assume that control of these processor resources is decentralized and that there is a peer manager at each processor. When a process comes into existence at a processor, the manager at that processor must decide which processor in the net should do the work. The managers will have the same global objective, which is to allocate processes to processors such that the system work load is balanced. This will allow the system to adapt to load changes in the processing environment and so supports system robustness. The managers will allocate the process to the least loaded processor. The decision will be based on information about the state (i.e., loading) of all of the processors in the network. Thus the correctness of the decision depends on the quality of the information. If it is required that the allocation decisions made by the peer managers be correct in the sense that they are the same decisions as would be possible with centralized control, then the decentralized managers will need perfect state information. In general, this would require that the managers would need to exchange full information. This could be very expensive in terms of communication overhead and so could degrade realtime performance.

An approach to this difficulty is to re-examine the assumed requirement that every new process must be assigned to the least loaded processor. In the example, if a process was occasionally sent to a processor other than the least loaded, it is doubtful that the effectiveness of the system would suffer. In this case the "requirement" for load leveling by allocating to the least loaded processor is instead a "goal" which only needs to be approached. This in turn implies that it is not necessary to exchange the full state information and that the research can be directed to developing techniques which lower the communication overhead of state messages exchanged by the peer managers, yet keep the quality of state information used by the managers high. Research in this area is currently in progress at CMU, where Team Decision Theory is being used as a framework for investigating techniques including:

- Only messages which identify "important states" are sent. Process to processor binding decisions are then based on this information, which gives an incomplete view of the system state.
- Each manager monitors all traffic on the bus and uses this information to estimate the state of all other processors. Decisions are thus based on the estimated system status model, which is not entirely accurate.

Process to processor binding is an example of a resource allocation decision performed by a resource control function. There will be control functions which allocate the other resources, including system data, the bus, and processes. These control functions might also have different global objectives than load leveling, for example, allocating the resource to the highest priority user.

It is expected that the same basic approach can be used to decentralize other resource allocation functions because, in general, the global objectives of functions which decide resource allocation can tolerate some deviation from the optimal. Thus there is the potential of using "best guess" decision-making based on inaccurate or incomplete data, which in turn should reduce the overhead of decentralized control.

3.4 Concurrency Control

Concurrency control is the resource control function which regulates concurrent access of shared resources. Depending on the specific resource, the control function may be responsible for:

- synchronization
- maintaining data consistency
- maintaining correct process execution precedence

Concurrency control mechanisms are responsible for enforcing the consistency constraints of shared objects where "consistency" constraints define the invariant measures or predicates on the state of the resources [Eswaran, 1976]. Examples of consistency constraints are:

- savings + checking = constant, where savings account and checking account are data resources and the debit/credit operations move money from one account to the other.
- multiple copies of the same track item in the IMSDB must be equal
- directory at M1 = directory at M2, where M1 and M2 are peer managers and the directory defines their shared resources.

A promising approach for decentralizing resource allocation functions (Section 3.3) is to allow some deviation from the "correct" behavior of a centralized implementation. It is not immediately intuitive that this "best guess" approach can be used for decentralizing concurrency control. For example, consider the function which synchronizes concurrent processes so that they are executed in the correct order. Any deviation from the defined policy may not be acceptable. This is true for distributed commercial applications such as banking, where debit/credit processes must be synchronized without fail for the bank records to be consistent.

Most existing examples of decentralized concurrency control are with distributed database management systems [Kohler, 1981 and Bernstein, 1981], where the concurrency control mechanisms provide the same absolute consistency as systems with centralized control. The overhead in maintaining this degree of consistency makes these systems too cumbersome for realtime applications. Fortunately, recent work indicates that it may be possible to "relax" consistency requirements to improve concurrency control performance.

There are applications where the requirement for absolute synchronization is not as strong as for banking systems. These are, typically, applications where the results of system processing would be probabilistic estimates, even if performed by a centralized system with absolute synchronization. For example, an experimental distribution of the Hearsay speech understanding system functions successfully despite imperfect synchronization of processes [Lesser, 1979]. This is encouraging for tactical systems such as ACDS which interpret sensor signals to build an estimate of the tracks in the tactical environment. Because some of the information describing the tracks is only an estimate (e.g., position), it may not be necessary to maintain absolute equality of distributed "copies" of the same track item. This should reduce the overhead of concurrency control.

Not only is the classical view of consistency too strict for application level resources of tactical systems, but there is new evidence from research by M. Fischer at Yale and Jensen at CMU that it is also too strict for operating system level resources, such as directories.

4. CONCLUSIONS.

A conclusion from the discussion in Section 3 is that when the requirements for resource control can be fully and precisely defined, then there is an opportunity to design resource control functions which support both robustness and performance. For example, if the precise consistency requirements for a given resource are less strict than the classical (and generally assumed) absolute consistency, then it may be possible to design efficient decentralized concurrency control functions. The next step in the OSTDPN study is to pursue this approach with quantitative analysis and experimentation.

REFERENCES

Bernstein, P. S. and Goodman, N. (1981). Concurrency control in distributed database systems. ACM Computing Surveys, Vol. 13, No. 2, 185-220.

Carruthers, J. F. (1981). Ship systems integration and local communications networks. Presented at AOC.

Enslow, P. (1978). What is a "distributed" data processing system? Computer, January, 13-21.

Eswaran, K. P., Gray, J. N., Lorie, R. A., and Traiger, I. L. (1976). The notions of consistency and predicate locks in a database system, Communications of the ACM, November, 624-633.

Hutchinson, C., and Kraft, T. (1981). Operating systems for tactical distributed processing networks, Study Report, Comptek Report No. M1081F-07-1.

Jensen, E. D. (1978). The Honeywell experimental distributed processor - an overview. Computer, January, 28-38.

Jensen, E. D. (1980). Distributed computer systems. Computer Science Research Review, Carnegie-Mellon University, 53-63.

Jones, A. K. (1979). The object model: a conceptual tool for structuring software. In R. Bauer, R. M. Graham, and G. Seegmuller (Ed.), Operating Systems: An Advanced Course, Springer-Verlag, New York. pp. 8-16.

Kohler, W. H. (1981). A survey of techniques for synchronization and recovery in decentralized computer systems. ACM Computing Surveys, Vol. 13,No. 2, 149-183.

Lesser, V. R. and Erman, L. D. (1980). Distributed interpretation: a model and experiment. IEEE Transactions on Computers, Vol. C-29, No. 12, 1144-1163.

Mauriello, R. (1981). A distributed processing system for military applications - part 4: the software. Computer Design, 14-25.

NAVSEA 0967-LP-027-8602. (1982). ACDS Architecture and Systems Engineering Handbook (Preliminary), July.

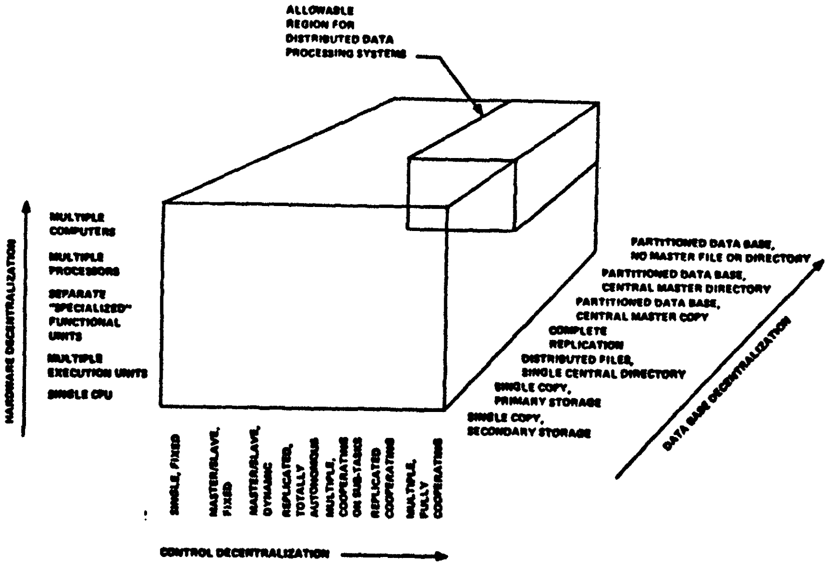


Figure 1: Dimensions Characterizing Distribution

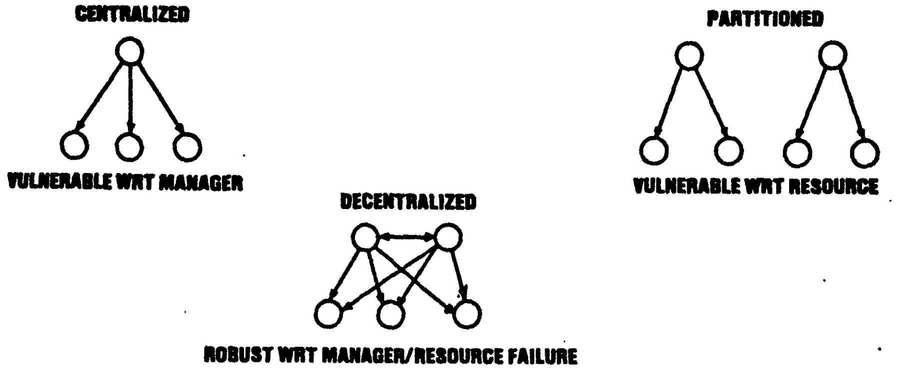
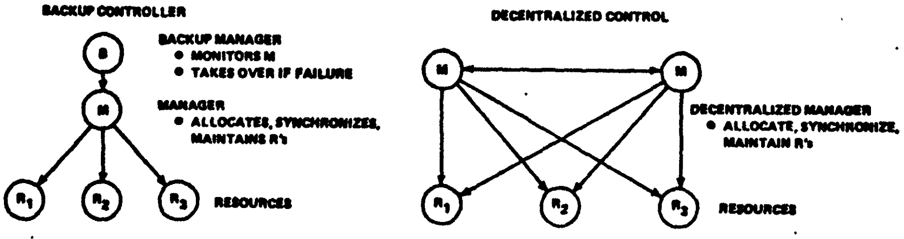


Figure 2: Motivation for Decentralized control is to Improve System Robustness



BACKUP MANAGER
- MONITORS M
- TAKES OVER IF FAILURE

MANAGER
- ALLOCATES, SYNCHRONIZES, MAINTAINS R's

DECENTRALIZED MANAGER
- ALLOCATE, SYNCHRONIZE, MAINTAIN R's

- SYSTEM PROCESSING DISRUPTED IF MANAGER FAILS
  - BACKUP DOESN'T KNOW STATE OF R's
  - MAY RESULT IN INCONSISTENT SYSTEM STATE
- HARD TO INCREMENT RELIABILITY
  - IF TWO (OR MORE) BACKUPS THEN ISSUE OF HOW THEY COOPERATE
  - "HOT" SPARE

- SYSTEM PROCESSING NOT DISRUPTED IF ONE M FAILS
  - OTHER MANAGER RECOGNIZES FAILURE AS PART OF NEGOTIATION PROTOCOL
  - M KNOWS STATE OF R's
- EASILY INCREMENT RELIABILITY
  - ADD ANOTHER M

Figure 3: Decentralized control versus Centralized control with Backup

A GENERIC COMMAND SUPPORT SYSTEM*

Maniel Vineberg
Clifford Warner

Naval Ocean Systems Center
San Diego, CA 92152

ABSTRACT

A generic command support system (CSS) is proposed. Two classes of CSS requirements, operational requirements and quality requirements, are defined. The generic CSS architecture is derived from those requirements. It is characterized by a network architecture based on layered protocols, a data-base management system architecture based on a layered, distributed, replicated data base, and a physical architecture mapped onto a "typical" military command structure. The proposed generic CSS architecture allows both commonality among CSSs for compatibility and uniqueness within CSSs for flexibility.

1.0 INTRODUCTION

This paper proposes a generic command support system (CSS) architecture. A generic CSS architecture will provide a baseline for future CSSs by providing common functions to meet common requirements as well as flexibility to provide unique functions to meet unique requirements and capture advances in CSS design. Commonality in the form of standard interfaces, protocols, data formats, etc, will increase CSS compatibility and decrease life-cycle costs. The approach is to derive the generic CSS architecture from requirements. Such features as the network, the database management system, and the physical organization, and such characteristics as layering, distribution, and redundancy, respond to explicitly stated requirements.

This paper uses work by Bronson [BRON], who argues that "C^3 systems should be designated as command support systems." Bronson's terminology and (implicit) definition of a CSS – a system which supports command processes (planning, organizing, coordinating, directing, and controlling) and staff functions (personnel, intelligence, operations, logistics, and fiscal) – have been adopted here. Bronson describes a "stimulus set" (situation, mission, administration and logistics, and command and signal), the information to be transformed as the commander, aided by his staff, executes the processes of command. Related work includes studies of HF Task Force Communication [BAKE], network survivability [GRNA], CSS description modeling [HARM], control modeling [KRAF], and design [BEAM].

CSS requirements and constraints are discussed in section 2. The generic CSS architecture is presented in section 3. The paper is summarized in section 4.

2.0 REQUIREMENTS AND CONSTRAINTS

CSSs are designed in response to two classes of requirements, operational requirements and quality requirements. Operational requirements are user requirements for services; quality requirements are the attributes of those services. In addition, CSSs are designed under certain constraints. The classes and subclasses of requirements described here are common to all CSSs. This partially justifies the proposed generic CSS. Unique requirements within these classes will be accommodated within the generic architecture.

2.1 Operational Requirements

Bronson [BRON] describes the processes of command, the major staff functions, and a stimulus set (data). These processes, functions, and data comprise the highest level of operational requirements for a CSS. However, to be useful in defining a CSS, these requirements must be given in terms of CSS services. The term "user," in the following paragraph, refers to the commander or any of his staff officers.

Five types of CSS service are defined here: 1) message processing (MP) – distribution, reception, display, and storage of user messages; 2) voice processing (VP) – transmission and reception of user speech; 3) data access (DA) – storage, retrieval, correlation, and deletion of data; 4) data processing (DP) – transformation of data; 5) word processing (WP) – composition of formatted or unformatted documents such as messages, reports, summaries, orders, etc. MP and VP allow a user to exchange information with other users and to transmit plans and orders and confirm that they have been received. DA allows a user to obtain plans, orders, and situation, mission administration and logistics, and command and signal data. DP allows a user to extract information from available data. WP allows a user to document decisions and compose orders, plans, and messages.

Staff officers who support the commander in the processes of command will require CSS services. Rather than construct an elaborate basis on which to predict how requirements for these services will vary from one staff officer to another (over a variety of CSSs), and from one major staff function to another, it is more efficient to assume that all staff officers will require all services for each function. However, it can be shown that the different staff officers will require access to different data. This implies a requirement to partition data spaces, spaces which may overlap, for presentation to each user.

2.2 Quality Requirements

Quality requirements will be specified quantitatively for various CSSs and will be met through design tradeoffs. CSS quality requirements – reliability, survivability, availability, responsiveness, security, flexibility, reconfigurability, ease of use, and compatibility – are defined here and discussed with respect to measurement, realization, and interrelationships.

---

*This work was sponsored by the Naval Electronics Systems Command under program element 62762N.

A "reliable" CSS can be depended upon to perform (respond to the user) according to specifications. Measures of reliability are mean time to failure and probability of successful operation (over a given time interval). Trouble reporting frequency and recurrence can indicate the rate of convergence of actual CSS performance to specified CSS performance. Fault tolerance and built-in test can be used to build reliable digital hardware. Scheduled maintenance can restore the margins of safety of fault-tolerant equipment on a periodic basis; this equipment must be maintenance free between maintenance periods. Use of a support environment can reduce software specification, design, and coding errors. Proper protocol design can ensure reliable communications. Comprehensive trouble reporting policies can facilitate the detection and correction of system problems. Redundancy, a technique which is useful to enhance reliability, may be implemented in added hardware and software and in time. Therefore, reliability must be traded against both CSS cost and responsiveness.

A "survivable" CSS continues to perform according to specification in spits of partial loss (eg, due to combat damage). Measures of survivability predict how much functional capability will be available after certain types and numbers of (CSS component) losses. Replication of critical functions and data and distribution of the communication subnetwork control among the network nodes can protect these CSS assets against loss. Adaptable message routing and communication protocols can allow the communication functions to "degrade gracefully" (remain viable at a lower level of capability) in the face of node destruction and communication link jamming. Survivability must be traded against system cost.

An "available" CSS is ready to use. A measure of availability is "down time" (the definition of "down" may vary among CSSs). Availability depends upon reliability and survivability since, during the scheduled-maintenance interval, the CSS must be maintenance-free.

A "responsive" CSS provides requested services with relatively little delay. High-performance digital hardware will help to reduce delay by bringing both functional capability and data close to the user. Network protocols can be designed and special-purpose communications processors can be used to achieve low message delay when data are not available locally. Responsiveness may be adversely affected by the provision of security and high reliability (although an unreliable CSS may not respond at all).

A "secure" CSS grants access to any system asset (service or data) to all persons authorized to use that asset and denies access to that asset to all other persons. Probabilities of deciphering codes and of compromising passwords can be computed. User passwords and asset keys can be used to deny unauthorized access. Data encoding and message encryption are techniques to disguise meaning and protect information. Data access can be restricted through efficient design of external schemas in the data-base management system. Alternative routings and dummy or background messages can disguise traffic patterns and, therefore, command hierarchy. Added security decreases reliability (additional hardware, software, and user procedures may introduce some errors and conceal others), decreases responsiveness, and increases sytem cost.

A "flexible" CSS is easily changed in response to changes in requirements and advances in technology. Flexibility is measured in terms of the costs (time and money) of making CSS changes. Well defined system interfaces, protocols, and data formats can facilitate system changes. Layered communications network and data-base management system architectures will simplify the addition or modification of data and of user (application layer) services, as well as the modification of support (lower layer) services. Flexibility can save time and money over the system life cycle. Conversely, failure to provide flexibility can shorten that life cycle.

A "reconfigurable" CSS performs as specified under any fixed or dynamically changing topology. Any deviation from this definition implies that the CSS is not reconfigurable. Distributed, adaptable network control protocols can allow the CSS to automatically reconfigure in response to changes in topology. Reconfigurability may be viewed as a subset of both reliability and survivability since, if the CSS is not reconfigurable, there will be situations (topologies) under which it cannot perform according to specifications.

A CSS is "easy to use" if, given "reasonable" CSS training, a user, with no other qualifications than a thorough knowledge of his area of military responsibility, is capable of using all relevant, available CSS assets to perform his job. Requests for assistance and consultation, "false" trouble reports, complaints about the CSS, unused features, and similar observations of user/CSS interaction are indications of ease of use. To exploit the capabilities of a sophisticated CSS effectively under the pressure of combat will require extensive training, no matter how well the CSS is designed. On-line training and assistance functions can help to reconcile CSS sophistication with the ease-of-use requirement. Monitoring of on-line functions, trouble reporting, and system usage patterns can expose problems in design, documentation, and training. In general, simplicity of use reflects sophistication of design, because a sophisticated CSS performs additional functions to simplify the user's task. Ease of use improves combined user/CSS reliability, since the easier the CSS is to use, the more likely the user is to use it correctly.

Two CSSs are "compatible" if they can exchange messages and share data. Compatibility requires strict adherence to certain communication protocol and data format standards. Therefore, compatibility may restrict flexibility.

2.3 Constraints

There are two types of constraint on a CSS. Constraints of the first type relate to resources available to develop and maintain the CSS. The primary resource constraints are time, money, personnel, existing artifact (hardware, software, etc.), and support tools. Time and money are dictated by the military and economic situation. Personnel are available in relation to available money; they are effective relative to experience and to available support. Existing artifact depends upon inventory (for "off-the-shelf" implementation), the state of the practice, and the state of the art. Support tools can be prepared in advance (under similar resource constraints). Constraints of the second type relate to the physical environment in which the CSS will operate. Examples of environmental constraints are shock, heat, noise, motion, and radiation tolerances as well as size, weight, shape and power limitations.

3.0 A GENERIC CSS

The generic CSS provides a framework within which to develop common solutions (standards) to meet requirements which transcend CSSs (eg, inter-CSS compatibility) and to isolate unique solutions, where common solutions are insufficient or unavailable, to meet unique requirements. The generic CSS is discussed in four parts: 1) network architecture: 2) data-base management system architecture: 3) physical organization; and 4) commonality and uniqueness issues.

3.1 CSS Network Architecture

The operational requirement for message and voice processing is satisfied within the CSS by a communications network. This network will also support data access. The CSS network architecture, shown in figure 1, is based upon the ISO seven-layer protocol architecture [TANN], which separates services into distinct layers.

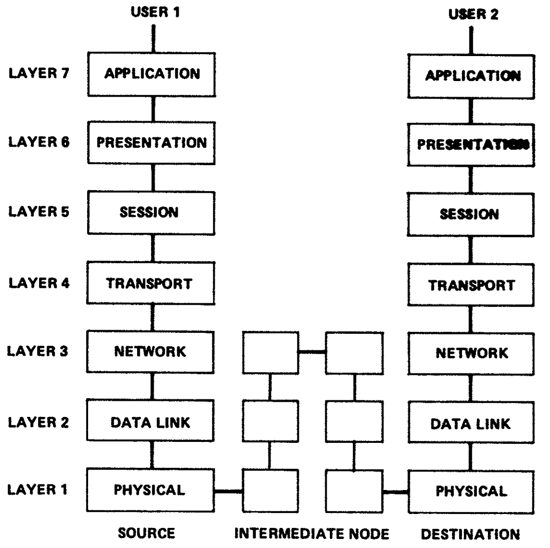


Figure 1. CSS network architecture.

A message from user 1 passes through the various layers from layer 7, the application layer, where the user message is composed, to layer 1, the physical layer, where the message is represented by electrical signals (representing binary ones and zeroes). The message might then be sent to an intermediate relay node where only the three lowest protocol layers need to be executed. The relay node would decode the message – from physical to network layer – and then recode it for transmission to user 2. It is received by user 2 at the physical layer, assembled and decoded as a message, and finally made available to him at the application layer. The application layer provides the services to satisfy user requirements. Each lower protocol layer provides services for the layer above it and relies upon services from the layer below it.

The layered CSS architecture responds to the quality requirement for flexibility. Layering will make it possible to provide services in separate, distinct processes (ie, hardware boxes or computer programs). This will allow the protocols in one layer to be designed and, if necessary, changed independently of those in other layers. Requirements on the processes at each layer will be written in terms of inputs, transformations on those inputs, and outputs. While (except in the layer closest to the user) these requirements are not visible to the user, they are visible from outside each process (at the interfaces) to system designers. A designer need only satisfy these requirements (and meet environmental constraints) to build a box for the CSS; when one box is replaced, the others need not be disturbed. The various layers and their associated functions are discussed below.

Functions in the application layer satisfy operational (user) requirements. Services provided by this layer include voice processing, data access, word processing, and concurrency control.

The presentation layer performs functions requested sufficiently often by application layer processes to warrant general support for them (rather than inclusion in each application layer program). Typical services include text compression, conversion between character codes, encryption, reconcilliation of format incompatibilities, verification of user access rights, virtual terminal protocols, and file transfer protocols.

A connection between two users (actually, between two presentation layer processes) is a session. In response to requests from the presentation layer, functions at this layer establish, maintain, and discontinue sessions. At this layer, parameters are collected to establish which services are required by the current user. Functions include binding (setting up a session between two processes), establishing terminal security, establishing priority level, identifying users, and preserving chain of command.

The transport layer accepts data from the session layer, groups them in smaller units, passes these to the network layer, and ensures that the units arrive correctly. Functions include point-to-point or multidestination service, determination of grade of service, determination of type of switching, circuit or packet, connection, pre-emption, packetizing of messages, packet accounting, and end-to-end flow control.

The network layer controls the operation of the communication subnet. Functions include routing of packets (the units of information exchanged at this layer) toward their destination and congestion control.

The data-link layer transforms individual bits into bit strings which appear free of transmission errors to the network. Functions include partitioning the input data into data "frames" and transmitting the frames sequentially, processing acknowledgement frames from receivers, creating and recognizing frame boundaries, synchronizing clocks, retransmitting destroyed frames, detecting duplicate frames, controlling flow, and controlling channel access.

The physical layer transmits bits over a communication channel. Services include defining voltage levels and pulse durations, providing full- or half-duplex channels, and establishing or disestablishing physical connections and other services related to the mechanical, electrical, and procedural interfaces with the communication medium.

3.2 Data-Base Management System (DBMS) Architecture

The CSS data base will be defined as all data collected in and available (to users) from the CSS. The data describe the stimulus set – situation, mission, administration and logistics, and command and signal. The CSS DBMS will be defined to be both the data base and the set of functions used to support data access.

In response to quality requirements for responsiveness and survivability, the DBMS will be distributed and replicated. Most data will be located at two or more different sites. Data necessary to support a user will be available to that user locally. Replication will ensure that if his copy of the data is lost, those data will not be lost to the CSS. The DBMS will be layered in response to a quality requirement for flexibility. Date [DATE] argues for "data independence", the "immunity of applications to change in storage structure and access strategy." That is, the application logic should not have knowledge of the data organization or the access technique. It should be possible to change one without changing the other. The layering, shown in figure 2, expands the ANSI/X3/SPARC three schema information architecture for centralized data bases [ANSI] to seven layers; two of the new layers, which specifically address distributed applications, were proposed recently [DEVO, BOUD].

User services are specified at the application layer. The user composes requests for data-base access in a query language; query language functions may also be imbedded in programs which the user may activate (eg, a program to display data). Services performed at this layer include query language processing, workspace management, and display. A request for service originating at this layer will be referred to as a "transaction". Transactions are created by a single user (at a single site) and are, in general, processed at more than one site (possibly including the originating site).

There may be several external schemas in a CSS DBMS. Each user sees a single external schema which bounds that part of the

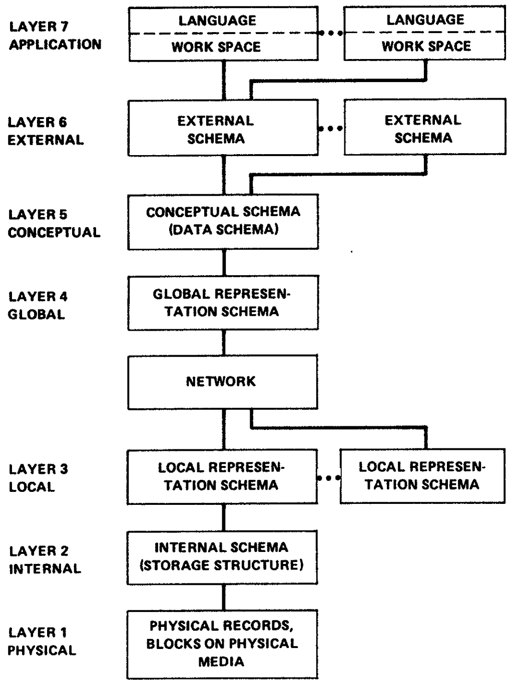


Figure 2. DBMS architecture.

data base and those operations available to that user. This allows data to be viewed differently (eg, called by different names) by different users. In particular, in response to the operational requirement for partitioning data with respect to major staff functions, users at different FAs would use different external schemas. In response to a quality requirement for security, certain data may be excluded from some external schemas, thereby denying access to certain users. The external schema insulates individual users from data-base expansion, change, and complexity. Requests for data-base access from the application layer are formulated in an "external data definition language", a language of storage and retrieval. Services at this layer include processing of storage and retrieval requests.

The conceptual schema is "a representation of the entire information content of the data-base" in an abstract form (relative to the form in which the data are physically stored) [DATE]. The conceptual schema is independent of data location, storage structure, or access strategy. Part of this layer is the data sublanguage which manipulates data entities or sets, under certain logical conditions, on the schema. Integrity constraints (data types, relations of magnitudes, ranges, etc) are enforced at this layer.

The global representation schema describes the entire data base, including which data are located at which sites. Each site has a copy of this schema enabling any site to coordinate transaction processing on the distributed DBMS. Services at this layer include selection of execution sites and data copies, selection of the order of operations, transmission of operation requests to other sites, and coordination of execution.

A local representation schema is the subset, for a particular site, of the global representation schema; it describes the portion of data at that site. Services at this layer include the mapping of operations to the internal schema based on storage structure.

The internal schema describes the stored files in the data base in terms of records, fields, and characters. The files may be sequential, indexed-sequential, hash-addressed, etc. The internal schema describes the structure of file records, the fields on which they are sequenced, the fields (if any) which can be used as search arguments for direct access, and the character formats (eg, decimal, binary, ASCII). Data sublanguage commands (from layer 3) are translated into file manipulation operations at this layer. These operations are based on such techniques as sequential access and direct access.

The physical layer includes the physical media (tapes, discs, etc) and the structure of data stored on those media. Functions at this layer include 1) associating stored fields to form records (eg, using adjacency or pointers); 2) sequencing (eg, using continuity, indices, or pointers); and 3) accessing data directly (eg, using indices, sequential scan, or hash addressing).

3.3 Physical Organization

The basic building blocks of the generic CSS architecture will be "CSS nodes" and communication "links" between those nodes. A CSS node will comprise equipment which supports a significant group of people and resources, where that group is normally confined to a relatively small, geographical area. Examples of prospective CSS nodes would be a collection of computers to support a ship, a battalion, or a squadron. A typical CSS will comprise CSS nodes linked by a long-haul communication subnetwork. CSS nodes will comprise networks of Major Staff Function Area computers (FAs). FAs will be linked within a CSS node by a broadcast medium (eg, by one or more data buses aboard ship).

A CSS must be survivable, available, and responsive. For many applications, a constraint will be low available communications bandwidth. A compensating technological opportunity will be the availability of relatively inexpensive, small, high-performance, reliable digital circuitry [DAVI]. One way to address the quality requirements within the communications bandwidth constraint is to exploit the technological opportunity afforded by new digital circuitry. Hardware can be employed "liberally" to distribute functional capability throughout a CSS, obviating the need to use communications resources to transfer functional capability. FAs can contain complete copies of those parts of the data base which might be of interest at their nodes. This distribution and replication will improve survivability (backup data and functional capability will be readily available), availability (users may continue to work even while temporarily isolated from the network), and responsiveness (data and functional capability will be available locally). In addition, hardware can be applied to provide local data reduction capability where necessary (eg, to sensors) to alleviate communications requirements. Figure 3 shows the configuration of a typical CSS node.

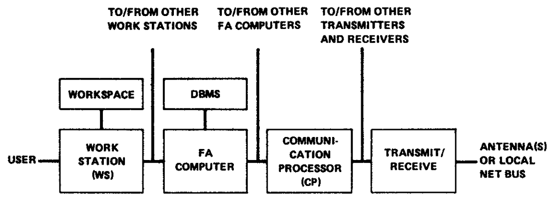


Figure 3. Typical CSS node.

3.3.1 Entry/Exit Points –

Three classes of entry/exit point are defined here: 1) user entry points (work stations and terminals); 2) sensors; and 3) automatic fire-control systems. A "work station" will be the primary user entry point into the CSS. Work stations will support all word processing, most data processing, and some data access, message processing, and voice processing. A work station will include a computer and such input/output components as a keyboard, a display, and a printer. At the interface level, work stations would be identical to allow any work station to be connected to any FA.

Each work station will maintain a workspace containing copies of recently modified data. This will allow a work station to be moved between FAs with a minimum of data loss. Terminals are remote user entry points. A terminal must implement all seven network protocol layers because the remote user will, at a minimum, need to format messages (WP, a layer 7 activity) and transmit signals (MP, at layer 1). Data access and data processing functions may also be included in the terminal.

Sensors must send data and receive commands. Scarce communications bandwidth will necessitate extensive data reduction; therefore, sensors will implement all seven network protocol layers. It is expected that sensor information will be sent to data correlation processes which reside at FAs. Therefore, sensors are not expected to require data access capability. Sensors will receive commands (to control operations) from users and control processes (probably resident on FAs).

Automatic fire-control systems must receive and interpret commands (to control operation) and return acknowledgements. Therefore, automatic fire-control systems must implement all seven network protocol layers. As threat sophistication decreases available response times, a requirement may arise for direct links between sensors and automatic fire-control systems.

3.3.2 Functional Area (FA) –

Each FA computer will perform peripheral control, DBMS, and communication functions. It will control peripheral devices and possibly a "backend" computer [MARY], to implement lower-level DBMS functions. FAs will support data access and message processing. It is expected that data processing capability will be limited to data correlation and automatic hardware (eg, sensor) control. The FA is expected to implement network layers 5 and 6 and DBMS layers 1 – 6 functions. A backend computer would implement DBMS layers 1 and 2 functions.

3.3.3 Communication Processors, Media, and Channels –

Each FA will connect to the communication subnetwork via communication processors which implement functions that support message and voice processing and data access. A CSS may include "communication nodes" comprising only a communication processor, transmitters, receivers, antennas, etc. Communication equipment and lower-layer protocols will vary among CSSs depending upon communication requirements and environmental constraints.

3.4 Commonality and Uniqueness

A generic CSS will provide common functions to satisfy common requirements and flexibility to meet unique requirements. In the layered network architecture, gateways between CSSs, at protocol layer 3, will allow inter-CSS communication. Only protocols in layers 4 and above will cross CSS boundaries. These protocols support a set of user services – message processing, voice processing, data access, data processing, and word processing. Since this set will transcend CSSs, common protocols can be used in these layers in response to the quality requirement for compatibility. Protocol layers 3 through 1 handle data and voice transmission over CSS communication subnetworks. These protocols depend upon traffic statistics and types, channel capacity and quality, propagation delays, postulated threats, throughput and robustness requirements, etc. CSSs can use unique protocols in these layers in response to unique communication requirements and environmental constraints.

Within the DBMS architecture, data will pass on the communication network (between the global and local layers) in conceptual schema format. Therefore, commonality within the conceptual schema will respond to the quality requirement for compatibility. Unique user requirements can be accommodated at the external layer while unique storage media characteristics can be accommodated at the internal and physical layers.

4.0 SUMMARY

A generic command support system (CSS) has been proposed. Two classes of CSS requirements, operational requirements and quality requirements, have been defined. In addition, two classes of CSS constraints, resource constraints and environmental constraints, have been defined. The proposed generic CSS comprises a network architecture based on layered protocols, a data-base management system (DBMS) architecture based on a layered, distributed, replicated data base, and a physical architecture mapped onto a "typical" military command structure and based on the concepts of distribution and replication to assure survivability and responsiveness. It was shown that the CSS architecture allows both commonality for compatibility and uniqueness for flexibility.

Within each layer of the network and DBMS architectures and for every "box" and link in the physical organization there are significant design issues. These issues can be partially addressed through development and experimental use of a testedbed. Issues such as system fault tolerance, robust communications, mobile operation, and multiple data copy correlation may require special attention.

5.0 REFERENCES

[ANSI] "The ANSI/X3/SPARC DBMS Framework Report of the Study Group on Data Base Management Systems," Tsichritzis, D. and Klug, A., eds., AFIPS Press, 210 Summit Ave., Montvale, NJ 07645, 1977.

[BAKE] Baker, D. Wieselthier, J. and Ephremides, E., "The HF Intra Task Force Communication Network Design Study," Proc of the Fourth MIT/ONR Workshop, San Diego, Vol III, October 1981, pp 7-18.

[BEAM] Beam, J. and Halushynsky, G., "A Systems Approach to Command, Control and Communications Systems Design," Proc of the Fourth MIT/ONR Workshop, San Diego, Vol II, October 1981, pp 227-241.

[BOLE] Bolen, N., "JTIDS Coordinates Tactical Elements," Defense Electronics, Vol 14, No. 1, January 1982, pp 104-105.

[BOUD] Boudenant, J., "SIRIUS-DELTA un cas concret: cohérence de bases de données réparties, partitionées et répliquées," INRIA, ctr-j-026, June 1980.

[BRON] Bronson, J., "Derivation of an Information Processing System (C3/MIS) Architectural Model – A user Perspective," Proc of the Fourth MIT/ONR Workshop, San Diego, Vol II, October 1981, pp 67-91.

[DATE] Date, C., "An Introduction to Database Systems," Second Edition, Addison Wesley, 1977.

[DAVI] Davis, R., "The DoD Initiative in Integrated Circuits," Computer, Vol 12, No. 7, July 1979, pp 74-79.

[DEVO] Devor, C., et al, "Five-Schema Architecture Extends DBMS to Distributed Applications," Electronic Design, Vol 30, No. 6, 18 March 1982, pp ss27-ss32.

[GRNA] Grnarov, A. and Gerla, M., "Multiterminal Reliability Analysis of Distributed Processing Systems," International Conference on Parallel Processing, Ohio State University, August 1981.

[HARM] Harmon, S. and Brandenburg, R., "Command, Control and Communications (C^3) Systems Model and Measures of Effectiveness (MOES)," Proc of the Fourth MIT/ONR Workshop, San Diego, Vol IV, October 1981, pp 182-206.

[KRAF] Kraft, T., and Murphy, T., "A Conceptual Control Model for Discussing Combat Direction System (C^2) Architectural issues," Proc of the Fourth MIT/ONR Workshop, San Diego, Vol II, October 1981, pp 93-108.

[MARY] Maryanski, F., "Backend Database Systems," ACM Computing Surveys, Vol 12, No. 1, March 1980, pp 3-25.

[TANN] Tannenbaum, A., "Network Protocols," ACM Computing Surveys, Vol 13, No. 4, December 1981, pp 453-489.

MESSAGE DELAY IN A COMMUNICATION NETWORK WITH UNRELIABLE COMPONENTS[1]

Victor O.K. Li

Department of Electrical Engineering, University of Southern California,
Los Angeles, CA 90089-0272

Abstract. In evaluating the performance of a communication network, researchers have traditionally distinguished between the issues of message delay and reliability. A network of queues model is generally employed to find the global and the end-to-end message delays, assuming that the network components are failsafe. On the other hand, other researchers have studied the network reliability, i.e. the probability that one can transmit messages between a pair of nodes, or between one node and the rest of the network. While reliability is an important performance measure, existing results do not allow us to study the impact of network failures on the message delay. This study of delay-reliability is important because, in the event of failure(s), a communication network will not simply stop operating, but rather will continue to operate at a degraded level of service. Existing research results are scarce, and they require enumerating all possible states of the system. Since the number of states of a communication network with n failure prone components is 2^n, these methods are restricted to small systems.

We present two new solution approaches that will not be doomed by the state-space explosion problem.

1. INTRODUCTION

In evaluating the performance of a communication network, researchers have traditionally distinguished between the issues of message delay and reliability. For example, **Kleinrock** [Kleinrock, 1964] determined the average message delay for all messages in a communication network, while in Li [Li, 1981], we developed a model for finding the end-to-end message delay, i.e. the message delay between a pair of nodes. Both studies, however, assume that the network components are failsafe. On the other hand, other researchers have studied the network reliability, i.e. the probability that one can transmit messages between a pair of nodes (See Ball [Ball, 1979]) for an excellent survey, or between one node and the rest of the network.(See Aggarwal and Rai [Aggarwal, 1981]. While reliability is an important performance measure, existing results do not allow us to study the impact of network failures on the message delay. This study of delay-reliability measure is important because, in the event of failures, a communication network will not simply stop operating, but rather will continue to operate at a degraded level of service. Existing research results are scarce. Bonaventura and colleagues [Bonaventura, 1980] proposed the following approach. The possible states of the network are enumerated, and the performance (e.g. message delay) of the network in each state is calculated. The performance-reliability is then the weighted sum of the performance at each state, where the weight is the probability of a particular state. Meyer (See Meyer [Meyer, 1980] and Meyer [Meyer, 1982]) used a similar approach to evaluate the performance of computer systems. However, while Bonaventura found the expected value of the performance measure, Meyer found the probability distribution of the performance measure. All three studies, however, predicate on enumerating the possible states of the system. Since the number of possible states of a communication network with n fail-prone components is 2^n, these methods are restricted to small systems. Other related studies in performance-reliability of computer systems are surveyed by Meyer [Meyer, 1980].

---

[1] This work was supported in part by the National Science Foundation under Contract No. ECS-8204495 and in part by the Joint Services Electronics Program under Contract No. F49620-81-C-0070.

Our goal is to develop a model to analyze the impact of network component failures on the message delay. We present two different approaches. In the first approach, we determine the impact of network failures on the routing variables. Since the message delay is a function of the routing variables, we can find the impact of failures on message delay through changes in the routing variables. In the second approach, we analyze the most probable states of the network. Since the network operates in these states most of the time, we can get a good approximation of the network performance without having to analyze all possible states.

2. IMPACT OF FAILURES ON ROUTING VARIABLES

Consider a communication network $G = (V,E)$, where V is the set of nodes, E is the set of links (Fig. 1). Let γ_{*i} be the generation rates of messages at node i; $1/\mu$ be the average message length; λ_{*i} be the total arrival rate of messages at node i, i.e. the sum of messages generated at node i and those in transit; and C_{ij} be the capacity of link (i,j). Messages are assumed to be generated in a Poisson manner, and their length is assumed to be exponentially distributed. Using a routing algorithm, such as Gerla's Flow Deviation Algorithm [Gerla, 1973], we can find the routing variables $\phi_{ij}(k)$, i.e. the fraction of messages at node i, destined for k, routed over link (i,j).

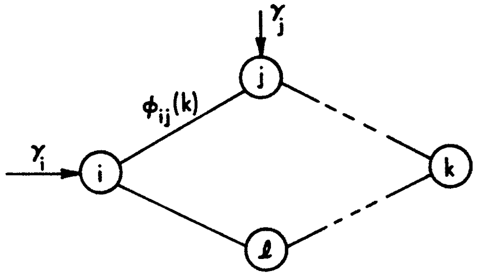


Figure 1. Communication Network Model

Therefore,

$$\lambda_{*i} = \gamma_{*i} + \sum_{j,k} \lambda_{*j}\,\phi_{ji}(k)$$

The total arrival rate of messages to link (i,j), denoted λ_{ij}, is given by

$$\lambda_{ij} = \lambda_{*i} \sum_{K} \phi_{ij}(k)$$

It is also assumed that the average message delay at link (i,j), denoted by t_{ij}, is a function of the traffic on link (i,j), i.e. $t_{ij} = f_{ij}(\lambda_{ij})$, where f_{ij} is any monotone increasing function of λ_{ij}. For example, in Kleinrock's model [Kleinrock, 1964],

$$t_{ij} = 1/(\mu C_{ij} - \lambda_{ij})$$

where C_{ij} is the capacity of link(i,j).

The average delay T for messages in the network is then given by

$$T = \frac{1}{\gamma}\sum_{(i,j)} L_{ij}$$

where γ is the total message generation rate and L_{ij} is the queue length at link (i,j). Alternately,

$$T = \frac{1}{\gamma}\sum_{(i,j)} \lambda_{ij}t_{ij} \tag{1}$$

using Little's Formula [Little, 1961].

To find the end-to-end delay T_{ij}, we note that

$$T_{ik} = \begin{cases} t_{ij} + T_{jk} & \text{prob } \phi_{ij}(k) \\ t_{ik} & \text{prob } \phi_{ik}(k) \end{cases}$$

Hence

$$T_{ik} = \sum_{(i,j)\in E} \phi_{ij}(k)(T_{jk} + t_{ij}) \qquad i,k\in V \tag{2}$$

where $t_{kk} = 0$.

If the routing is loop-free, we can find T_{ik} recursively. First, we solve Eqn. 2 for T_{rk} for all $(r,k)\in E$. Then we solve for T_{jk} for all $(j,r)\in E$, etc., until we find T_{ik}. If the routing is not loop-free, then we have to solve the set of equations represented by Eqn. 2 simultaneously.

From Eqn. 1 and 2, we see that T, T_{ij} can be found once we determine the routing variables $\phi_{ij}(k)$ and the link delays t_{ij}. Thus changes in delay due to failures can be analyzed if we can find the changes in the routing variables.

Our approach is best illustrated by a simple example. We have a communication network $G=(V,E)$, where the nodes are fail-safe and each link, say link (i,j), is available with probability p_{ij} and unavailable with probability $1 - p_{ij}$. Link failures are assumed to be independent. Consider a message at node i, destined for s, to be routed over link (i,j), which is unavailable. Suppose the routing strategy dictates that this message will be re-routed over the remaining links (i,k), for all k where $(i,k) \in E'$, proportional to $\phi_{ik}(s)$, then

$$\phi_{ij}(k) = \frac{\phi_{ij}(K)\cdot p_{ij}}{\sum_{(i,j)\in E'} \phi_{ij}(K)\cdot p_{ij}}$$

In addition, the flows λ''_{ij}(bits per second) on link (i,j) corresponding to the routing variables $\phi''_{ij}(s)$ must be scaled by a factor of $1/p_{ij}$. This is because link (i,j) is operative a fraction p_{ij} of the time. Therefore,

$$\lambda'_{ij} = \lambda''_{ij} / p_{ij}$$

The above procedure is summarized in Figure 2.

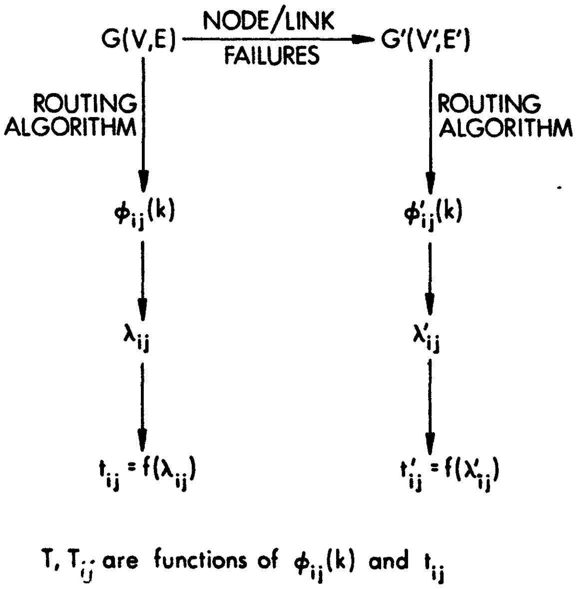


Figure 2. Changes in Routing Variables due to Network Failures.

In particular, if each link has the same probability of operation p, then $\phi'_{ij}(s) = \phi_{ij}(s)$, $\lambda''_{ij} = \lambda_{ij}$, $\lambda'_{ij} = \lambda_{ij} / p$. The average message delay is given by Eqn. 1:

$$T' = \frac{1}{\gamma} \sum_{(i,j)} \lambda'_{ij} f_{ij}(\lambda'_{ij})$$

Suppose $t_{ij} = 1/(\mu C_{ij} - \lambda_{ij})$, then

$$T/T' = \frac{\sum \lambda_{ij} / \mu C_{ij} - \lambda_{ij}}{\sum \lambda_{ij} / p\mu C_{ij} - \lambda_{ij}}$$

If traffic is light, i.e., $\lambda_{ij} << p\mu C_{ij}$, then

$$T/T' = \frac{\sum \lambda_{ij} / \mu C_{ij}}{\sum \lambda_{ij} / p\mu C_{ij}}$$
$$= p.$$

We have used a simple example to illustrate our approach. However, this approach can be used for a general unreliable network. In particular, we can generalize the approach to failure dependencies between adjacent links, links with different failure probabilities, and other than proportional re-routing in the event of failures.

3. ANALYSIS OF MOST PROBABLE STATES

Our second approach is described next. Instead of enumerating all possible failure states, we shall only consider the most probable states, say, the n most probable states in a network with n fail-prone components. Since the network is operating in these states most of the time, we can get a good approximation of the network performance without having to analyze all possible states. Identifying these most probable states is not a trivial problem. Suppose component i operates with probability p_i, and fails with probability $q_i = 1 - p_i$. The most probable state is given by $S_1 = (M_1, M_2, \ldots M_n)$ such that

M_i = operating mode if $p_i \geq q_i$
failed mode otherwise

where M_i is the current mode of component i, i =1,...n.

The second most probable state is derived from S_1 by switching the mode of one of the components, from the operating mode to the failed mode, or vice versa. This component, say k, is given by

$$P(M_k) = \min_i P(M_i)$$

Other most probable states derived from S_1 by switching one component can be identified easily. However, the rest of the most probable states cannot be found so easily. This is a problem that we are going to investigate.

We now illustrate our approach with the simple example shown in Fig. 3.

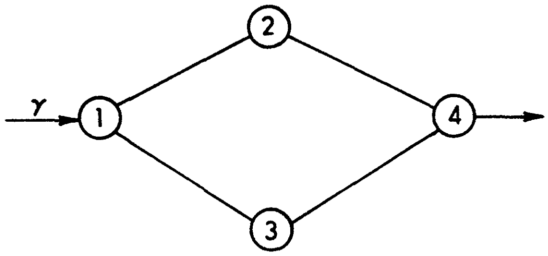


Figure 3 . Example Unreliable Network.

This network has four unreliable links. With probability 0.9, each link operates with the normal transmission capacity of 50 Kbps, while with probability 0.1, it operates in the failed mode, with a capacity of only 20 Kbps. This network has $2^4=16$ states: one state with no failures, four states with one failure, two states with two failures on the same path, i.e. links (1,2),(2,4) or (1,3),(3,4) have both failed, four states with two failures on different paths, four states with three failures, and one state with four failures.

We consider each of these 16 states, taking advantage of symmetry, and find the average global delay T. Since our network has only one external source of message arrival, the global delay is the same as the end-to-end delay from node 1 to node 4. The message delays as a function of the message arrival rate are plotted in Fig. 4.

The following observations were made. 66% of the time, the network operates with no failure, 29% of the time, it operates with only one failure. Thus 95% of the time, the network operates in one of the five most probable states. In other words, to get a good estimate of the network performance, we need only analyze the most probable states. The four-failure case gives us a worst-case scenario. The weighted delay, where the weights are the probability the network will be in those particular states, is not a very useful performance measure, since under no circumstances will the network be operating on this delay curve. An interesting measure is the average capacity delay. We calculate the average capacity of each link as 0.9 X 50 Kbps + 0.1 X 20 Kbps = 47 Kbps. The average capacity delay curve corresponds to the delay if each link has this capacity of 47Kbps. This curve gives us the delay characteristics if the network is one where the states change very rapidly compared to the message arrival rates.

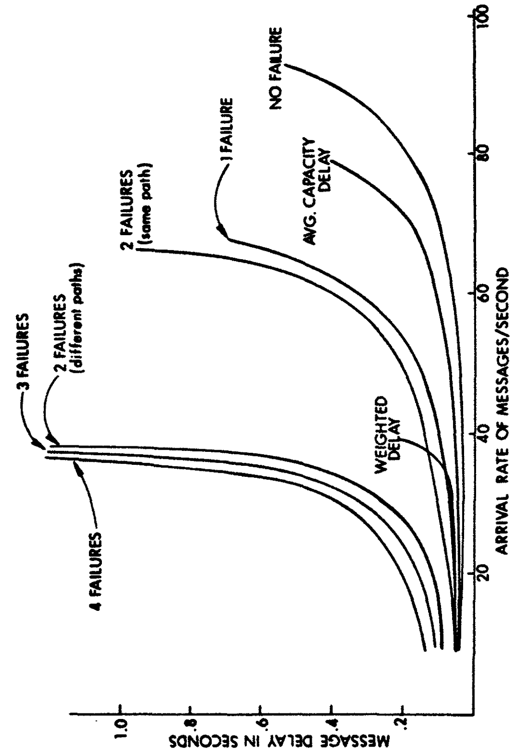


Figure 4. Message Delay under Different States.

4. CONCLUSIONS

We have presented two different approaches to analyze the impact of network failures on the message delay, and have illustrated our approach with two simple examples. There are several extensions of the work reported in this paper. First, we would like to extend our results to dependencies between component failures. For the first approach, we have to determine the new routing strategy in the event of failures, while for the second approach, it is necessary for us to find an algorithm to generate the most probable states without having to enumerate all possible states. Some of these extensions have already been initiated and will be reported in a subsequent paper.

REFERENCES

Aggarwal, K.K., and Rai, S. . Reliability Evaluation in Computer-Communication Networks. *IEEE Trans. on Reliability*, April 1981, *R-30(1)*, 32-35.

Ball, M.O. Computing Network Reliability. *Operations Research*, July-Aug 1979, *17(4)*, 823-838.

Bonaventura, V., et. al. *Service Availability of Communication Networks*, pages 15.2.1-15.2.6. IEEE, 1980.

Gerla, M. *The Design of Store-and-Forward(S/F) Networks for Computer Communications*. Technical Report UCLA-ENG-7319, School of Engineering and Applied Science, UCLA, 1973.

Kleinrock, L. *Communication Nets: Stochastic Message Flow and Delay*. : McGraw-Hill 1964.

Li, V. *Performance Models of Distributed Database Systems*. Technical Report LIDS-TH-1066, MIT Laboratory for Information and Decision Systems, February 1981.

Little, J.D.C. A Proof of the Queueing Formula $L = \lambda W$. *Operations Research*, 1961, *9*, 383-387.

Meyer, J.F. On Evaluating the Performability of Degradable Computing Systems. *IEEE Trans. on Computers*, Aug 1980, *C-29(8)*, 720-731.

Meyer, J.F. Closed-Form Solutions of Performability. *IEEE Trans on Computers*, July 1982, *C-31(7)*, 648-657.

VOICE RECOGNITION INPUT TO THE ARMY'S TACTICAL FIRE DIRECTION SYSTEM (TACFIRE)

G. K. Poock

Department of Operations Research, Naval Postgraduate School, Monterey, California

E. F. Roland

Rolands & Associates Corporation, Monterey, California

Abstract. This report will discuss the preliminary results of an Army sponsered research study on the applicability of voice recognition input to the tactical fire direction system, TACFIRE. The report will briefly review the purpose of TACFIRE and why the research was conducted. The results and conclusions of the year long research effort will be presented along with a detailed explanation of the three major phases of the research program. Although voice data entry to the system proved to be efficient in the laboratory environment, numerous implementation problems were encountered. Recommendations for implementation of a voice system for TACFIRE are summarized along with the requirements for future research.

INTRODUCTION

If there is a major conflict, the United States Army will face a tough, capable, and numerically superior enemy force. This force stresses rapid movement of about 50 to 60 kilometers per day protected by a lethal air defense and an extensive electronic combat capability. The United States has applied a large amount of technological resources to target acquisition to increase the ability to detect and locate the profusion of targets. At the same time technological resources have also been applied to the area of weapon effectiveness to allow attack of targets at ever increasing ranges.

Since the battlefield of the future will be characterized by numerous fast moving targets which must be attacked under short time constraints, the battlefield commander must have fast and accurate access to a prioritized list of possible targets. To assist in this complex job, the Army has developed a computer based management information system to improve field artillery command and control so a maneuver commander can employ his assets more effectively. This system, TACFIRE, provides the means to receive, store, combine, and sort target information; allocate firepower; compute ballistic firing data; and send firing orders to field artillery units.

Naturally, as with any management formation system, the command and co control support will be only as good as the information entered into the TACFIRE system. The speed and accuracy of the information entered into TACFIRE is the basis for a useful and timely command and control system. In an effort to increase the capability to enter data, the Army funded a one year research effort to investigate the advantages and disadvantages of current voice input technology. This paper will present the reasons why the research was undertaken, and the results and conslusions drawn from the study.

CURRENT STATUS OF INPUT TO TACFIRE

There are numerous ways to get information in and out of TACFIRE, and the majority of them include using one of the following input/output devices.

1. Digital Message Device (DMD) is a hand held keyboard with some internal memory and buffer capability. When a forward observer (FO) detects a target and wants to send the target information back to the TACFIRE system, a formatted message is created with the necessary information. The message is then burst transmitted back to

TACFIRE. The DMD can also be used to send fire unit update messages back to TACFIRE. This information, on the results of an artillery attack, will then be used to update the status of a target.

2. Variable Format Message Device (VFMED) is a remote TACFIRE computer terminal. It allows unit sites supported by field artillery to have access to the TACFIRE computer and the information it holds. To request and receive information from the TACFIRE computer a request for a message format template is made and sent to TACFIRE. The TACFIRE computer sends back the template via the established communication lines. The VFMED operator then fills in the necessary entries and sends the message back to the TACFIRE computer. If a valid message is sent, TACFIRE will return the requested information or acknowledge the entered data. If the message is created improperly, an error message is returned to the VFMED.

3. Artillery Control Console is colocated with the TACFIRE computer. It has two displays and has all of the capability of the VFMED. Through the console and operator can input new information to TACFIRE, retrieve existing information, and monitor messages which TACFIRE is receiving from other VFMED sites. This console is typically located in a large mobile van. This van also includes a digital plotter map and an electronic tactical plotter. The map and plotter are used to display the current battlefield situation to the battlefield commander. Informatio from these maps are used by commanders to update and revise operation concepts and orders held by TACFIRE.

All information is passed to and from TACFIRE through the highly formatted and rigid template messages. These templates have little flexibility in where and how information can be placed in the specified fields. Operators have difficulty in creating and transmitting properly created messages. In particular, the entry of information through the DMD is nearly impossible during night operations or in a nuclear, biological and chemical (NBC) warfare environment when protective gear is worn.

For this reason, alternative methods of entering data have been investigated. One of these methods is voice data entry. The Army had investigated this methodology early in 1974 when it let a two year contract to Scope Electronics. The two year effort was to produce a voice recognition, voice identification, and voice output system which could be used by the FO and other Army Tactical Data Systems (ARTADS) operators. ARTADS eventually evolved into TACFIRE, but the voice recognition capability was never implemented. Little is known about the results of that contract, but voice recognition equipment was never permanently used. It appears as if the applicability of voice data entry was apparent even in 1974, but the technology was not advanced enough to handle the expectations and requirements of the Army for this application.

VOICE RECOGNITION TECHNOLOGY

The accuracy and capability of voice recognition technology has come a long way in the last ten years. The accuracy and capability of the equipment was demonstrated to numerous Army officers at the Field Artillery System Program Review at Fort Sill, Oklahoma and the Command, Control, and Communication System Program Review at Fort Leavenworth, Kansas. As a result of these demonstrations this research was started. The demonstration pointed out that voice recognition input had some qualities and characteristics which might help solve some of TACFIRE's input problems. Voice input has the following characteristics which make it very attractive for this specific application.

1. It is usually faster and more accurate than normal keyboard entry. A study by Poock (1980) in which voice data entry was compared to manual keyboard entry on a distributed computerized network, showed that voice entry was 18% faster and had 183% fewer errors than the manual approach. This improvement was realized with only 3 hours of training with voice input.

2. In the same study an individual was allowed to work on a secondary task besides entering information and operating the computer network. The people who were using voice accomplished 25% more of the secondary task than the people who were using the keyboard. If these results are transferable to the TACFIRE application, it would be possible, especially at the Artillery Control Console, for operators to accomplish other tasks while performing their primary task at the console. There are numerous secondary tasks to choose from since it is a fairly busy command center.

3. The operator is not tied to the keyboard with voice entry. He has the ability to move around, read information off a map, and enter data

from accross the room. In the case of the FO, no keyboard would be necessary if the equipment worked properly.

4. There is no language restriction with voice entry. This is a very attractive feature because TACFIRE will be entering the European theatre shortly. It would be possilbe for allied personnel to input information into TACFIRE in their own language and have the output of the recognition system be the English commands required to run TACFIRE.

5. The voice equipment can be used easily in low light or dark conditions. For TACFIRE operators this is very important especially during night operations.

6. The voice equipment allows for free use of the hands and eyes. This can be of great benefit in numerous situations encountered in the TACFIRE application. The majority of the people available as TACFIRE operators are not required to have typing skills; therefore, they need to usually watch the keyboard as they input information. Furthermore, the FO is busy carrying equipment (gun, binoculars, etc.) as well as watching the intended target. The FO can not readily use his hands and eyes to prepare and transmit the target data.

All of these characteristics of input to a tactical fire direction computer existed 8 years ago. The advantages of voice entry were realized then as much as they are realized today. The reason it was not effective and could not be implemented operationally eight years ago was because of the extreme battlefield conditions under which the equipment would have to be used. The present research effort was undertaken to see if voice input technology had improved enough to be reconsidered for implementation in TACFIRE.

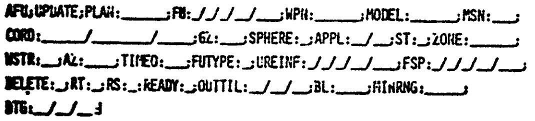
RESEARCH RESULTS

The research study was jointly funded through the Army's High Technology Testbed project at Fort Lewis, Washington and the Training command at Fort Leavenworth, Kansas. It consisted of three phases. The first phase was to write a simulation program on a microcomputer with a built in voice recognition board. This program simulated the majority of the input capabilities of the TACFIRE system. It was primarily used for vocabulary development, initial capability experimentation, and demonstration purposes. Phase 2 interfaced a voice recognition system to an actual TACFIRE computer system for more detailed applicability experimentation. The final phase consisted of laboratory experimentation to investigate possible user problems within an Army battlefield environment.

Phase 1 - Simulation Model

A simulation model was created in FORTRAN on an Interstate Electronics Corporation VRT101 voice recognition terminal. This terminal is a Heath-Zenith Z89 based microcomputer with a voice recognition board. The simulation was designed to totally simulate the input characteristics of the TACFIRE system. The input system was not exactly duplicated because of an error discovered in the Microsoft Inc. version of the FORTRAN compiler whic was available. The only difference which is observed in the simulated input is the need to refresh the displayed message after an entry is made. TACFIRE does not need to refresh the screen after each input. Other than that small inconvenience the simulation appears to operate exactly as TACFIRE would in an input mode.

This phase of the research immediately indicated some possible problem areas with trying to retrofit voice capability into TACFIRE. Figure 1. is a typical TACFIRE message format. This specific template is used to update a possible fire unit's combat readiness or status. When an operator fills in the blank areas he has complete freedom to move a cursor anywhere on the CRT screen. He has the capability to move the cursor in all four directions, and a tab capability which will move the cursor to the next encountered colon (:). For example, if the name of a fire unit was to be input and the cursor was at the top left of the message template, the operator would depress the down cursor key to bring the cursor to the second line. He would then depress the tab key twice to move it to the colon just after the abbreviation FU for Fire Unit. At this point a series of up to 8 alphanumerics must be input with appropriate spacing for the name of the Fire Unit.

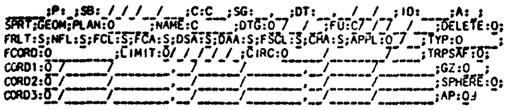


```
_____;P:_;SB:_/_/_/__/___;C:___;SG:_,__;DT:__/__/__/__;ID:____;A:_;
AFU;UPDATE;PLAN:______;FU:_/_/_/__/___;WPN:______;MODEL:______;MSN:___;
CORD:______/_______/_____;GZ:___;SPHERE:_;APPL:_/__;ST:_;ZONE:______;
MSTR:__;AZ:____;TIMEO:___;FUTYPE:_;UREINF:_/_/_/__/___;FSP:_/_/_/__/___;
DELETE:_;RT:_;RS:_:READY:_;OUTTIL:_/__/__;BL:____;MINRNG:_____;
DTG:__/__/__;
```

Fig. 1. TACFIRE message template for updating a fire unit.

The voice entry system would simply have the operator say "fire unit" which would then send the ASCII character strin necessary to position the cursor at the top of the template, move it down the one line, and tab the cursor over to the FU position. Then the operator would say the name of the specific unit desired, and the voice recognition unit would send the appropriate ASCII string with the name of the fire unit.

This system works beautifully and is very impressive when demonstrated. A problem does exist and it was highlighted immediately when developing the simulation. Figure 2. is another TACFIRE message. It too requires the input of a fire unit name, but the ASCII string of reset cursor, move it down one line, and tab twice will not position the cursor at the FU input position. The templates are not uniformly designed. Therefore, if voice data entry is to be used separate output character strings must be programed for each message template. This means software must be built to automatically load the recognition unit with the required output strings when a message template is displayed. In other words, each template will bring in its own vocabulary.



```
_____;P: ;SB: / / / __/ ___;C:C __;SG: , ;DT: , / / ;ID: ___;A: ;
SPRT;GEOM;PLAN:O     ;NAME:C     ;DTG:O 7 /  ;FU:C7 /7 /  ;DELETE:O;
FRLT:S;NFL:S;FCL:S;FCA:S;DSA:S;DAA:S;FSCL:S;CHA:S;APPL:O 7 ;TYP:O    ;
FCORD:O      ;LIMIT:O/ / / / / ;CIRC:O    /      7   ;TRPSAF:O;
CORD1:O 7______7_______,__7_____/_______,__/______7_______;GZ:O ;
CORD2:O /______/_______,__/______/_______,__/______/_______;SPHERE:O;
CORD3:O /______/_______,__/______/_______,__/______/_______;AP:O
```

Fig. 2. TACFIRE message template for entering a geometry plan.

The simulation also amplified some very real advantages to voice entry into TACFIRE. There are numerous fields in TACFIRE messages which can be considered as ON/OFF switches. The normal maunal entry into the system would be to move the cursor to the blank space after the keyword and place an "X" in it. With the voice equipment by saying the name of the field's keyword, the system will position the cursor, place the "X" in the empty field and reset the cursor to the top of the template.

TACFIRE messages also make use of numerically coded input fields. The codes are generally hard to remember or look up when needed. The voice equipment allows the operator to say the unencoded word and then it will output the numeric code expected by TACFIRE.

Problem areas noted with the use of the voice data entry system during the simulation phase included the need for continuous digit input especially in the coordinate fields. It is a relatively slow process to input a long string of digit with a discrete word recognizer. Furthermore, it is possible to have a misrecognition which could erase a portion of the template. This problem has already been noted by operators while using keyboard entry.

Phase 2 - TACFIRE Interface

Phase 2 of the research was to take a voice recognizer and connect to an operational TACFIRE computer. A Threshold Technology model 600 voice recognition unit was taken to Fort Sill on three occasions. An interface device had to be developed to give the user the capability to operate all of the keus on the TACFIRE keyboard by voice. The cursor control keys were not ASCII characters; therefore, the interface created was a translator between the Threshold unit and TACFIRE.

The Threshold unit would output a special ASCII character when a cursor movement was desired and the interface would translate that into the appropriate electrical signal expected by TACFIRE. The interface worked fine and all keys on the keyboard could be accessed through voice commands. Unfortunately, some of the cursor controls such as cursor reset required a 200 millisecond delay between electrical signals. The voice recognition equipment could not be slowed down this much. While TACFIRE was waiting the 200 milliseconds before looking for another signal, the voice unit was continuing to send the next ASCII character. It was not possible to devise a method to have the recognition unit stop sending characters during this time period.

The only operational solution to this problem was to place two null ASCII characters after each of the cursor commands which required this delay. This solution worked fine, but caused further problems. The output character strings are limited in length. Many of the vocabulary words developed in the simulation phase could not be implemented. It is relatively easy to have the manufacturers of the voice equipment increase the output capability. This is viewed as an inappropriate way to solve the problem. What is needed is a way to eliminate the need for an interface.

One other interesting phenomena which became apparent while trying to interface the equipment was a built in error correction function. TACFIRE assumes that no one can type faster than one character every 100 milliseconds. If TACFIRE receives two or more characters in a shorter time period, it will ignore the second or subsequent keystrokes. It was designed to eliminate some typing errors. This meant that the voice equipment could not send characters any faster than 110 baud. Anything faster was assumed to be an error by TACFIRE. Therefore, the voice output could only "type" as fast as the fastest possible person on a keyboard. Speed was one of the major characteristics which made voice a nice alternative to manual keyboard entry.

Phase 3 - Laboratory Experimentation

The final phase of the research effort was to conduct a series of laboratory experiments. These experiments were designed to determine if the voice recognition equipment could maintain its accuracy levels under varied operational battlefield conditions. There are complete reports written on each of these experiments, but a quick summary will be discussed here.

The first experiment conducted concerned the use of the voice equipment with protective masks. Two types of masks were tested. The first mask is known as a stenographer's mask and is being tested by the Army for use by personnel operating close to enemy positions. It is intended to muffle the voice while engaged in radio communication. The second device was a standard gas mask.

The experiment showed that if a user was experienced in the use of masks in general it was possible to keep accuracy rates at an acceptably high level. For the steno mask the error rate for experienced users went up to 2% while unexperienced users had an error rate of about 5%. Overall the users had a 1.8% error rate without the steno mask. The gas mask results were not quite so promising. Under the best conditions the error rate for the gas mask condition was 9%. These experiments did not test the equipment under stressful situations or while the subjects were participating in strenuous exercise. It is expected that further experiments will show that increased respiration due to hard work will cause a poorer accuracy rate in the gas mask condition.

The second experiment conducted looked at the present capability of making the recognition system group independent. It was previously mentioned that the equipment used was a user dependent system. This means that a user must train the system to recognize his voice. This could prove to be a problem if someone was killed and there was little time to load another individuals voice patterns. This experiment was very successful and showed that a group of up to five people could have their patterns resident in the voice processor at the same time. The increase in error rate was very slight. The only problem encountered was that this method reduced the possible vocabulary size down to a level of about 50 or 60 words.

Two other experiments are presently being conducted. The first is another method to investigate group independence but maintain a capability of 250 words. The second experiment is designed to determine if there will be any degrade in recognition performance due to the amount of feedback an individual receives from the recognizer. This experiment is needed to help determine where a recognition unit should be placed. It might not be feasible to place the voice recognition unit in the field next to the user. In this manner the user will not have certain formas of feedback available on the status of a recognized word. It is of interest to determine whether this has an effect on the recognition accuracy.

CONCLUSIONS

The following items can be concluded from the research effort.

1. It is not efficient to retrofit the current version of TACFIRE.

2. The state of the art of voice recognition is still not advanced enough to be used by the FO.

3. Voice recognition definitely will be useful at the Artillery Command Console.

4. More research, especially in the area of stress, is needed before a final decision can be made about the use of voice recognition equipment in a battlefield environment.

5. If the decision to use voice is made, it should be built in and planned for as part of the total system design. This is to allow for uniform templating and to make maximum use of the voice recognition characteristics.

BIBLIOGRAPHY

Poock, G. K. Naval Postgraduate School report NPS55-80-016.

ACCURATE ACOUSTIC PULSE GENERATION

J. Douglas Birdwell

Department of Electrical Engineering, The University of Tennessee,
Knoxville, TN 37996-2100

Abstract. Initial results in the application of modern control techniques to the improvement of the quality of acoustic imaging systems are described. A feedback control law based on the LQG tracking regulator is implemented in discrete time to sequentially generate a desired pressure signature from a high frequency acoustic projector. The specialized hardware, the algorithm, and initial tests on an equivalent circuit are described. A discussion of problems encountered in the implementation is included.

INTRODUCTION

A feedback technique for interatively constructing a desired response from a linear system is applied to acoustic imaging to increase image resolution. A resolution of 0.1mm is required in the image of an object 2-4cm below the surface. This resolution requires a pulse width of approximately 200ns with no ringing. Delays in the return signal can be as large as 8μs.

The input to the acoustic projector is synthesized digitally. Since the algorithm requires measurements of the return signal and corrects the input waveform accordingly, 10 samples are used across the 200ns pulse. The data rate requirement is set at 50MHz, or 20ns/sample. From computer simulations of the algorithm, an 8 bit resolution has been determined sufficient. This requirement is necessary for the dynamic range rather than for absolute accuracy, since the maximum pulse height is determined by the dynamic range.

The algorithm has been applied to audio frequency waveforms (Mazzola, Birdwell and Athans, 1979, 1981); however, the equipment was not interfaced directly to the computer. In addition, several problems have been observed which were not significant at the much lower data rates of the previous work. A complete description of the initial results of this research was published by Birdwell and colleagues (1982).

DESIGN PHILOSOPHY

A feedback control law based on the linear quadratic Gaussian (LQG) tracking regulator is implemented to sequentially generate a desired waveform. Since the algorithm is closed-loop, it can correct for some types of unmodeled errors. Alternatives, such as the FFT, are open loop and must rely on the accuracy of the projector model.

Most of the calculations necessary for the implementation are done offline on a large time-sharing computer and transferred to the local microcomputer over an asynchronous communications line. The local computer implements the control loop, which consists of an optimal filter and the LQ tracker, and controls the specialized hardware using an IEEE-488 instrument bus.

The controlled impulse generator (CIG) is designed to meet the extraordinarily high data rates required by this application. It produces a waveform programmed by the computer and samples the return signal after a programmed delay.

CONCLUSIONS

The principle difficulties encountered in the closed-loop implementation evolve from the algorithm's sensitivity to delay in the control loop, modeling errors in the loop gain, and DC offset in the analog system. A delay in the loop of a regulator can cause oscillations and instability. We have found that delays of 10ns can cause oscillations, and a 20ns delay can cause instability. Although the CIG has a time resolution of 1.25ns on its programmable delay, there is a constant, unknown delay in the sampler. It has proven quite difficult to measure this delay to an accuracy better than ±5ns. Errors in the loop gain and DC offset result in a linearly increasing offset in the closed loop input signal. As long as the digital portion of the CIG does not saturate, this offset can be subtracted from the input signal. Since the projector has two zeros at the origin of the complex plane, it is insensitive to constant or ramp inputs, so this procedure has no effect on the output waveform. Most of the problems with offset and gain errors have been eliminated by the implementation of a computerized calibration procedure on the microcomputer. This software estimates the gain, offset, and delay and stores the values on a disk for reference by the other software.

ACKNOWLEDGEMENTS

EDO Corporation is building this medical instrument for Bowman Gray School of Medicine under NIH Contract NO1-HV-12902. EG&G Oak Rige Operations has designed the CIG Controller for EDO. J. D. Birdwell is the EDO Consultant on Algorithm Development.

The author gratefully acknowledges the contributions of colleagues at EG&G Oak Ridge Operations. Dr. T. J. Paulus directed the development of the CIG. Mr. C. W. Britton contributed to the design of the Sampling Unit in the CIG, and Mr. D. Cole contributed to the design of the CIG Controller. Both made valuable contributions to the prototype system integration and test.

REFERENCES

Birdwell, J. D., T. J. Paulus, C. J. Mazzola and L. Czapla (1982). On the generation of accurate high frequency acoustic pulses using modern control theory. *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*., 323-326.

Mazzola, C. J., J. D. Birdwell and M. Athans (1979). On the application of modern control theory to improving the fidelity of an underwater projector. *J. Acoust. Soc. Am.*, *66*, 739-750.

Mazzola, C. J., J. D. Birdwell and M. Athans (1981). Control of radiated pressure using state variable feedback. In L. Bjorno (Ed.), *Underwater Acoustics and Signal Processing*. D. Reidel, Dordrecht, Holland. pp. 281-284.

List of Attendees of the MIT/ONR C^3 Workshop

Jeff Abram
Advanced Information & Decision Systems
201 San Antonio Circle #286
Mountain View, CA 9404

Thomas A. Adams
Naval Ocean Systems Center
271 Catalina Blvd.
Code 7134
San Diego, CA 92152

David S. Alberts
Special Asst to Vice President
& General Manager
The MITRE Corporation
1820 Dolley Madison Blvd.
McLean, VA 22102

Michael Athans
35-406/LIDS
Department of Electrical Engineering
Massachusetts Institute of Technology
77 Massachusetts Avenue
Cambridge, MA 02139

Daniel A. Atkinson
CTEC, Inc.
7777 Leesburg Pike
Falls Church, VA 22043

Reidar Avestad
Naval Postgraduate School
Monterey, CA 93940

A. V. Balakrishnan
School of Eng. and Applied Science
University of California
Los Angeles, CA 90024

Tamer Basar
Coordinated Science Laboratory
University of Illinois
Urbana, IL 61801

Chip Barnett
Applied Physics Laboratory
Johns Hopkins University
Johns Hopkins Road
Laurel, MD 20810

Glenn Barrowman
Naval Postgraduate School
Monterey, CA 93940

Martin G. Bello
ALPHATECH, Inc.
3 New England Executive Park
Burlington, MA 01803

J. Douglas Birdwell
The University of Tennessee
College of Engineering
Dept. of EE
Knoxville, Tenn 37916

Robert Bliss
Defense Systems Inc.
6804 Poplar Place
McLean, VA 22102

K. L. Boettcher
35-401/LIDS
Massachusetts Institute of Technology
77 Massachusetts Avenue
Cambridge, MA 02139

William Bond
Lockheed Missile & Space Co.
Building 529
Dept. T110
Sunnyvale, CA 94086

Paul L. Bongiovanni
Naval Underwater Systems Center
Code 3521 Bldg. 1171-2
Newport, RI 02840

Barbara J. Bowyer
Naval Postgraduate School
Monterey, CA 93940

Lt. Col. Paul Bragaw
Deputy Under Secretary
of Defense (Policy)
Directorate for Command
and Control Policy
The PENTAGON
2C252
Washington D.C. 20301

Edward C. Brady
The MITRE Corportaion
1820 Dolley Madison Blvd.
McLean, VA 22102

Stuart L. Brodsky
Dept. of the Navy
Office of Naval Research
Arlington, VA 22217

D. T. Bucchioni, Jr.
Defense Systems, Inc.
6804 Poplar Place
McLean, VA 22102

Rex V. Brown
Decision Science Consortium, Inc.
7700 Leesburg Pike
Falls Church, Va 22043

Rudolph C. Brown, Sr.
Westinghouse Electric Corp.
P. O. Box 746, MS-434
Baltimore, MD 21203

B. H. Browne, Jr.
Sperry Aerospace
P. O. Box 4648
Clearwater, Florida 33518

Hal H. Burke
Combat Systems Div. (DRX54-CS)
Army Mateiral Systems
Analysis Agency
Aberdeen Proving Ground, MD 21005

Alfonso Calero
Naval Postgraduate School
Monterey, CA 93940

Monti Callero
Rand Corporation
1700 Main Street
Santa Monica, CA 94061

David Castanon
ALPHATECH, Inc.
3 New England Executive Park
Burlington, MA 01803

Michael D. Carregio
Naval Postgraduate School
Monterey, CA 93940

Nicholas Carriero
US Army HEL
Aberdeen Proving Ground
Maryland 21005

C. Y. Chong
Advanced Information & Decision Systems
Suite 286
201 San Antonio Circle
Mountain View, CA 94040

Mr. J. N. Clare
SCICON Consultancy International Ltd.
Sanderson House
49-57 Berners Street
London W1P 4AQ

Gerald Clapp
Naval Ocean Systems Center
Code 8105
San Diego, CA 92152

George W. Conner
Naval Postgraduate School
Monterey, CA 93940

Norman Crane
Naval Postgraduate School
Monterey, CA 93940

Jose B. Cruz, Jr.
Coordinated Science Laboratory
University of Illinois
1101 W. Springfield Avenue
Urbana, Illinois 61801

Francis C. Deckelman
Naval Electronic Systems Command
ATTN: 6131
Washington D.C. 20360

Richard Diamond
Naval Postgraduate School
Monterey, CA 93940

Robin Dillard
Naval Ocean Systems Center
Code 824
San Diego, CA 92152

Elizabeth R. Ducot
Laboratory for Information and
Decision Systems
Massachusetts Institute of Technology
Rm 35-410A
Cambridge, MA 02139

Donald Edmonds
The MITRE Corporation
1820 Dolley Madison Blvd.
McLean, VA 22102

Glen F. Filley
Naval Postgraduate School
Monterey, CA 93940

R. N. Forrest
Naval Postgraduate School
Monterey, CA 93940

R. Fratila
ELEX-330
Naval Electronics Systems Command
Washington DC 20360

Kenneth Frank
Naval Postgraduate School
Monterey, CA 93940

Stephen H. Goetchins
Naval Postgraduate School
Monterey, CA 93940

I. R. Goodman
Naval Ocean Systems Center
Code 7232
Bayside Bldg. 128 Rm 122
San Diego, CA 92152

Richard B. Grahlman
Naval Postgraduate School
Monterey, CA 93940

Geoffrey Grant
Naval Postgraduate School
Monterey, CA 93940

Robert W. Grayson
The MITRE Corporation
1820 Dolley Madison Blvd.
McLean, VA 22102

Frank Greitzer
Navy Personnel R&D Center
San Diego, CA 92152

Yacov Haimes
Case Institute of Technology
Dept. of Systems Engineering
Cleveland, Ohio 44106

Stanley M. Halpin
US Army Research Institute
5001 Eisenhower Avenue
Alexandria, VA 22333

Bruce W. Hamill
Johns Hopkins University
Applied Physics Laboratory
Johns Hopkins Road
Laurel, Maryland 20707

James Hartman
Naval Postgraduate School
Monterey, CA 93940

Richard W. Heathcote
Naval Postgraduate School
Monterey, CA 93940

William D. Helling
Naval Postgraduate School
Monterey, CA 93940

Ray L. Hershman
Navy Personnel R&D Center
Code P305
San Diego, CA 92152

Charles Holland
Office of Naval Research
Code 411MA
800 North Quincy Street
Arlington, VA 22217

William Hopper
Naval Postgraduate School
Monterey, CA 93940

Scott Horning
Naval Postgraduate School
Monterey, CA 93940

Lt. Cdr. Huff
Commander, Naval Oceanography
NSTL Station
BAY Saint Louis, Mississippi 39529

Cdr. Kent S. Hull, USN
Deputy Director, Mathematical &
Information Sciences
Office of Naval Research
Code 430B
800 N. Quincy
Arlington, VA 22217

Carolyn Hutchinson
Comptek Research Inc.
10731 Treena Street
Suite 200
San Diego, CA 92131

Paul Jensen
ESL Inc.
495 Java Drive
Sunnyvale, CA 94086

Stephen Kahne
Case Institute of Technology
Dept. of Systems Engineering
Cleveland, Ohio 44106

Donald Kirk
Naval Postgraduate School
Monterey, CA 93940

Albert C. Knoblock
Code 2012
Naval Air Development Center
Warminster, PA 18974

Michael Kovacich
Comptek Research Inc.
Mare Island Dept.
P. O. Box 2134
Vallejo, CA 94592

Timothy Kraft
Comptek Research Inc.
10731 Treena Street
Suite 200
San Diego, CA 92132

Leslie C. Kramer
ALPHATECH, Inc.
3 New England Executive Park
Burlington, MA 01803

Alfred F. Krochmal
Naval Electronic System Command
Code ELEX 621
Washington DC 20360

Richard T. Lacoss
MIT Lincoln Laboratory
P. O. Box 73
Lexington, MA 02173

Joel S. Lawson, Jr.
Naval Electronic Systems Command
Washington, D.C. 20360

Barry Leiner
I.P.T.O.
Room 730
DARPA
1400 Wilson Blvd.
Arlington, VA 22209

Dennis Lenahan
Naval Postgraduate School
Monterey, CA 93940

Wendy Lenahan
Naval Postgraduate School
Monterey, CA 93940

Dan Leonard
Naval Ocean Systems Center
Code 8105
San Diego, CA 92152

Edward A. Lenio
Naval Postgraduate School
Monterey, CA 93940

Alexander H. Levis
Laboratory for Information
and Decision Systems
Massachusetts Institute of Technology
Rm 35-410B
Cambridge, MA 02139

Victor O.K. Li
Dept. of EE & Systems
PHE 526
University of Southern California
Los Angeles, CA 90007

Allan Lied
Naval Postgraduate School
Monterey, CA 93940

Glenn R. Linsenmayer
Westinghouse Electric Corporation
P. O. Box 746
MS 434
Baltimore, MD 21203

Peter Luh
Dept. of EE&CS
University of Connecticut
Box -U- 152
Storrs, CT 06268

Kenneth Loparo
Dept. of Systems Engr.
Case Western Reserve University
Crawford 612
Cleveland, Ohio 44106

John Machado
ELEX-330
Command and Control Division
Naval Electronic System Command
Washington DC 20360

Daryl Marsh
Naval Ocean Systems Center
San Diego, CA 92152

Dennis McCall
Naval Ocean Systems Center
Code 8242
San Diego, CA 92152

Charles McCorkle
Naval Postgraduate School
Monterey, CA 93940

Charles McKeone
Intel Division
D025 MCDEC
Quantico, VA 22134

Koeing McLeron
Naval Postgraduate School
Monterey, CA 93940

Michael Melich
Code 7577
Naval Research Laboratory
Washington, D.C. 20375

Carl Menyhert
Naval Postgraduate School
Monterey, CA 93940

Admiral William Meyers
9621 Ceralene Drive
Fairfax, VA 22032

H. G. Miller
Code 8302
Naval Ocean Systems Center
San Diego, CA 92152

Glenn E. Mitzel
Johns Hopkins University
Applied Physics Laboratory
Johns Hopkins Road
Laurel, MD 20810

Michael H. Moore
Systems Development Corporation
4025 Hancock Street
San Diego, CA 92037

Paul H. Moose
Code 62 ME
Naval Postgraduate School
Monterey, CA 93940

Martin Morf
Stanford University
Durand 109
Stanford,, CA 94305

Peter Morgan
SCICON Consultancy International
49-57 Berners Street
London W1P 4AQ
United Kingdom

Douglas Neil
Naval Postgraduate School
Monterey, CA 93940

Bruce Noel
The MITRE Corporation
1820 Dolley Madison Blvd.
McLean, VA 22102

Craig E. Opel
Naval Postgraduate School
Monterey, CA 93940

Cdr. James Offutt
Office of Naval Research
1030 East Green Street
Pasadena, CA 91106

Roland Payne
Advanced Information & Decisions Sys.
201 San Antonio Circle #286
Mountain View, CA 94040

Wayne Perras
Naval Postgraduate School
Monterey, CA 93940

Lloyd S. Peters
Center for Defense Analysis
SRI International
333 Ravenswood Avenue
Menlo Park, CA 94025

Chester Phillips
Johns Hopkins University
Applied Physics Laboratory
Johns Hopkins Road
Laurel, MD 20810

Robert Phillips
Naval Postgraduate School
Monterey, CA 93940

Charles R. Poppe
4502 Carley Lane
Alexandria, VA 22309

David Porter
Business & Tech. Sys. Inc.
10210 Greenbelt Road
Seabrook, MD 20801

Harilaos Psaraftis
Marine Systems
Massachusetts Institute of Technology
Cambridge, MA 02139

Barry L. Reichard
US Army Ballistic Research Lab
ATTN: DRDAR-BLB
Aberdeen Proving Ground, MD 21014

Russell Robinson
Naval Postgraduate School
Monterey, CA 93940

Joseph Rockmore
Systems Control Technology, Inc.
1801 Page Mill Road
Palo Alto, CA 94340

Ronald J. Roland
Rolands & Associates Corp.
500 Sloate Avenue
Monterey, CA 93940

Ellen F. Roland
Rolands & Associates Corp.
500 Sloate Avenue
Monterey, CA 93940

Thomas P. Rona
Boeing Aerospace Company
MS84-56 Box 3999
Seattle, Washington 98124

John Russel
Naval Postgraduate School
Monterey, CA 93940

Vivek S. Samant
ORINCON Corporation
33656 N. Torrey Pines Court
LaJolla, CA 92037

Daniel Schutzer
Naval Intelligence
Chief of Naval Operations
NOP 009T
Washington, D.C. 20350

Edward J. Shanahan, Jr.
Georgia Institute of Technology
EES/ECSL/CCB-Room 209-ERB
Atlanta, Georgia 30332

Barbara Sherlock
Naval Postgraduate School
Monterey, Ca 93940

Glen Simon
Naval Postgraduate School
Monterey, CA 93940

James R. Slagle
Code 7510
4555 Overlook Avenue SW
Washington, D.C. 20375

Dana Small
Naval Ocean Systems Center
San Diego, CA 92152

James L. Smith, Jr.
5464 Black Oak Way
San Jose, CA 95129

Lt. Cdr. Lary Smith
Navy Tactical Interoperability
Support Activity
200 Catalina Blvd.
San Diego, CA 92145

James G. Smith
Lockheed Missiles & Systems
Sunnyvale, CA

Michael Sovereign
Naval Postgraduate School
Monterey, CA 93940

Stuart H. Starr
DUSD(C3I),OSD
Rm 3E182
The PENTAGON
Washington, D.C. 20301

Gene Steffanetta
Naval Postgraduate School
Monterey, CA 93940

Joseph Swingle
Naval Postgraduate School
Monterey, CA 93940

Julian E. Teske
Naval Postgraduate School
Monterey, CA 93940

William Thomas
Naval Postgraduate School
Monterey, CA 93940

Joel Trimble
Code 221
Office of Naval Research
800 N Quincy
Arlington, VA 22217

Edison Tse
Dept. of Engineering-
Economic Systems
Stanford University
Stanford, CA 94305

Richard M. Tong
Advanced Information & Decision Systems
201 San Antonio Circle
Suite 286
Montain View, CA 94040

Donald Van Arman
The MITRE Corporation
1820 Dolley Madison Blvd.
McLean, VA 22102

William VanHoy
Naval Postgraduate School
Monterey, CA 93940

Harry L. VanTrees
Linkabit Corporation
7927 Jones Branch Drive
McLean, VA 22102

Willard Vaughn
Naval Postgraduate School
Monterey, CA 93940

Maniel Vineberg
Naval Ocean Systems Center
Code 8112
San Diego, CA 92152

Joseph H. Wack
Westinghouse Electric Corp.
P. O. Box 746 MS-237
Baltimore, MD 21203

Sharon Ward
AFAMRL/HEC
Dept. of the Air Force
Wright-Patterson Air Force Base
Ohio 45433

Professor Washburn
Naval Postgraduate School
Monterey, CA 93940

John Welt
Naval Postgraduate School
Monterey, CA 93940

Anne Werkheiser
Software Engr. Applied Section
Code 7595
Naval Research Laboratory
Washington, D.C. 20375

Jeffrey E. Wieselthier
Naval Research Laboratory
Code 7521
Washington, D.C. 20375

James Wilson
Naval Postgraduate School
Monterey, CA 93940

Richard P. Wishner
Advanced Information & Decision Systems
201 San Antonio Circle
Suite 286
Mountain View, CA 94040

K. E. Woehler
Naval Postgraduate School
Monterey, CA 93940

Joseph G. Wohl
ALPHATECH, Inc.
3 New England Executive Park
Burlington, MA 01803

John M. Wozencraft
Naval Postgraduate School
Code 74
Monterey, CA 93940

William Wren
Naval Postgraduate School
Monterey, CA 93940

Howard Yellen
Naval Postgraduate School
Monterey, CA 93940

Wayne W. Zachary
ANALYTICS
2500 Maryland Road
Willow Grove, PA 19090

Lotfi A. Zadeh
University of California
Computer Science Division
Berkeley, CA 94720

5TH MIT/ONR WORKSHOP ON C^3 SYSTEMS
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA

AUGUST 23 to AUGUST 27, 1982

FINAL PROGRAM
(AS OF JUNE 15, 1982)

MONDAY MORNING, AUGUST 23, 1982

<u>SESSION 1</u>: INGERSOLL AUDITORIUM 120: C^3 SYSTEMS

<u>Chairman</u>: *Professor Michael Athans, MIT*

8:00 - 8:30 A.M.	REGISTRATION
8:30 - 9:00 A.M.	INTRODUCTION AND WELCOME *Prof. M. Athans, MIT* *Dr. S. Brodsky, Office of Naval Research*
9:00 - 9:30 A.M.	CONCEPTS AND THOUGHTS CONCERNING A CONTROL STRATEGY FOR CONDUCTING INFORMATION WARFARE *Dr. D. Schutzer, Office of Naval Intelligence*
9:30 - 10:00 A.M.	ACCURACY STUDIES IN A DISTRIBUTED SURVEILLANCE SYSTEM USING ACOUSTIC SENSORS *Drs. T. Kurien, D.A. Castanon, and L.C. Kramer, ALPHATECH, Inc.*
10:00 - 10:30 A.M.	BREAK
10:30 - 11:00 A.M.	ACE(Artillery Control Experiment) FIRE SUPPORT CONTROL SIMULATOR TECHNOLOGY *Drs. H. Walter and B. Reichard, Army Ballistics Research Labs*
11:00 - 11:30 A.M.	EVALUATING C^3 SYSTEM SURVIVABILITY BASED ON RELIABILITY AND NETWORK ANALYSIS THEORY *Drs. D. R. Edmonds and G. Emami, MITRE*
11:30 - 12:15 P.M.	DISCUSSION
12:15 - 1:30 P.M.	LUNCH

MONDAY AFTERNOON, AUGUST 23, 1982

<u>SESSION 2</u>: INGERSOLL AUDITORIUM 120: SURVEILLANCE

<u>Chairman</u>: *Professor Robert R. Tenney, MIT*

1:30 - 2:00 P.M.	ISSUES AND RESULTS IN DISTRIBUTED DETECTION *Drs. G. Lauer, D. Teneketzis, and N. Sandell, Jr. ALPHATECH, Inc.*
2:00 - 2:30 P.M.	DISTRIBUTED DETECTION WITH LIMITED COMMUNICATIONS *Prof. R. Tenney and Dr. L. Eckhian, MIT/LIDS*
2:30 - 3:00 P.M.	A SUMMARY OF RESULTS IN MULTIPLATFORM CORRELATION AND INTERSHIP ALIGNMENT IN THE NAVAL BATTLEGROUP *Dr. M. Kovacich, COMPTEK*

3:00 - 3:30 P.M.	BREAK
3:30 - 4:00 P.M.	SENSOR SCHEDULING *Prof. R. Tenney, MIT/LIDS*
4:00 - 4:30 P.M.	SURVEILLANCE TRACKERS: SIMILITUDE AND PHYSICS *Dr. A. D. Atkinson, Applied Physics Laboratory*
4:30 - 5:00 P.M.	ACOUSTIC SENSOR NETWORK *Dr. R. Lacoss, MIT/Lincoln Laboratories*

MONDAY AFTERNOON, AUGUST 23, 1982

SESSION 3: SPANGLE 100A: SYSTEM METHODOLOGIES (I)

Chairman: *Dr. Alexander H. Levis, MIT*

1:30 - 2:00 P.M.	EVOLUTION ACQUISITION AND EVALUATION OF C^2 SYSTEMS *Dr. D. S. Alberts, MITRE*
2:00 - 2:30 P.M.	FUNCTION MAINTENANCE IN HIERARCHICAL COMMAND STRUCTURES *Prof. S. Kahne, Case Institute of Technology*
2:30 - 3:00 P.M.	A TOTAL SYSTEM APPROACH TO COMMAND SYSTEM DESIGN *Mr. M. D. Morgan, SCICON*
3:00 - 3:30 P.M.	BREAK
3:30 - 4:00 P.M.	HIERARCHICAL-MULTIOBJECTIVE FRAMEWORK FOR C^3 SYSTEMS *Prof. Y. Haimes, Case Institute of Technology*
4:00 - 4:30 P.M.	VOICE RECOGNITION INPUT TO THE ARMY's TACTICAL FIRE DIRECTION SYSTEM (TACFIRE) *Dr. G. K. Poock, Naval Postgraduate School* *Ms. E. F. Roland, Rolands & Associates*
4:30 - 5:00 P.M.	AN APPROACH TOWARDS EVOLUTIONARY DEVELOPMENT OF COMMAND AND CONTROL SYSTEMS *Dr. E.J. Shanahan, Jr., Georgia Institute of Technology*

TUESDAY MORNING, AUGUST 24, 1982

SESSION 4: INGERSOLL AUDITORIUM 120: HUMAN FACTORS

Chairman: *Professor Michael Athans, MIT*

8:30 - 9:00 A.M.	DECISIONS AND DATA BASES *Dr. J. L. Lawson, Naval Electronics Systems Command*
9:00 - 9:30 A.M.	FUNDAMENTAL ISSUES IN DISTRIBUTED DECISIONMAKING *Prof. M. Athans and J. N. Tsitsiklis, MIT/LIDS*
9:30 - 10.00 A.M.	PRESCRIPTIVE ORGANIZATION THEORY IN THE CONTEXT OF SUBMARINE COMBAT SYSTEMS *Dr. R. V. Brown, Decision Science Consortium*

10:00 - 10:30 A.M. BREAK

10:30 - 11:00 A.M. MEASURING THE HUMAN FACTORS IMPACT OF COMMAND AND CONTROL DECISION AIDS

Dr. W. W. Zachary, Analytics, Inc.
Dr. J. Hopson, Naval Air Development Center

11:00 - 11:30 A.M. DETECTING A CHANGE IN TARGET LOCATION: A COMPARISON OF HUMAN AND OPTIMAL PERFORMANCE

Drs. R. L. Hersham and F. L. Greitzer
Navy Personnel Research & Development Center

11:30 - 12:15 P.M. DISCUSSION

12:15 - 1:30 P.M. LUNCH

TUESDAY AFTERNOON, AUGUST 24, 1982,

SESSION 5: INGERSOLL AUDITORIUM 120: TRACKING AND ESTIMATION

Chairman: *Professor Robert R. Tenney, MIT*

1:30 - 2:00 P.M. ESTIMATION AND CONTROL THEORY APPROACHES TO DETERMINE SENSOR CONTROL STRATEGIES IN A MULTI-SENSOR/MULTI-TARGET ENVIRONMENT

Mr. V. S. Samant, ORINCON

2:00 - 2:30 P.M. AN APPROACH TO THE DATA ASSOCIATION AND TRACKING PROBLEM THROUGH POSSIBILITY THEORY

Dr. I. R. Goodman, NOSC

2:30 - 3:00 P.M. DATA FUSION FOR NONLINEAR SYSTEMS

Dr. D. A. Castanon, ALPHATECH

3:00 - 3:30 P.M. BREAK

3:30 - 4:00 P.M. MINIMAL TIME DETECTION ALGORITHM FOR TARGET MANUEVER

Prof. A. V. Balakrishnan, UCLA

4:00 - 4:30 P.M. OPTIMAL PLATFORM MANEUVERS FOR PASSIVE TRACKING WITH TIME DELAY MEASUREMENTS

Prof. P. T. Liu, University of Rhode Island
Dr. P. L. Bongiovanni, Naval Underwater Systems Center

4:30 - 5:00 P.M. MANEUVERING TARGETS AND GUN FIRE CONTROL SOLUTIONS

Mr. H. Burke, Army Material Systems Analysis Agency

TUESDAY AFTERNOON, AUGUST 24, 1982

SESSION 6: SPANGLE 100A: ARTIFICIAL INTELLIGENCE AND DECISION AIDS

Chairman: *Ms. Elizabeth R. Ducot, MIT*

1:30 - 2:00 P.M. EXPERT SYSTEMS TO AID TACTICAL AIR FORCE

Drs. L. Hempshill, R. A. Riemenschneider, and A. J. Rockmore
Systems Control Technology, Inc.

2:00 - 2:30 P.M. DECISION PROCESSING IN A DISIMINATION SYSTEM FOR MILITARY MESSAGES

Drs. A. Werkheiser, H. Froscher, and J. Bachenko
Naval Research Laboratory

2:30 - 3:00 P.M.	BATTLE: AN EXPERT DECISION AIR FOR FIRE SUPPORT COMMAND AND CONTROL *Drs. J. R. Slagle, R. R. Cantone, and E. Halpern* *Naval Research Laboratory*
3:00 - 3:30 P.M.	BREAK
3:30 - 4:00 P.M.	THE DEMPSTER-SHAFER THEORY APPLIED TO TACTICAL DATA FUSION ON AN INFERENCE SYSTEM *Dr. R. A. Dillard, NOSC*
4:00 - 4:30 P.M.	AN ALGORITHM FOR AUTOMATICALLY CONTROLLING A ROBOTIC FIGHTER AIRCRAFT *Dr. M. H. Moore, System Development Corp.*
4:30 - 5:00 P.M.	PROCESSING OF UNCERTAIN EVIDENCE IN EXPERT SYSTEMS *Prof. L. Zadeh, University of California at Berkeley*

WEDNESDAY MORNING, AUGUST 25, 1982

SESSION 7: INGERSOLL AUDITORIUM 120: AIR DEFENSE

Chairman: *Professor Michael Athans, MIT*

8:30 - 9:00 A.M.	SYNOPSIS OF MISSION-ORIENTED EVALUATION OF THE DEFENSIVE AIR BATTLE *Dr. S. H. Starr, Office of Secretary of Defense, C^3I*
9:00 - 9:30 A.M.	A QUEUEING METHODOLOGY FOR THE ANALYSIS AND MODELING OF AIR-DEFENSE COMMAND AND CONTROL SYSTEMS *Drs. M. G. Bello, J. R. Delaney, and L. C. Kramer* *ALPHATECH*
9:30 - 10:00 A.M.	DYNAMIC MANY-ON-MANY INTERCEPTION STRATEGIES IN AIR DEFENSE SYSTEMS *Prof. M. Athans and Y. L. Chow, MIT/LIDS*
10:00 - 10:30 A.M.	BREAK
10:30 - 11:00 A.M.	DYNAMIC SEQUENCE ASSIGNMENT IN MISSION PLANNING *Drs. V. Rutenburg, R. P. Wishner and J. M. Abram* *Advanced Information & Decision Systems* *Prof. E. Tse, Stanford University*
11:00 - 11:30 A.M.	SUBJECTIVE MEASUREMENT OF C^2 *Monti Callero, Rand Corporation*
11:30 - 12:15 P.M.	DISCUSSION
12:15 - 1:30 P.M.	LUNCH

WEDNESDAY AFTERNOON, AUGUST 25, 1982

SESSION 8: INGERSOLL AUDITORIUM 120: COMMUNICATIONS

Chairman: *Ms. Elizabeth R. Ducot, MIT*

1:30 - 2:00 P.M.	SPREAD SPECTRUM MULTIPLE ACCESS ISSUES IN THE HF INTRA TASK FORCE COMMUNICATION NETWORK *Dr. J. E. Wieselthier, Naval Research Laboratory*

2:00 - 2:30 P.M.	MESSAGE DELAY IN A COMMUNICATION NETWORK WITH FAILING NODES AND LINKS *Prof. O. K. Li, University of Southern California*
2:30 - 3:00 P.M.	MULTIPLE TACTICAL DATA INFORMATION LINK (MULTI-TADIL) INTEROPERABILITY CONCEPT *Capt. B. W. Churchill, USN, (OP-NAV 942G)*
3:00 - 3:30 P.M.	BREAK
3:30 - 4:00 P.M.	OPTIMAL LOCATION OF DISTRIBUTED DATA BASES IN A FAILING COMMUNICATION NETWORK *Prof. M. Athans and M. Ma, MIT/LIDS*
4:00 - 4:30 P.M.	ON THE USE OF A COMPUTER TESTBED IN THE ANALYSIS OF INFORMATION FLOW IN C^2 NETWORKS *Ms. E. R. Ducot, MIT/LIDS*
4:30 - 5:00 P.M.	A GENERIC COMMAND SUPPORT SYSTEM *Dr. M. Vineberg and C. Warner, NOSC*

WEDNESDAY AFTERNNON, AUGUST 25, 1982

SESSION 9: SPANGLE 100A: SYSTEM METHODOLOGIES (II)

Chairman: *Dr. Alexander H. Levis, MIT*

1:30 - 2:00 P.M.	THE ROLE OF TEAMS IN THE ORGANIZATION OF WORK *Mr. J. N. Claire, SCICON*
2:00 - 2:30 P.M.	INFORMATION MANAGEMENT FOR COMMAND AND CONTROL *Mr. P. D. Morgan, SCICON*
2:30 - 3:00 P.M.	C^3 OVER THE PAST 3 TO 5 YEARS - A PERSONAL LEARNING EXPERIENCE *Dr. T. P. Rona, Boeing Aerospace Company*
3:00 - 3:30 P.M.	BREAK
3:30 - 4:00 P.M.	AN APPLICATION BASED DESIGN METHODOLOGY FOR DISTRIBUTED C^3 SYSTEMS *Profs. V. Matula and M. Buchner, Case Institute of Technology*
4:00 - 4:30 P.M.	ACCURATE ACOUSTIC PULSE GENERATION *J. Douglas Birdwell, University of Tennessee*
4:30 - 5:00 P.M.	INFORMATION/CONTROL SELECTION IN STOCHASTIC SYSTEM WITH RANDOM STRUCTURE *Profs. K. Loparo and B. Griffiths, Case Institute of Technology*

THURSDAY MORNING, AUGUST 26, 1982

SESSION 10: INGERSOLL AUDITORIUM 120: C^3 SYSTEMS (II)

Chairman: *Professor Michael Athans, MIT*

8:30 - 9:00 A.M.	SHIPBOARD AUTOMATED TACTICAL SUPPORT SYSTEMS *Capt. B. W. Churchill, USN (OP-NAV 942G)*
9:00 - 9:30 A.M.	OPERATING SYSTEMS FOR TACTICAL DISTRIBUTED PROCESSING NETWORKS *Drs. C.Hutchinson and Timothy Kraft, COMPTEK*
9:30 - 10:00 A.M.	A GAME THEORETIC ANALYSIS OF PLATFORM POSITIONING UNDER A DISTRIBUTED DECISION DOCTRINE *Drs. J.R. Delaney, D.A. Castanon, M. Athans, L.C. Kramer ALPHATECH, Inc.*
10:00 - 10:30 A.M.	BREAK
10:30 - 11:00 A.M.	DECISIONMAKING ORGANIZATIONS WITH ACYCLICAL INFORMATION STRUCTURES *Dr. A. H. Levis and K. L. Boettcher, MIT/LIDS*
11:00 - 11:30 A.M.	MODELING HUMAN DECISIONMAKING PROCESSES IN AN ANTI-SUBMARINE WARFARE ENVIRONMENT *Mr. J. W. Wohl and Dr. E.E. Entin, ALPHATECH, Inc.*
11:30 - 12:15 P.M.	DISCUSSION
12:15 - 1:30 P.M.	LUNCH

THURSDAY AFTERNOON, AUGUST 26, 1982

SESSION 11: SPANGLE 100A: SURVEILLANCE AND COMMUNICATIONS

Chairman: *Professor Robert R. Tenney, MIT*

1:30 - 2:00 P.M.	HUMAN MEMORY LIMITATIONS IN MULTI-OBJECT TRACKING *Drs. F. L. Greitzer, R. L. Hershman, and R. T. Kelley Navy Personnel Research & Development Center*
2:00 - 2:30 P.M.	MYOPIC AND PRESBYOPIC APPROACHES TO A MULTI-SENSOR, MULTI-TARGET RESOURCE ALLOCATION PROBLEM *Prof. H. N. Psaraftis and A. N. Perakis, MIT*
2:30 - 3:00 P.M.	QUATITATIVE MEASURES APPLICABLE TO AN OCEAN SURVEILLANCE SYSTEM SIMULATION *Drs. R. D. Cook, NOSC - R. H. Worsham, Westinghouse*
3:00 - 3:30 P.M.	BREAK
3:30 - 4:00 P.M.	MULTI-OBJECT TRACKING *Prof. R. R. Tenney, MIT/LIDS*
4:00 - 4:30 P.M.	PACKET RADIO IN A TACTICAL ENVIRONMENT *Prof. J. M. Wozencraft, Naval Postgraduate School*
4:30 - 5:00 P.M.	EXTENSIONS OF LANCHESTER'S EQUATIONS *Prof. P. H. Moose, Naval Postgraduate School*

THURSDAY AFTERNOON, AUGUST 26, 1982

SESSION 12: INGERSOLL AUDITORIUM 120: C^2 THEORY

Chairman: *Ms. Elizabeth R. Ducot, MIT*

1:30 - 2:00 P.M.	C^3 SYSTEMS EFFECTIVENESS ANALYSIS *Dr. A. H. Levis, MIT/LIDS*
2:00 - 2:30 P.M.	DECISION SUPPORT SYSTEM FOR ORGANIZATION DECISION-MAKING *Prof. E. Tse, Stanford University* *Dr. R. M. Tong, Advanced Information & Decision Systems*
2:30 - 3:00 P.M.	AN ANALYTICAL MODEL FOR INVESTIGATION OF REFLEXIVE RESPONSE ASMD APPLICATIONS COMMAND AND CONTROL SITUATIONS *Drs. T. Kraft and C. Hutchinson, COMPTEK*
3:00 - 3:30 P.M.	BREAK
3:30 - 4:00 P.M.	MEASURES OF EFFECTIVENESS FOR THE COMMAND/CONTROL OF C^3 COUNTERMEASURES *Drs. R. Cook, NOSC - G. Linsenmayer, Westinghouse*
4:00 - 4:30 P.M.	SPECULATIONS ON C^2 THEORY *Prof. M. Athans, MIT/LIDS*
4:30 - 5:00 P.M.	THREE-LEVEL HIERARCHICAL DECISION PROBLEMS *Professor P. Luh, University of Connecticut*

AUTHORS' INDEX