EFFECT OF ORGANIZATIONAL STRUCTURE ON PERFORMANCE OF DECISION MAKING TEAMS

by

VICTORIA Y. JIN

B.S. in Mechanical Engineering,
University of New Hampshire, 1983

M.S. in Mechanical Engineering,
Massachusetts Institute of Technology, 1985

SUBMITTED TO THE DEPARTMENT OF
MECHANICAL ENGINEERING
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May, 1990

Signature of Author_____
Department of Mechanical Engineering
May, 1990

Certified by _____
Alexander H. Levis
Senior Research Scientist, Laboratory for Information and Decision Systems
Thesis Supervisor

Accepted by_____
Ain Sonin
Chairman, Departmental Committee on Graduate Students
Department of Mechanical Engineering

EFFECT OF ORGANIZATIONAL STRUCTURE ON
PERFORMANCE OF DECISION MAKING TEAMS

by

VICTORIA Y. JIN

Submitted to the Department of Mechanical Engineering
on May 23, 1990
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

ABSTRACT

Distributed decision making (DDM) organizations consist of human decision makers (DMs) and equipment, structured so as to accomplish a set of given tasks. A multi-person, model-driven experiment has been designed to investigate the effect of organizational structure on performance. The experiment has been designed on the basis of an existing mathematical model of interacting DMs and organizational structures. In the experiment, a distributed decision making environment was created. Two organizational structures were used in the investigation: a parallel organization and a hierarchical organization.

The experimental results show that the variation in performance is less between different teams than between different individual DMs within a team. Therefore, organizational performance is more predictable than individual performance. Interaction among DMs in an organization compensates for differences in individual performance characteristics. The main controlled variable in the experiment was the available time for perform a task. Decrease in available time introduced time pressure. The experimental results confirm a hypothesis which predicts that with decreasing available time, a significant degradation of performance occurs first in the organization which has the highest minimum required workload.

These results are consistent with the findings from the theoretical model. Furthermore, the critical value of the ratio of response time to available time for doing a task is an observable measure of the bounded rationality constraint and can be used as a key parameter in organization design.

Thesis Supervisor: Dr. Alexander H. Levis
Title: Senior Research Scientist

To

the memory of my father

Hua-kuang

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to Dr. Alexander H. Levis for his guidance and patience during the years I studied at Massachusetts Institute of Technology. As a scientist and an educator of profound knowledge and exploratory spirit, he made studying with him a valuable experience both professionally and personally.

I would like to thank Professor Derek Rowell and Professor Thomas Sheridan for their advice and comments on the thesis. As thesis committee members, they provided helpful suggestions and clarity of the destination.

Thanks to Mr. Kent Hall for serving as a thesis reader and participating in the thesis committee. His comments and suggestions, specially in the naval outer air battle are greatly acknowledged.

I also wish to thank Professor Ilze Levis for her advice and help in the technical writing aspect , especially for her help on checking the final draft of this thesis.

Thanks to Dr. Jeff Cassey for his comments and suggestions with respect to the human subjects during the experimental design.

Macintosh Tech Group of Boston Computer Society provided great help in answering the questions about the network. Specially, Vicent's help was very much appreciated.

I wish to thank Cindy, Dregan, Ianis, Jim, Keith, Louis, Stamos, and Shumei for their support and friendship during my years in MIT.

Finally, I would like to thank my family for their support in making my studies successful. Especially, I would like to thank my mother for her endless support and courage through my years of study, and My husband George, without whose love and support this thesis would have been impossible.

This research was carried out at the Laboratory for Information and Decision Systems at Massachusetts Institute of Technology with the support of the Office of Naval Research under contract No. N00014-84-K-0519 and the Basic Research Group of the Joint Directors of Laboratories through ONR under contract No. N00014-85-K-0782.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1
INTRODUCTION

1.1 BACKGROUND

Distributed decision making (DDM) organizations consist of human decision makers (DMs) and equipment, structured so as to accomplish a set of given tasks. In the past few years, several research efforts have been started to explore the characteristics of the DDM organizations and related design and evaluation methodologies[1]. However, the design and evaluation of DDM organizations is still a developing subject in the field of engineering.

Because human DMs are integral elements of decision making systems, design and evaluation become more complex than in machine-only systems. The distributed nature of the decision process in DDM organizations results in even more uncertainty in predicting their performance. There is a rich literature in the field of psychology where the behavior of humans as information processors and decision makers has been studied. A major finding relevant to this thesis is the existence of bounded rationality, a constraint on the human's capability to process information and make decisions.

Bounded rationality describes the limitation of human beings as processors of information and as problem solvers. Simon (1956) examined the informational and computational limits of human rationality and suggested that the list of constraints on choice should include properties of human beings as decision makers.

A decision making organization has been considered as an organized anarchy, characterized by its problematic preference, unclear technology, and fluid participation (Cohen, March, and Olsen, 1972). Cohen et al. developed an explicit computer simulation model of a "garbage can" decision process; in it they imply that ambiguity in the organization is introduced mainly by bounded rationality.

Over the past twenty years, a considerable effort has been made to study the concept of bounded rationality and its suggestion that human decision making is limited by the decision maker's cognitive capabilities.

With evidence from studies by experimental psychologists and cognitive scientists, many system scientists and engineers have devoted their efforts on modeling DDM organizations

---

[1] The Distributed Tactical Decision Making program of the Office of Naval Research provided the impetus for such studies.

(Charnes and Cooper, 1963; Kenn, 1977; Greitzer and Hershman, 1984). Many models have been proposed for representing the characteristics of DDM organization with human decision makers. The information processing model used an analogy between communication channels and human behavior in response to stimuli to characterize human information processing behavior (Sheridan and Ferrell, 1974). A four-stage information processing and decision making model of the interacting decision maker with bounded rationality has been developed by Boettcher and Levis (1982). These authors used information theory (Shannon and Weaver, 1949) to characterize the amount of cognitive activity.

In recent years, a design and evaluation methodology has been developed on the basis of the mathematical model by Boettcher and Levis. The design process starts with the generation of organizational structures for a given task. Then, the procedures and protocols needed to perform the task are determined. The task is characterized by the uncertainty that needs to be reduced in order to make decisions. Information theory is applied to compute the entropy of the input, a measure of uncertainty. Then, the performance of the organizational design can be evaluated (Levis, 1990).

While the model and the methodology were motivated by empirical evidence from a variety of experiments and by the concept of bounded rationality, there were no direct experimental data to support it. An experimental program was undertaken to test the theory and obtain values for the model parameters. The problems under study are those that relate organizational structure directly to performance, as measured by accuracy and timeliness and, more indirectly, to cognitive workload. The first experiment was a single person experiment designed to verify the existence of the bounded rationality constraint (Louvet, Levis, and Casey, 1988). The experiment provided evidence that bounded rationality exists and that for well-defined tasks the degradation of the performance that it causes can be predicted.

To explore the characteristics of organizational performance, multi-person experiments are necessary. The study reported in this thesis shows the results from a model-driven, multi-person experiment which is designed to investigate the effect of organizational structure on performance of DDM organizations.

1.2 CONTRIBUTIONS OF THE THESIS

Analysis and Evaluation for Model-driven Experiments

Experiments involving several human DMs and computer simulations are generally complex and difficult to design and control. One of the difficulties in designing an experiment

for DDM organizations is that a large number of parameters is involved and it is difficult to determine which parameters should be varied and over what range. On the other hand, since human decision makers participate in the experiment, a large number of trials is not feasible. No useful guidelines for model-driven experiments appear available.

In order to design a controllable experiment in a complicated environment, a model is necessary for determining appropriate variables which ought to be controlled or measured. The model of the interacting decision maker developed by Boettcher and Levis (1982) was used as the basis for the experiment design. Dimensional analysis, a technique from the physical sciences, has been extended to include the cognitive aspects of the distributed decision making environment. Then dimensional analysis was applied to the design of a model-driven experiment for the analysis and evaluation of performance.

The model was used to simulate the experiment and predict organizational performance; the results then led to the formulation of hypotheses that could be tested experimentally. Fig.1.1 shows the block diagram of the model-driven experiment design methodology.

Experimental design starts at the problems and issues in distributed decision making (tactical decision making). The first stage is to carry out a theoretical analysis of the problem which includes 1) generating an organization which will carry out the task; 2) designing procedures for doing the task; 3) specifying protocols for interaction between organizational members; 4) defining the strategy space according to the procedures and protocols; 5). applying dimensional analysis to select controlled and measured variables and construct dimensionless groups; 6) running a small-scale pilot experiment to determine the range of response time and available time; 7) applying the evaluation procedure CAESAR to simulate the operation of the organization and predict its performance.

Then hypotheses are generated from model predictions of performance.

The second stage is to design the experiment. Based on the theoretical analysis and the hypotheses, the experiment has to be designed so that the experimental results can be used to test the hypotheses. The assumptions of the theoretical model have to be consistent with the experiment. The ranges of controlled parameters need to be determined so that the experiment is controllable and feasible. Data collection is an critical aspect in the experimental design. It must be emphasized that the completeness and precise data are the keys to success of the experiment.

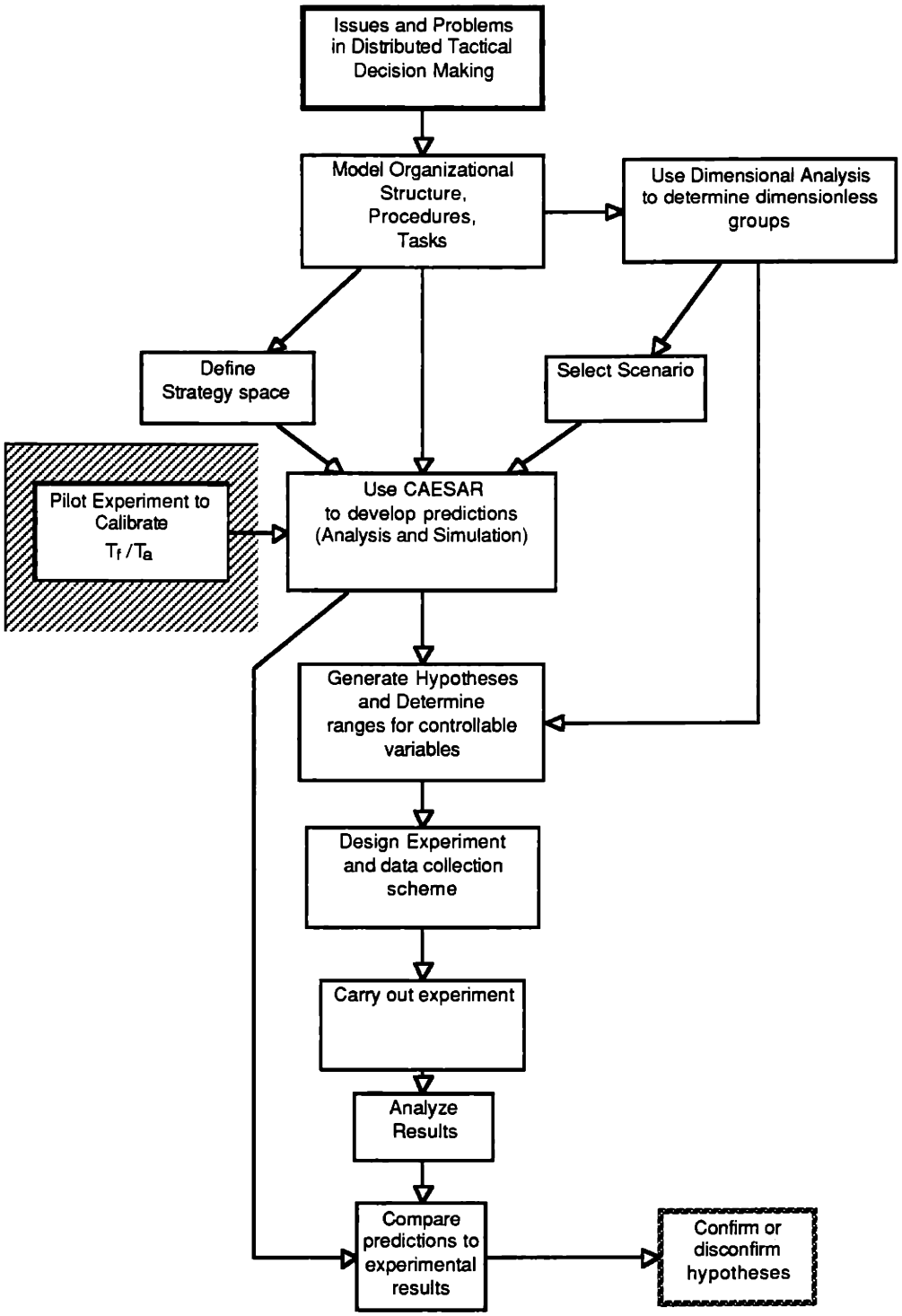


Figure 1.1 Design Sequence Using the Methodology

The third stage is the data analysis. The procedure for analyzing the experimental data should be designed so that the experimental results can be used to test the hypotheses. In some cases data transformation may be necessary for the hypothesis testing.

When the experimental results are available, a comparison of the model predictions and of the experimental result is carried out the fourth stage. Hypotheses are tested and conclusions are drawn.

The methodology for model-driven experiment design reported in this thesis is shown to be feasible and practical; it has led to the design of the experiment and has guided the collection and analysis of the data.

Performance of a DDM organization is measured by the accuracy of the organizational response, the timeliness of the response, and the cognitive workload of individual human decision makers. These three measures are called measures of performance (MOPs). While accuracy measures the quality of organizational response, response time is the amount of time necessary to generate the response. The cognitive workload *rate* that individual DMs may maintain without serious degradation in performance is another design constraint, that is, the bounded rationality constraint.

In Fig.1.2, the *Theoretical Analysis* block shows the use of the mathematical model and of the evaluation methodology. For a given task, an organizational structure is designed. The procedure for carrying out the task is then developed. The evaluation methodology is used to generate three measures: accuracy J^*, cognitive workload G^i, and required number of communications N_{rc}.

The *Experimental Results* block contains the results obtained from the model-driven experiment: actual accuracy J, response time T_f, and actual number of communications among DMs, N_c. Because in a distributed decision making organization the members must interact to perform a task, knowledge on how the members are coordinated is essential to the design. N_c provides insights on the behavior of interacting decision makers (DMs) for a given task, especially on their strategies for coping with increases in the workload rate.

The *MOP* block in Fig.1.2 shows how MOPs are obtained form the theoretical and experimental results. The Performance-workload (P-W) locus characterizes the performance space of the organization under evaluation. Every point in the J-T_f-G locus is a possible operating point of the organization..The comparison of MOPs between the model prediction and the experimental result provides the information to verify the predictions of the theoretical model.

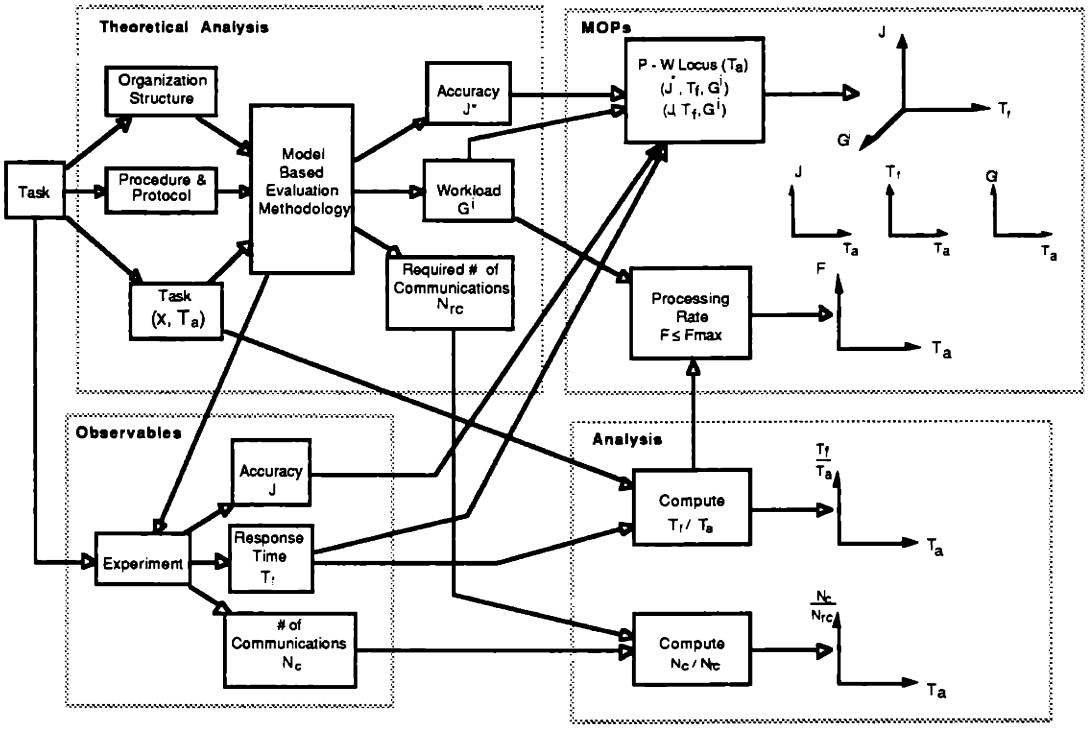


Figure 1.2 Design and Evaluation Procedure for Measures of Performance

The block named as *Processing rate* directly represents the bounded rationality constraint. Because the human operator is an essential element in the decision making organization, cognitive activity during the execution of the task critically affects performance. Bounded rationality can be represented by the existence of the maximum processing rate, denoted by F_{max}. In general, when the time available to do a task is reduced, the processing rate will increase so that the amount of work that needs to be done will be completed. Analytical and experimental results have shown that when the processing rate reaches a maximum value, a further decrease in available time will cause degradation of performance because less work than that required by the task will be done. Therefore, it is necessary to evaluate the information processing rate, F, required by the design to ensure that it remains less than the bound F_{max}.

By comparing the theoretical and experimental results, the evaluation of the organizational performance is completed. All the relationships characterizing organizational performance can be generated and performance can be predicted. For a given design, measures of performance (MOPs) can be computed so that a designer can evaluate and modify the design, if necessary, until requirements are met before actual implementation.

In the *analysis* block, experiment results combined with the task attributes can be used to compute two ratios: the time ratio T_f/T_a, which is the ratio of the response time to the available time to do the task, and the communications ratio N_c/N_{rc}, which is the ratio of the actual number of communications to the required number of communications as specified by the protocol design.

Bounded rationality characteristics can be observed through the critical time ratio at which the performance degrades significantly. For a given task, time spend to complete it is some proportion of the available time. When the available time become shorter and shorter, the response time has to be faster and faster in order to completing the task. The shorter response time, the higher processing rate. However, when the available time is so short that the processing rate reaches a maximum value, further decrease of the available time will not result in a faster response time. Consequently, the performance will degrade because the work required to do the task cannot be done. The time ratio at this point carries the property of the bounded rationality.

The communications ratio indicates the interaction level for an organization. For the tasks under some level of time pressure, the communications ratio reflects the strategies used to perform the task under different time stress levels. It may be expected that DMs reduce amount of interaction when time pressure is very high.

Although the response time and the available time depend on specific tasks, the ratio between the two is a constant at the condition in which the maximum rate is reached. Similar to the time ratio, the communication ratio does not depend on specific tasks although N_c and N_{rc} are task specific. Therefore, the time ratio and the communications ratio are useful for the future experiment design.

Characteristics of Organizational Performance

The results of the experiment capture the general characteristics of DDM organizations.

A special class of organizations was considered - a team of well-trained decision makers repetitively executing a set of well-defined cognitive tasks under severe time pressure. The cognitive limitations of decision makers impose a constraint on the organizational performance. Performance, in this case, is assumed to depend mainly on the time available to perform a task and on the cognitive workload associated with the task. When the time available to perform a task is very short (time pressure is very high), decision makers are likely to make mistakes (human error) so that performance will degrade.

The experimental results show, as predicted, that the accuracy of the response decreases as the available time to do a task is reduced. The variation in performance is less between

different teams than between different individual DMs within a team, which means that organizational *performance is more predictable than individual performance*. It has also been found that degradation of accuracy as a function of available time is less abrupt for organizations than for individuals. *Interaction among DMs in an organization compensates for differences in individual performance characteristics*. These results are consistent with the predictions from the theoretical model. Furthermore, *the critical value of the ratio of response time to available time for doing a task is an observable measure of the bounded rationality constraint*. Therefore, this ratio, which is observable from simple experiments, can be used in future organization designs as a key design parameter.

In summary, the contribution of this thesis is two-fold. The first is *the development of a theoretically based methodology for the design of model-driven experiments*, which can be used for future experiment design. The second is *an increased understanding of the behavior of DDM organizations and validation of a model for predicting performance.*

Organizational design is not unique. In the early stages of the design, there may be several or many structures that seem to be suitable for the task. Which one to choose to proceed with the detailed design is an important question. The models supporting this work and the insights obtained on organizational behavior from the experimental investigation provide useful guidance for addressing that question.

1.3 OVERVIEW OF THE THESIS

This thesis is structured as follows. First, the basic concepts of the relevant theories are introduced, particularly, Petri Net theory and Information theory. The mathematical model of interacting decision makers and the evaluation methodology are introduced as the theoretical basis for this study. Next, dimensional analysis, a scientific and engineering method, is extended and applied to establish the experiment model. The pilot experiment and the actual experiment then are discussed. Finally, the results of the experiment and conclusions are presented.

In Chapter 2, elements of Petri Net theory and Information theory used in the thesis are described briefly.

In Chapter 3, the mathematical model of interacting decision makers and the evaluation methodology are presented.

In Chapter 4, the methodology for designing model-driven experiment is described. The stages in the design are explained.

In Chapter 5, the model-driven experiment is discussed. A naval air defense task is used as the task in the experiment. A hierarchical and a parallel organizational structure are the two structures studied in this thesis. The procedures and the algorithms are explained. The decision strategies and task workload are described.The sample size is determined, and the probability distributions of input parameters are described. Theoretical predictions on the performance and workload are derived.

In Chapter 6, dimensional analysis is extended to include the cognitive aspect of information processing and decision making organizations. The experiment model is established to direct the design of the experiment.

The actual experiment is presented in Chapter 6, including a description of the physical set up.

In Chapter 7, the pilot experiment, its purpose, and its results are presented.

Data analysis and hypothesis testing are described in Chapter 8. In the same chapter, the experiment results are presented and explained. Comparison between the model prediction and the experimental result is shown.

In Chapter 9, the findings of this study are summarized and possible research directions for the future are indicated.

Chapter 2

REVIEW OF RELEVANT THEORIES RELATED TO THE WORK

2.1 PETRI NET THEORY

2.1.1 Introduction

Petri Nets were introduced by Carl Adam Petri in his PhD thesis (1962). Petri Nets are a special class of graphs. They are a modeling and analysis tool that is well suited for the study of the architecture of Distributed Decision Making (DDM) organizations. The use of Petri Nets leads to a mathematical description of the system structure that can then be investigated analytically. In this section, the basic definitions and properties of Petri Nets, relevant to this research, are presented.

2.1.2 Ordinary Petri Nets

Petri Nets are *bipartite* graphs. This means that they have *two* types of nodes. Different symbols are used to distinguish the two types of nodes. By convention, the first type of node is called a ***place*** and is denoted by a circle or ellipse. The second type is called a ***transition*** and is denoted by a solid bar, or a rectangle. The edges of a Petri Net are called ***arcs*** and are always *directed*. The symbols are shown in Fig. 2.1.

A bipartite graph has a special property: an arc can connect only two nodes that belong to different types. Therefore, there can be an arc from a place to a transition, from a transition to a place, but *not* from a place to a place or a transition to a transition.

Definition 2.1 A **Petri Net** is a bipartite directed graph represented by a quadruple

$PN = (P, T, I, O)$ where:

$P = \{p_1, ..., p_n\}$ is a finite set of places.

$T = \{t_1, ..., t_m\}$ is a finite set of transitions.

$I(p,t)$ is a mapping $P \times T \rightarrow \{0,1\}$ corresponding to the set of directed arcs from places to transitions.

$O(t,p)$ is a mapping $T \times P \rightarrow \{0,1\}$ corresponding to the set of directed arcs from transitions to places.

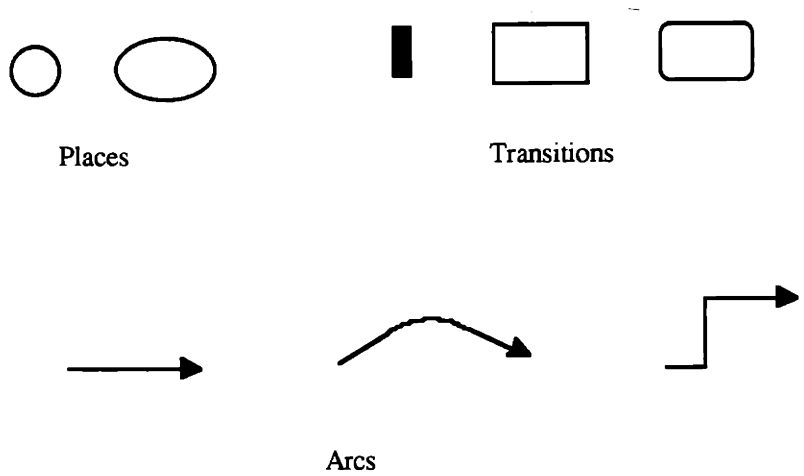


Figure 2.1 Places, Transitions, and Arcs

The nets under consideration in this thesis, where I and O take the values of 0 or 1, are called Ordinary Petri Nets. An example of a Petri Net is shown in Fig. 2.2; let it be denoted PN_1. Places are represented by circles and transitions by bars. This is the convention that will be adopted for Ordinary Petri Nets.

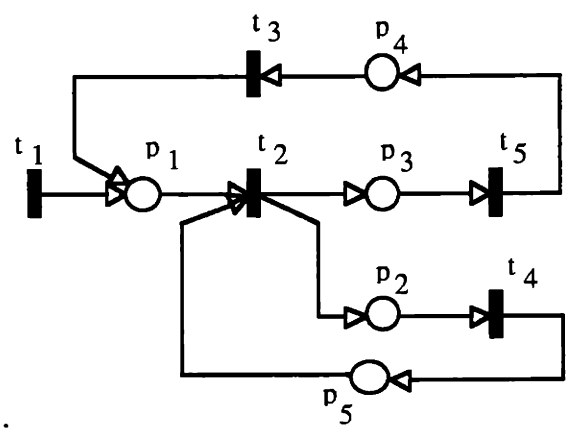


Figure 2.2. Petri Net PN_1.

Self-loops and Pure Petri Nets

A place p and a transition t are on a *self-loop,* if p is both an input and an output place of t. A Petri Net will be *pure,* if it does not contain self loops.

The Petri Net PN_1 on Fig. 2.2 is pure, while the Petri Net PN_2 in Fig. 2.3 is not: it contains the self-loop (t_2, p_3).

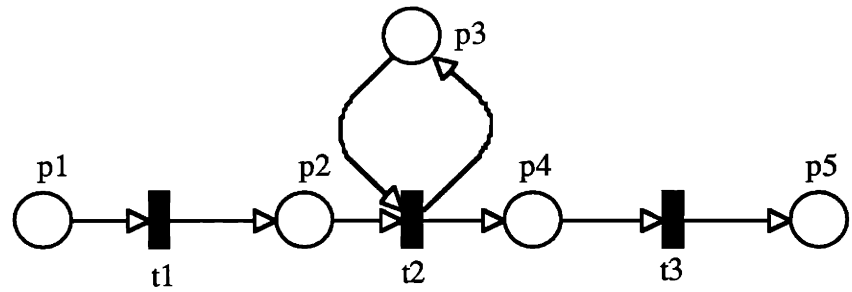


Figure 2.3 A Petri Net PN_2 with a Self-Loop (t_2, p_3)

2.1.3 Markings and Execution

Petri Nets would not be very useful if all we could do is draw a diagram describing the relationships among the objects represented by the nodes. An essential feature of Petri Nets is that they can be executed; one can observe the interactions between the components and study the dynamics of the system modeled by a Petri Net. For this purpose, the marking of a Petri Net is introduced.

In addition to the two types of nodes - places and transitions - and the arcs, a fourth object is introduced in order to describe the dynamics of a Petri Net. This object is the *token,* denoted by a solid dot • , and residing inside the circles representing the places. In Ordinary Petri Nets, the tokens do not represent specific information and are not distinguishable. They are only markers, indicating the presence or absence of whatever they represent - a condition, a signal, a piece to be machined, etc. Places can hold an arbitrary number of tokens, or they can be restricted as to the number they can hold. Also, arcs can have capacities associated with them: they can let a single token go through at a time, or they can let a finite number.

Marking

A *marking* of a PN - denoted by M - is a mapping: $P \rightarrow \{0, 1, 2, ...\}$ which assigns a non-negative integer number of tokens to each place of the net. A marking can be represented by a n-dimensional integer vector whose components correspond to the places of the net. In Fig. 2.2, no tokens are shown. The marking of this net is denoted by the following null vector:

$$PN_1: \qquad M^0 = [0\ 0\ 0\ 0\ 0]^T$$

since the net has five places.

The marking vector represents the state of the Petri Net, i.e., the distribution of tokens in the places of the net defines its state. The system state changes when the distribution of tokens changes. It should be apparent that, in general, the number of states of a Petri Net is very large. Consider, for example, a net in which the places are allowed to hold at most one token. If there are n places in the net, then the possible number of states is 2^n. The process by which the distribution of tokens changes is the firing of transitions. Recall that we are considering nets with places that can hold an arbitrary number of tokens and with arcs that have capacity unity. Then, the following definitions hold:

Enablement and Firing

A transition t is *enabled* by a given marking M if and only if there is at least one token in each input place of t. When a transition is enabled it can *fire*. A token is removed from each of the input places of t (the preset of t) and a token is placed in each of the output places of t (the postset of t). The new marking M' reached after the firing of t is defined as follows:

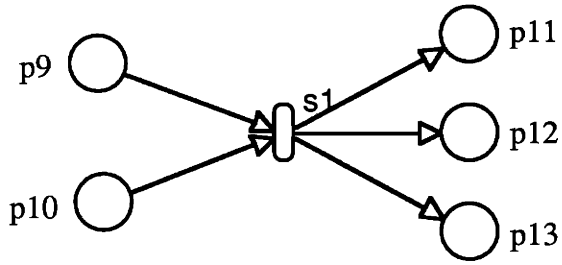
$$(\forall p \in P) \quad M'(p) = M(p) + O(t,p) - I(p,t). \tag{2.1}$$

Switch

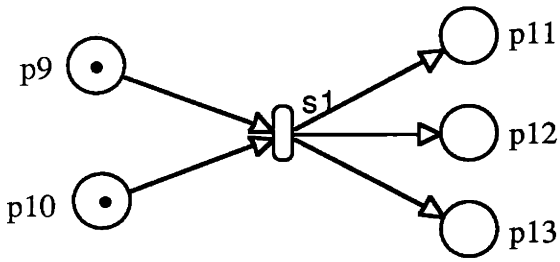
A *switch* is a transition with multiple output places and some *decision rule* which directs the generation of a token after firing in one and only one of its output places.

Since a switch is really a transition, the firing rules for a switch are identical to the firing rules for a transition: a switch will fire if all its input places contain at least one token. Unlike regular transitions however, all the output places of a switch will not receive a token. Only one of them will. This place will be chosen by the internal decision rule associated with the switch. An example of a switch with two input places and three output places is shown in Fig. 2.4. The output places of the switch are called the *branches* of the switch.

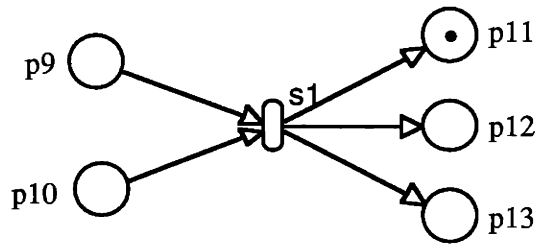
The decision rules associated with the switch can be deterministic or stochastic. For example, they can implement priority rules based on the marking of the preset of the switch. Decision rules may be represented by algorithms, but they can also involve techniques derived from artificial intelligence. While there are virtually no limitations on the kind of decision rules to be associated with a switch, note that the presence of such rules makes the Petri Net not be an ordinary one. The more sophisticated the decision rules, the more involved the analysis of the resulting Petri Net. The trade-off has to be made by the designer of the net.



(a) Representation of Switch S1



(b) Switch s1 is enabled



(c) Only output place p11 contains token

Figure 2.4 Example of a Three-Branch Switch.

In this section, some basic definitions and fundamental properties of Ordinary Petri Nets were defined. The purpose was to introduce a graphical and analytical formalism for use in the modeling and evaluation of distributed decision making organizations, a formalism that allows to model in a consistent way both humans and intelligent machines.

2.2 INFORMATION THEORY

Information theory was first developed by Shannon (Shannon and Weaver, 1949). It started by addressing problems in communications, but has since evolved as a valid mathematical theory in its own right, and it is useful for applications in many disciplines, including human decision making (Levis, 1984).

Consider an experiment in which there are n possible, distinct outcomes denoted by

$$x_i, \quad i = 1, \ldots, n.$$

Each outcome can occur with probability

$$p(x = x_i) = p(x_i) = p_i$$

which satisfies the following two conditions:

$$p_i \geq 0 \quad \text{and} \quad \sum_{i=1}^{n} p_i = 1$$

The set of possible outcomes, the values that the discrete random variable x can take, is denoted by X and may be referred to as the alphabet of x. The probabilities p_i express the uncertainty associated with the outcome of the experiment.

The first quantity of interest in information theory is *entropy*: The entropy H(x) of the discrete random variable x is defined to be

$$H(x) \equiv - \sum_{x} p(x) \log p(x) \tag{2.2}$$

and is measured in bits when the base of the algorithm is two. Entropy expresses the uncertainty associated with the next outcome of the experiment - uncertainty as to what the next value of x will be.

A number of useful properties of entropy follow.

(a) Entropy is a non-negative quantity:

$$H(x) \geq 0$$

The proof of this statement is clear from the definition, equation (2.2). The probabilities p(x) are non-negative and have values in [0, 1]. Consequently, log p(x) is non-positive, and the sum of the products is also non-positive. The presence of the minus (-) sign makes entropy a non-negative quantity. Note that for completeness, we need to define that

$$\text{for } p = 0 \qquad p \log p = 0.$$

(b) If $p(x = x_i) = 1$ for some i, then $H(x) = 0$.

This second property shows that if the outcome of the experiment is known with probability 1, i.e., the experiment has one and only one outcome, then the uncertainty regarding the outcome is zero. While this property shows the condition for minimum entropy, the next property establishes the condition for maximum entropy.

(c) The entropy of a discrete random variable x attains its maximum value when p(x) is the uniform probability distribution:

$$H(x) = H(p1, p2, \ldots, pn) \leq H(\frac{1}{n}, \ldots, \frac{1}{n})$$

The definition of entropy can be extended to N random variables. For two random variables, i.e., given two variables x and y, elements of the alphabets X and Y, and given p(x), p(y), and p(x|y) (the conditional probability of x, given the value of y):

$$H(x, y) = - \sum_x \sum_y p(x, y) \log p(x, y) \tag{2.3}$$

Similarly, conditional entropy is defined as follows

$$H_y(x) \equiv - \sum_y p(y) \sum_x p(x|y) \log p(x|y) \tag{2.4}$$

which expresses the uncertainty in x given full knowledge in the outcomes y. The following identity relates the entropy of two random variables and the corresponding conditional entropies:

(d) $H(x, y) = H(x) + H_x(y) = H(y) + H_y(x)$

If two events are independent, in which case $p(x, y) = p(x) \bullet p(y)$, then

$$H(x, y) = H(x) + H(y)$$

The other quantity of interest is average mutual information or transmission or throughput. The transmission between x and y, T(x:y), is defined to be

$$T(x : y) \equiv H(x) - H_y(x) \tag{2.5}$$

It is easy to show (Boettcher, 1981) that the following relationships hold:

(e) $T(x:y) = H(x) - H_y(x) = H(x) + H(y) - H(x, y) = H(y) - H_x(y) = T(y:x)$

McGill (1954) generalized this basic definition of transmission to N dimensions by extending equation (2.5):

$$T(x_1:x_2:\ldots:x_N) \equiv \sum_{i=1}^{N} H(x_i) - H(x_1:x_2:\ldots:x_N) \tag{2.6}$$

The advantage of this definition is that it can be expressed as the sum of similar, but simpler quantities. For example, for N = 4:

$$T(x_1 : x_2 :x_3 :x_4) = T(x_1 :x_2) + T(x_3 :x_4) + T(x_1 ,x_2 :x_3 ,x_4)$$

2.3 THE PARTITION LAW OF INFORMATION (Conant, 1976)

The Partition Law of Information is defined for a system with N-1 internal variables, w_1 through w_{N-1}, and an output variable, y, also called w_N. The law states:

$$\sum_{i=1}^{N} H(w_i = T(x:y) + T_y(x:w_1:w_2:\ldots:w_{N-1}:y) + H_x(w_1,w_2,\ldots,w_{N-1},y) \tag{2.7}$$

The left-hand side of equation (2.7) refers to the total activity of the system, also designated by G. Each of the quantities on the right-hand side has its own interpretation. The first term, T(x:y), is called *throughput* and is designated G_t. It is the transmission between input and output and measures the amount by which the output of the system is related to the input. The second quantity,

$$T_y(x:w_1,w_2,\ldots,w_{N-1}) = T(x:w_1,w_2,\ldots,w_{N-1},y) - T(x:y) \tag{2.8}$$

is called *blockage* and is designated G_b. The first term on the right hand side of equation (2.8) is the transmission between the input and all the variables of the system including the output; the second term is the transmission between input and output, or G_t. Their difference is a

measure of the amount of information in the input that is not included in the output, but has been blocked in the internal variables of the system.

Throughput and blockage represent, in a sense, the partition of the information contained in the input, some of it is transmitted to the output and some is blocked within the system. Another possibility is that some of the information in the input never enters the system, that it is rejected outside the system boundary. This will be the uncertainty remaining about the input when both the output and the set of internal variables of the system are known:

$$G_r = Hw_1, w_2, \ldots, w_{N-1}, y\ (x)$$

This term, G_r, is called *rejection*. It can be shown (Conant, 1976) that

$$H(x) = G_t + G_b + G_r$$

If there is no rejection, and if the entropy of the input is known, only throughput or only the blockage needs to be calculated.

The third term, $T(w_1{:}w_2{:}\ldots{:}w_{N-1}{:}y)$ is called *coordination* and is designated G_c. It is the N-dimensional transmission of the system, i.e., the amount by which all of the internal variables in the system constrain each other. It can be thought of as a measure of all the interactions among system variables, whether they are directly related to the production of an output or not. Since the mathematical expression for coordination is that of transmission among N variables, the decomposition property of transmission can be used to express the coordination in a system as the sum of the coordination within and among its subsystems.

The last term, $H_x(w_1, w_2, \ldots, w_{N-1}, y)$, designated by G_n, represents the uncertainty that remains in the system variables when the input is completely known. This *noise* should not be construed to be necessarily undesirable, as it is in communication theory; it may also be thought of as internally-generated information supplied by the system to supplement the input and facilitate the decision making process.

The partition law may be abbreviated:

$$G = G_t + G_b + G_c + G_n \tag{2.9}$$

The calculation of total activity as well as the components in which it is partitioned depend on the structure of the problem and on the input alphabet and the uncertainty associated with it. If the probability distribution of x is changed, then different numerical values will be obtained. If the alphabet is changed, then G will change; if the decision rules and the protocols

are changed, then G will change. This is one of the features of the information theoretic expressions in the partition law that make G attractive as a measure of cognitive task workload.

2.4 SUMMARY

In this chapter, elements of Petri Net theory and Information theory have been presented. In the following chapters, these theories will be applied to model distributed decision making organizations. Petri Nets will be used to model the organizational protocol and operation sequence. Information theory will be used to compute a measure for the cognitive workload of DMs. The next chapter illustrates the application of the results to the modeling of the interacting decision maker.

Chapter 3

ANALYSIS OF DECISION MAKING ORGANIZATIONS

3.1 THE DECISION MAKER MODEL

The basic model of the memoryless decision maker with bounded rationality is based on the hypothesis of F.C. Donders (1983) that information processing is done in stages. Specifically, it is assumed that the two stages are (a) situation assessment (SA), and (b) response selection (RS), which correspond to March and Simon's (1958) two-stage process of discovery and selection. The structure of this model is shown in Fig. 3.1.

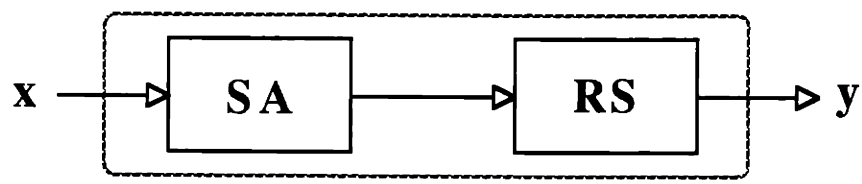


Figure 3.1 Two-Stage Model

This model is a general one and as such it is not very useful for analysis and design. We need to add internal structure to the two stages, if we are to model the information processing and decision making tasks. We assume first that the decision maker is well trained for the tasks to be performed, i.e., for processing the input x to produce the output y. Indeed, we assume that he has several procedures or algorithms with which he can accomplish the situation assessment task and the response selection task. Since we have assumed that he is well trained, we can consider the set of procedures as given and that the DM will not change them as a task is executed. The implication of the last assumption is that there is no learning while an input is being processed. While the number and nature of the algorithms can change with time, it is assumed that the time required is much longer than the interarrival time of the inputs to be processed.

The internal structure of the situation assessment (SA) stage is shown in Fig. 3.2. The alternative algorithms or procedures are depicted as a set of U transitions in parallel, labelled f_1 to f_U. The switch represents the act of selecting from among these procedures the one to be used to process the incoming input x. The rule for selecting is described by the probability distribution p(u). The discrete variable u represents the position of the switch; if u = 1 then f_1

will be used to process x. Note that the rule does not depend on x. The reason is simple: if some information about x were known, then that information could be used to select the appropriate procedure for processing x; we could use p(u|x) instead of p(u). But this is the situation assessment stage and x is received from the environment; there is no prior information about x.

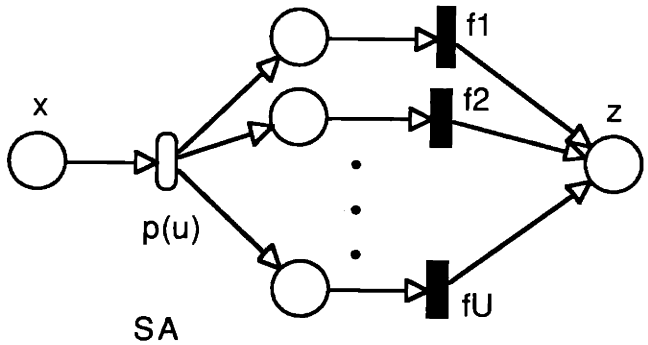


Figure 3.2 Petri Net Representation of the Situation Assessment Stage

This is not the case for the response selection stage, Fig. 3.3. The RS stage receives the *internal* input z and selects a procedure h_j, j = 1, . . . , V to produce the output y. The rule that determines the position of the switch, p(v | z), depends on the assessed situation z.

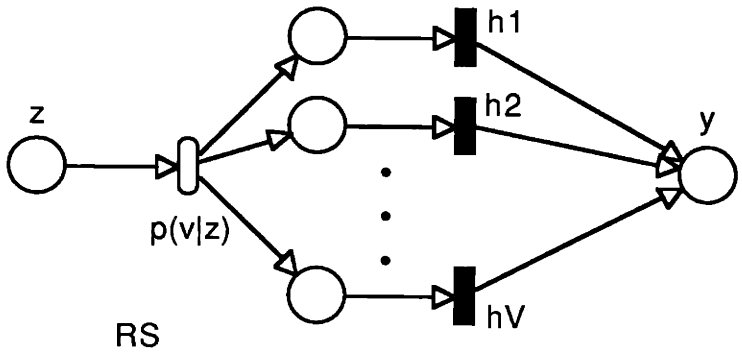


Figure 3.3 Petri Net Representation of the Response Selection Stage

These two models are based on the view that a decision consists of two parts, judgment and choice (Einhorn and Hogarth, 1980). It is the sequencing of these two processes that is used to model the decision maker. Consider the model that results when the two stages are concatenated as shown in Fig. 3.4.

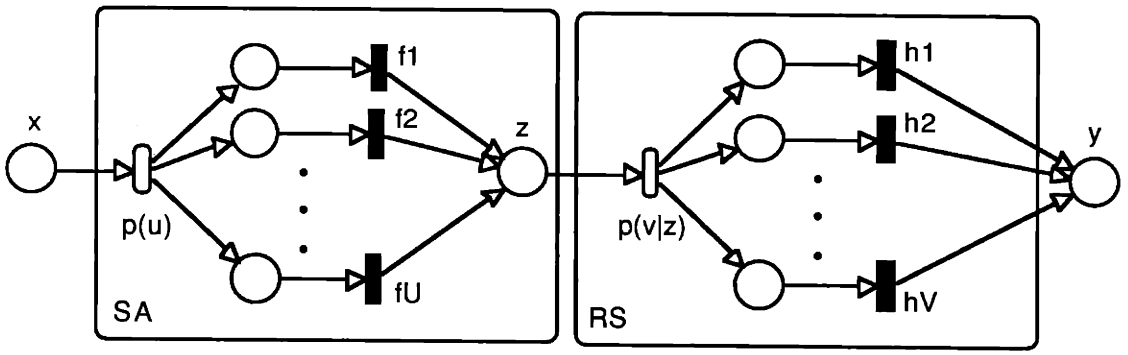


Figure 3.4 Petri Net Representation of the DM Model

Since there is no preprocessing of the input, the DM chooses at random one of the algorithms f. Thus choice can be interpreted as the selection among options on the basis of rules. The algorithms that process the data to arrive at the assessed situation are the instantiation of judgment. Given the results of that process, a new choice is made. The result of the choice is the processing of the assessed situation to produce the output. It is a process analogous to judgment, except that it will affect the first choice of the recipient of the output y.

3.2 THE INTERACTING DECISION MAKER MODEL

In order to model distributed decision making organizations, the single two-stage DM model (Fig. 3.4) was extended (Boettcher and Levis, 1982) to include interactions with other organization members (Fig. 3.5).

The DM receives signals $x \varepsilon X$ from the environment with interarrival time τ. The SA stage contains algorithms that process the incoming signals to obtain the assessed situation z. In the interacting decision maker model, the assessed situation z may be shared with other organization members; the DM may receive the supplementary situation assessment z' from other parts of the organization; the two sets z and z' are combined in the information fusion (IF) processing stage to obtain z".

The possibility of receiving guidance, instructions, or commands from other organization members is modeled by the variable v'. A command interpretation (CI) stage of processing is necessary to combine the final situation assessment z" and v' to arrive at the choice v of the appropriate strategy to use in the response selection (RS) stage. The RS stage contains algorithms that produce outputs y in response to the situation assessment z" and the command inputs. The Petri Net representation of this model is shown in Fig. 3.6.

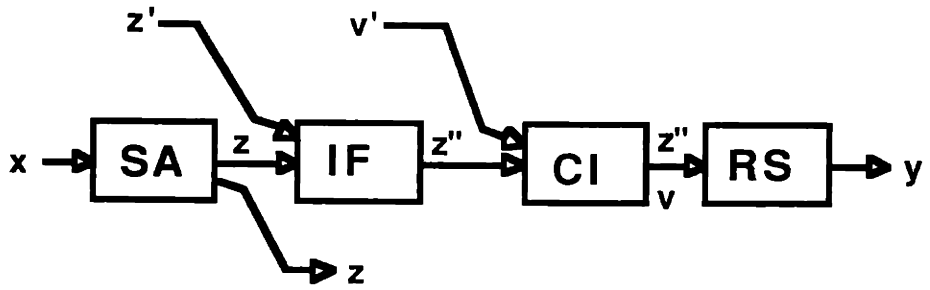


Figure 3.5 Four-Stage Interacting Decision Maker Model

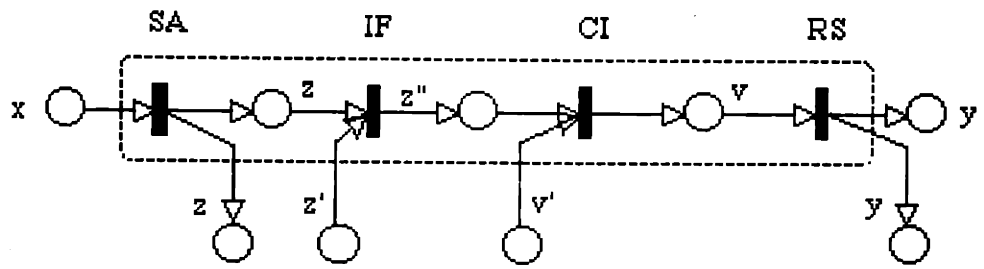


Figure 3.6 The Memoryless Interacting Decision Maker Model

This model can be made explicit by showing the internal structure of the SA and RS stages. As in the case of the non-interacting DM, the situation assessment stage consists of a set of U algorithms (deterministic or not) that are capable of producing some situation assessment z and the response selection stage consists of V algorithms that can produce the output y. The Petri Net model that corresponds to Fig. 3.4 is given in Fig. 3.7.

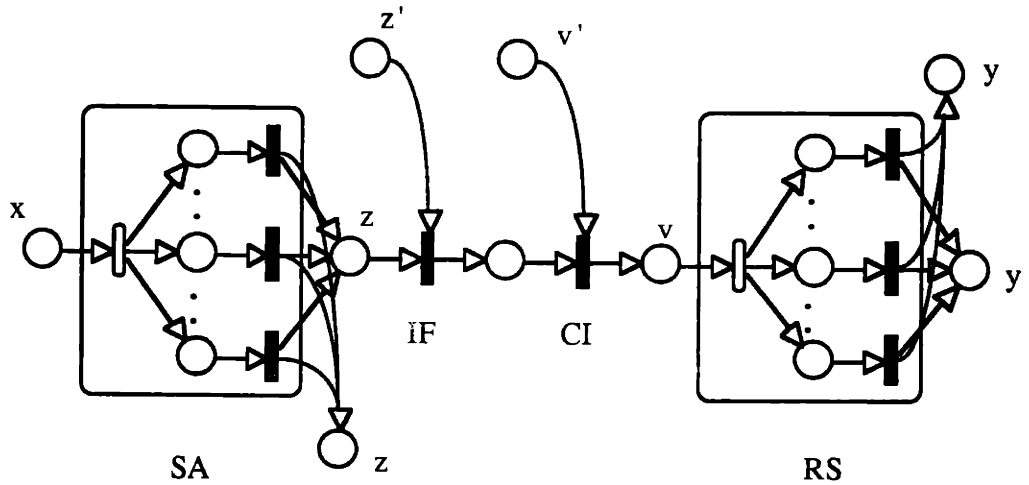


Figure 3.7 Petri Net Representation of Interacting DM Model

3.3 TASK MODEL

An organization interacts with its environment; it receives signals or messages in various forms that contain information relevant to the organization's tasks. These messages must be identified, analyzed, and transmitted to their appropriate destinations within the organization. From this perspective, the organization acts as an information user.

Let the organization receive data from one or more sources (N') external to it. Every τ_n units of time on the average, each source n generates symbols, signals, or messages x_{ni} from its associated alphabet X_n, with probability p_{ni}, i.e.,

$$p_{ni} = p(x_n = x_{ni}) \quad ; \quad x_{ni} \in X_n \quad i = 1, 2, \ldots, \gamma_n \tag{3.1}$$

$$\sum_{i=1}^{\gamma_n} p_{ni} = 1; \quad n = 1, 2, 3, \ldots, N' \tag{3.2}$$

where γ_n is the dimension of x_n. Therefore, $1/\tau_n$ is the mean frequency of symbol generation from source n.

Rather than considering these sources separately, one supersource, composed of these N' sources, is created. The input symbol x' may be represented by an N'-dimensional vector with each source corresponding to a component of this vector, i.e.,

$$\underline{x}' \equiv (x_1, x_2, \ldots, x_{N'}) \quad ; \quad \underline{x}' \in X \tag{3.3}$$

To determine the probability that symbol $\underline{x}'_j$ is generated, the independence between components must be considered. If all components are mutually independent, then p_j is the product of the probabilities that each component of $\underline{x}'_j$ takes on its respective value from its associated alphabet:

$$p_j = \prod_{n=1}^{N'} p_{nj} \tag{3.4}$$

If two or more components are probabilistically dependent on each other, but as a group are mutually independent from all other components of the input vector, then these dependent components can be treated as one supercomponent, with a new alphabet. Then a new input

vector, $\underline{x}$, is defined composed of the mutually independent components and these super-components.

This model of the sources implies synchronization between the generation of the individual source elements so that they may, in fact, be treated as one input symbol. Specifically, it is assumed that the mean interarrival time τ_n for each component is equal to τ. It is also assumed that the generation of a particular input vector, $\underline{x}_j$, is independent of the symbols generated prior to or after it. That is, the source is memoryless. The last assumption can be weakened, if the source is a discrete stationary ergodic one with constant interarrival time τ that could be approximately by a Markov source. Then the information theoretic framework can be retained (Hall, 1982).

The vector output of the source is partitioned into groups of components that are assigned to different organization members. The j-th partition is denoted by $\underline{x}^j$ and is derived from the corresponding partition matrix π^j which has dimension n_j x N and rank n_j, i.e.,

$$\underline{x}^j = \pi^j \underline{x}. \tag{3.5}$$

Each column of π^j has at most one non-zero element. The resulting vectors $\underline{x}^j$ may have some, all, or no components in common.

The set of partitioning matrices $\{\pi_1, \pi_2, \ldots, \pi_n\}$, shown in Fig. 3.8 specify the components of the input vector received by each member of the subset of decision makers that interact directly with the organization's environment. These assignments can be time invariant or time varying. In the latter case, the partition matrix can be expressed as:

$$\pi^j(t) = \begin{Bmatrix} \pi_0^j & \text{for } t \in \{T\} \\ 0 & \text{for } t \notin \{T\} \end{Bmatrix}$$

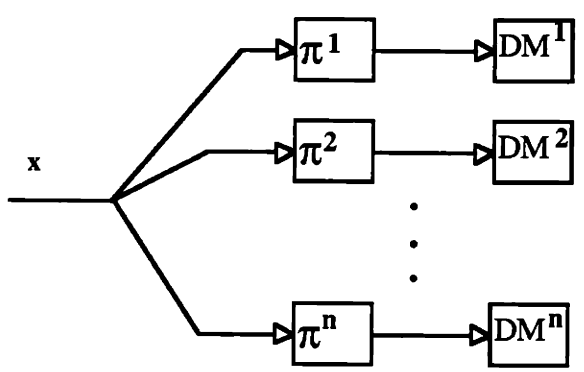


Figure 3.8 Information Structures for Organizations

The times {T} at which a decision maker receives inputs for processing can be obtained either through a deterministic (e.g., periodic) or a stochastic rule. The question of how to select the set of partition matrices, i.e., design the information structure between the environment and the organization, has been addressed by Stabile (1981, 1984).

3.4 TOTAL ACTIVITY

The analytical framework presented in section 3.2, when applied to the single interacting decision maker with deterministic algorithms in the SA and RS stages, yields the four aggregate quantities that characterize the information processing and decision making activity within the DM. The model has three inputs, x, z', and v'. In addition to the output y it communicates to the rest of the organization its situation assessment z. Therefore, the expression for the throughput includes all inputs and outputs:

Throughput:

$$G_t = T(x, z', v' : z, y) \tag{3.6}$$

If there is no rejection, then blockage can be determined by:

Blockage:

$$G_b = H(x, z', v') - G_t \tag{3.7}$$

Internally generated information (Noise):

$$G_n = H(u) + H_{z'',v'}(v) \tag{3.8}$$

Note that the noise G_n includes two terms; the first term is simply H(u) since the decision rule p(u) does not depend on any other variables. The second term denotes the uncertainty due to the second stage decision strategy, $p(v \mid z'', v')$ which is conditioned on the final situation assessment z'' and any option restriction signal or command v' from the rest of the organization.

Coordination:

$$G_c = \sum_{i=1}^{U} [p_i g_c^i(p(x)) + \alpha_i \mathbf{H}(p_i)] + H(z) + g_c^{IF}(p(z,z')) + g_c^{CI}(p(z'',v'))$$
$$+ \sum_{j=1}^{V} [p_j g_c^j(p(z'' \mid v=j)) + \alpha_j \mathbf{H}(p_j)] + H(y) + H(z) + H(z'') + H(z'',v) \qquad (3.9)$$

In the expression 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 v = j. The quantities g_c represent the internal coordinations of the corresponding algorithms and depend on the probability distribution of their respective inputs; the quantities α_i, α_j are the number of internal variables of the algorithms f_i and h_j, respectively. Finally, the quantity $\mathbf{H}(p)$ is the entropy of a binary random variable that takes one of its two values with probability p.

$$\mathbf{H}(p) = - p \log_2 p - (1-p) \log_2(1-p) \qquad (3.10)$$

The expression for the coordination term reflects the presence of the two switches in the model as well as the two fusion processes, the information fusion stage and the command interpretation stage. The first sum contains two types of terms. The first term weights the coordination associated with each algorithm f, denoted by g_c^i, with the probability p_i that this algorithm will be selected. The term has been derived under the assumption that the U algorithms constitute disjoint subsystems (they have no variables in common) and that no two algorithms are active at the same time. The second term in the sum arises from the activity required for switching among algorithms. Since α_i is the number of variables in algorithm i, the term may be interpreted as the workload required to initialize the variables of that algorithm before it processes x. Similar interpretation can be made of the two types of terms in the second sum.

The first term H(z) appears because it is the only variable that is related to all the algorithms in the SA stage and the term H(y) appears because it is related to all the algorithms in the RS stage. The remaining entropy terms arise from the coordination among subsystems.

Equations (3.6) to (3.9) determine the total activity G of the decision maker according to the partition law of information, equation (2.9). Since the quantity G may be interpreted as the total information processing activity of the system, it can serve as a *measure for the workload* of the organization member in carrying out a decision making task.

3.5 WORKLOAD AND BOUNDED RATIONALITY

The qualitative notion that the rationality of a human decision maker is not perfect, but is bounded (March, 1978), will be modeled as a constraint on the total activity G. The specific form for the constraint has been suggested by the empirical relation

$$t = c_1 + c_2 G_t$$

where t is the average reaction time, i.e., the time between the arrival of the input and the generation of an output y. It is assumed that the decision maker must process his inputs at a rate that is at least equal to the rate with which inputs arrive. The latter has been model by τ, the mean symbol interarrival time:

$$t = c_1 + c_2 G_t \leq \tau$$

or

$$\frac{1}{c_2} t = \frac{c_1}{c_2} + G_t \leq \frac{1}{c_2} \tau$$

The modeling assumptions in this work are that

$$\frac{c_1}{c_2} = G_b + G_n + G_c$$

and that c_2 does not depend on p(x). Then, the bounded rationality constraint takes the form

$$G = G_t + G_b + G_n + G_c \leq \frac{1}{c_2} \tau = F \tau \tag{3.11}$$

where F can be considered as a rate of total activity and is measured in bits per second. Inequality (3.11) represents a mathematical expression of only one aspect of bounded rationality. Other formulations are possible.

In the experimental psychology and behavioral analysis literature, one may find two different approaches which may be related to the concept of human bounded rationality: decision making under time pressure and the Yerkes-Dodson 'law'. Considerable experimental psychological work has been done on the influence of arousal on performance in various types of tasks (Casey, 1988). Figure 3.9 shows the relationship between arousal and performance (the Yerkes-Dodson 'law'). This relation is shown when arousal is varied over an extremely wide range. Arousal is influenced by a variety of factors including cognitive workload. At

very low arousal, performance is low due to boredom and vigilance limitations. At very high arousal, performance is also low, but it is then due to stress and sensory overload. In a well designed organization, all decision makers should be operating near the top of the curve at the designed organizational performance limit.

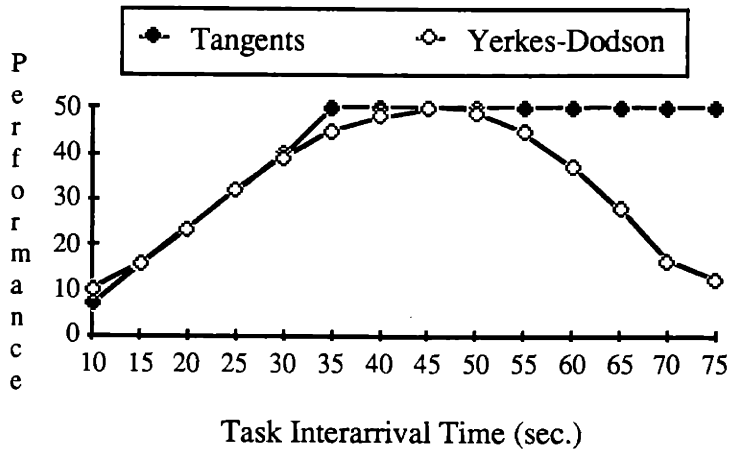


Figure 3.9 The Yerkes-Dodson Law.

Given the objective of this work, namely, the design of experiments for evaluating distributed decision making organizations, the Yerkes-Dodson law is approximated by the two asymptotes shown in Fig. 3.9. The horizontal asymptote is justified by the realization will operate under time pressure. Consequently, the degradation of performance due to boredom or lack of vigilance is not a phenomenon that need concern us. In the information theoretic model, it is assumed that simple information processing tasks are performed with little error when both the rate of information processing imposed by the input interarrival rate is low and the decision maker is not bored. The degradation due to inadequate time to perform the task is of primary concern. The decrease in performance has been approximated by the tangent to the left portion of the curve. The point of interest that leads to the determination of the bounded rationality constraint is the intersection of the two asymptotes. To avoid degradation of performance, a necessary condition is to keep the interarrival time larger than the critical value that corresponds to the intersection.

To interpret the reason for the degradation of performance, consider the following. Let us postulate the existence of a task that can be performed by three different procedures or algorithms. The higher the workload associated with an algorithm, the better the performance. For this example, let the three algorithms have task workload G equal to 50, 100, and 200 bits per task, respectively. The corresponding workload rate F is obtained by dividing the task

workload G by the interarrival time, as shown in Fig. 3.10. Consider now a task that requires the use of the procedure for which G3 is 200 bits per task in order for performance to be perfect. When the interarrival time is 40 seconds, the resulting rate is 5 bits per second. As the interarrival time is decreased, the rate increases as shown by the top curve (F(G3)) in Fig. 3.10. Let us assume that the maximum rate F_{max} for no degradation of performance is 10 bits per second and it is achieved when the interarrival time is 20 seconds. If the input interarrival time is decreased further, the decision maker cannot increase his information processing rate in order to accomplish the task: the decision maker will be overloaded.

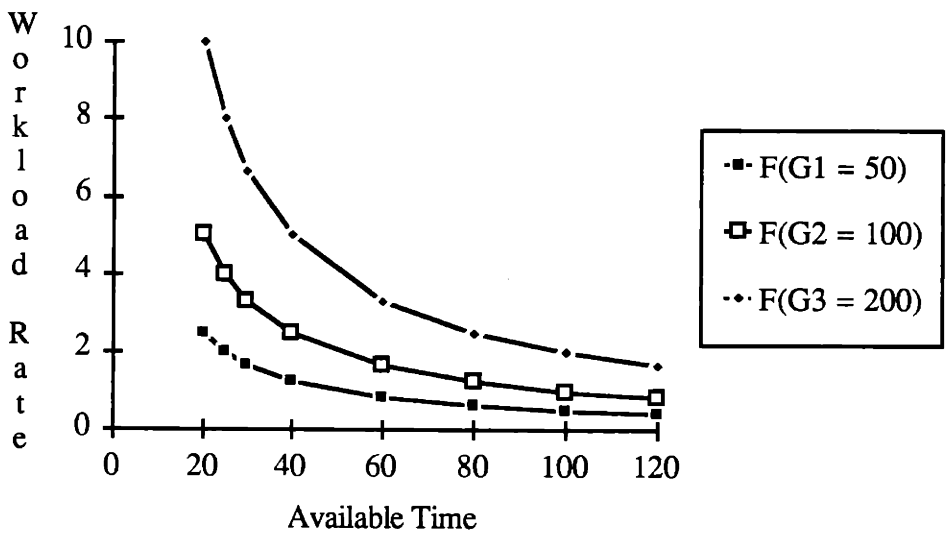


Figure 3.10 Workload Rate and Available Time for Various Task Workloads.

What the DM can do is to change procedure (and thus shed load) so that he can still accomplish the task. In Fig. 3.10, this means that the operating point stops moving along line F(G3), and drops to the next line under it, to the curve F(G2). The DM can use this procedure with some degradation in performance until a new decrease in interarrival time forces him to move to a lower workload curve (F(G1)) while maintaining the maximum possible workload rate without overloading. As a result, a "saw-tooth" shaped curve is obtained (Fig. 3.11). This process leads to a gradual decrease in performance that is consistent with the left hand side of the Yerkes-Dodson law.

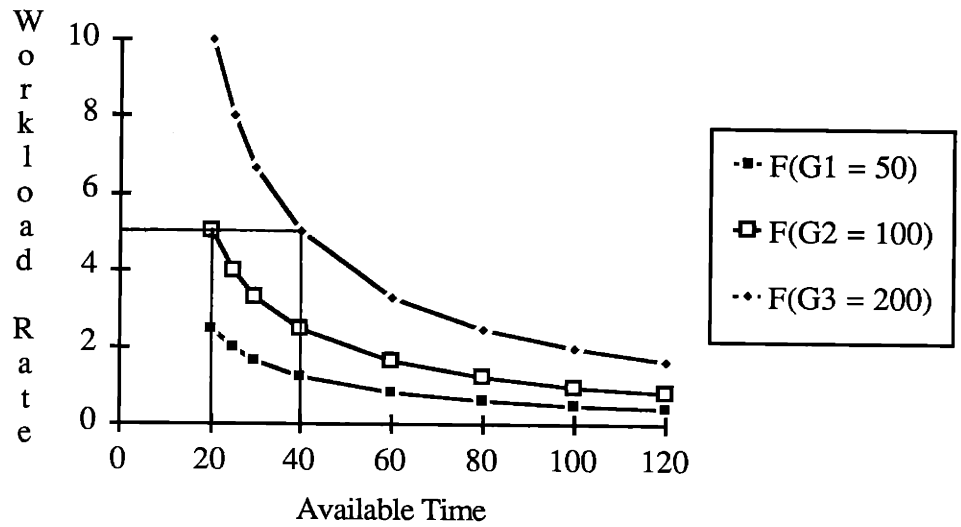


Figure 3.11 Workload Rate and Discrete Available Time for Various Workloads

The decision maker's coping strategies are not statistically predictable and may take many forms. Examples of coping strategies may be to ignore entire inputs, simplify the algorithms used to give less accurate responses, etc.(Miller, 1969). The existence and the behavior of the bounded rationality constraint were tested with the experiment described in (Louvet, Casey, and Levis, 1988).

Weakening the assumption that the algorithms are deterministic changes the numerical values of the noise term G_n and of the coordination term G_c (Chyen, 1984). If memory is present in the model, then additional terms appear in the expressions for the coordination rate and for the internally generated information rate (Hall, 1982).

3.6 MEASURES OF PERFORMANCE

The organization's task is defined as the processing of the input symbols x to produce output symbols. This definition implies that the organization designer knows a priori the set of desired responses Y and, furthermore, has a function or table L(x) that associates a desired response or a set of desired responses to each input $x \varepsilon X$. One measure of performance (MOP) of the organization that reflects the degree to which the actual response matches the desired response can be computed as shown in Fig. 3.12.

The decision maker's actual response y can be compared to the desired response Y and a cost is assigned using the cost function d(y, Y). If this function is a binary one, i.e.,

$$d(y, Y) = \begin{cases} 0 & \text{if } y = Y \\ 1 & \text{if } y \neq Y \end{cases} \tag{3.12}$$

then the expected value of this cost denotes the probability that the wrong decision is made, i.e., it is the probability of error.

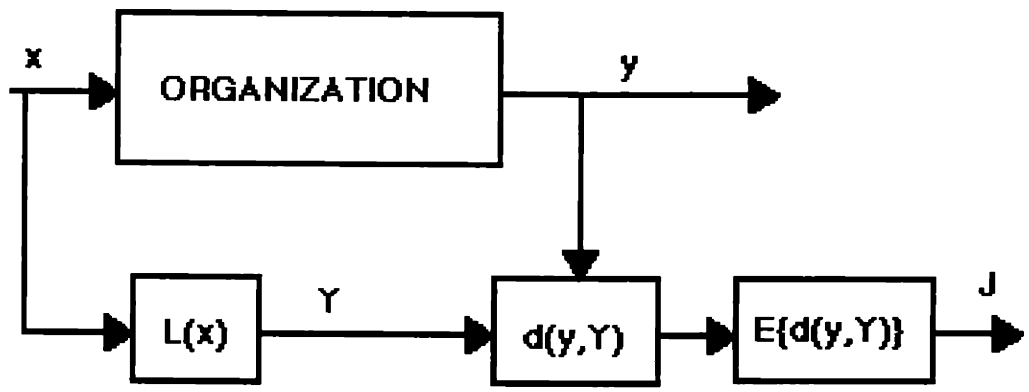


Figure 3.12 Performance Evaluation of an Organization

In general, however, there is a cost c_{ij} associated with selection y_i when the desired response to input x_j is Y_j :

$$c_{ij} = d(y_i, Y_j) \tag{3.13}$$

so that

$$J = \sum_j p(x_j) \sum_i c_{ij}\, p(y_i | x_j) \tag{3.14}$$

This measure of performance can be interpreted as a measure of the *accuracy* of the response, to the extent that a cost is associated with the degree with which the actual decision deviates from the desired one.

This class of performance measures, described generically by (3.14), is not the only one that we will consider. Two other measures of performance are considered.

Response time

The response time measures the delay in an organization's response. Many information processing and decision making tasks, such as air traffic control, nuclear power plant emergency handling, or military operations, have time constraints. The value of information depends on when it is received, since there is a certain time period, called the window of opportunity, in which the organizational response is defined.

Workload

Because the human operator is an essential element in the decision making organization, cognitive processing during the execution of the task critically affects performance. The bounded rationality limitation can be modeled by the existence of a maximum processing rate. In general, when the time available to do a task is reduced, the processing rate will increase to finish the amount of work that needs to be done. Analytical and experimental results have shown that when the time is so short that the processing rate reaches the maximum, a further decrease in available time will cause degradation of performance because the amount of work done is less than that required by the task. Therefore, estimating the cognitive workload of DMs is necessary for the designer so that the DMs are not overloaded during the execution of the task.

3.7 PERFORMANCE-WORKLOAD LOCUS

A useful way for describing the properties of the decision maker model, which is generalizable to the properties of an organization, is through the performance workload locus. In the case of a single performance measure, the accuracy measure J, and a single decision maker with workload G, a two-dimensional space is defined with ordinate J and abscissa G. The locus is constructed by considering the functional dependence of J and G on the internal decision strategies of the single decision maker.

Let an internal strategy for a given decision maker be defined as pure, if both the situation assessment strategy p(u) and the response selection strategy p(v|z) are pure, i.e., an algorithm f_i is selected with probability one and an algorithm h_j is selected also with probability one when the situation is assessed as being z_k:

$$D_k = \{p(u=i) = 1 \quad ; \quad \{p(v=j \mid z=z_k) = 1\}\} \tag{3.15}$$

for some i, some j, and for each z_k element of the alphabet Z. There are n possible pure internal strategies,

$$n = U \cdot V^M \tag{3.16}$$

where U is the number of f algorithms in the SA stage, V the number of h algorithm in the RS stage and M the dimension of the set Z. All other internal strategies are mixed (Boettcher, 1981) and are obtained as convex combinations of pure strategies:

$$D(p_k) = \sum_{k=1}^{n} p_k D_k \tag{3.17}$$

where the weighting coefficients are probabilities. Corresponding to each $D(p_k)$ is a point in the simplex

$$\sum_{k=1}^{n} p_k = 1, \quad p_k \geq 0 \ \forall \ k \tag{3.18}$$

The possible strategies for an individual DM are elements of a closed convex polyhedron of dimension n-1 whose vertices are the unit vectors corresponding to pure strategies. For example, let n = 3, then the strategy space will be a two dimensional space (3-1=2), that is, a plane. According to equation (3.18), such a strategy space is a triangle plane which intersects with all three axes at 1 (Fig. 3.13).

The total activity G, the cognitive workload, is a convex function of the decision strategy (Boettcher, 1981), i.e.,

$$G\,(\,D(p_k)) \geq \sum_{k=1}^{n} p_k G_k \tag{3.19}$$

where G_k is the workload that results when the pure strategy D_k, given by equation (3.15), is used.

The accuracy measure J can be related to the decision strategies in a similar manner. Corresponding to each pure strategy D_k is a value of the performance measure, denoted by J_k. Since each strategy is a convex combination of pure strategies, the value of J for an arbitrary $D(p_k)$ is given as a convex combination of the values of J_k, i.e.,

$$J\,(\,D(p_k)) = \sum_{k=1}^{n} p_k J_k \tag{3.20}$$

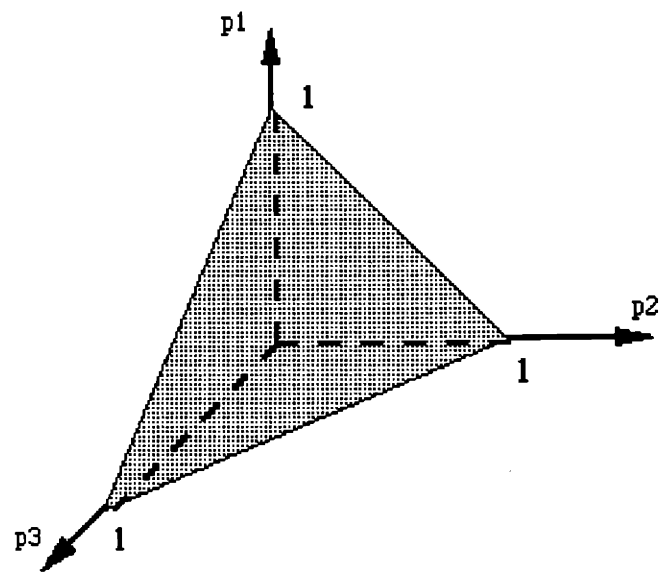


Figure 3.13 Strategy Space for the Case of Three Pure Strategies

The two expressions (3.19) and (3.20) can be used now to determine the locus of points in the (J,G) space that characterize the decision maker.

In the general case, there are n pure strategies, as given by equation (3.16). To each strategy corresponds a value for the accuracy J and a value for the workload G. If J and G are computed for all possible strategies in the strategy space, the performance-workload locus (P-W Locus) is obtained. The P-W locus is constructed as follows:

First, the values of (J_i,G_i) for the n pure strategies are determined. This corresponds to evaluating the performance and the workload for the values of p_k, equation (3.18), that correspond to the vertices of the strategy space. The result is a set of n points in the two-dimensional P-W space.

Then, the binary variations between each possible pair of pure strategies are considered. This corresponds to the mapping of the edges of the strategy space. For example, consider the two pure strategies D_i and D_k. Let $p_i = \delta$ and $p_k = (1-\delta)$, then

$$D = (1-\delta)\, D_i + \delta D_k; \qquad 0 \leq \delta \leq 1$$

for all combinations (i, k) where i = 1,...,n and k = 1,...,n and for which $i \neq k$. By varying δ from 0 to 1, the locus $(J_{ik}(\delta), G_{ik}(\delta))$ is obtained. This convex line joining the two boundary points is shown in Fig. 3.14. Three such loci are shown in Fig. 3.15. These binary loci are quite useful, since they define the minimum workload locus for any feasible value of J.

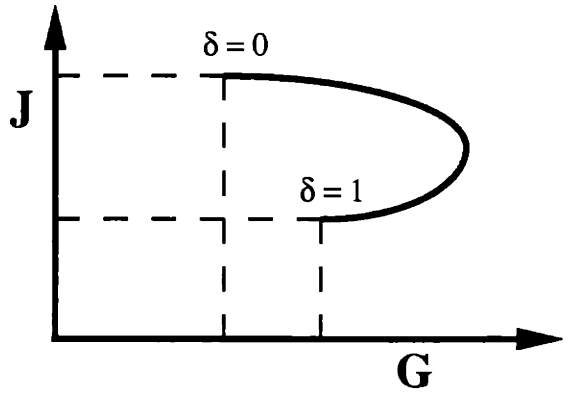


Figure 3.14 Performance-Workload Locus for Example

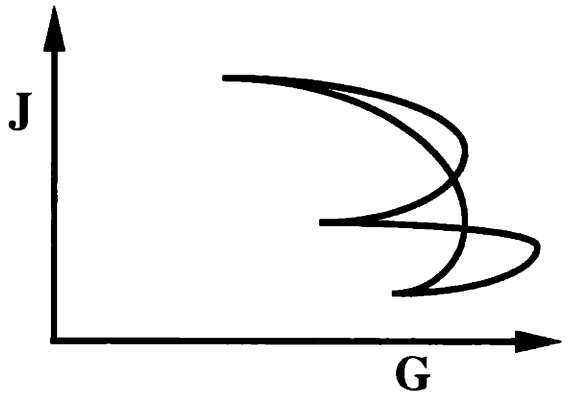


Figure 3.15 Performance-Workload Locus for the Pure and the Binary Strategies

The third step consists of considering, successively, the binary variation between all possible binary strategies until all mixed strategies are accounted for. The result is a locus such as the one shown in Fig. 3.16 for the case when there are three pure strategies.

Thus, the decision maker model can be considered as a system that maps the strategy locus, the simplex defined by equation (3.18), into the Performance-Workload (J,G) locus. Any change in the algorithms f or h, or the functions in IF and CI, or the input x will affect the mapping.

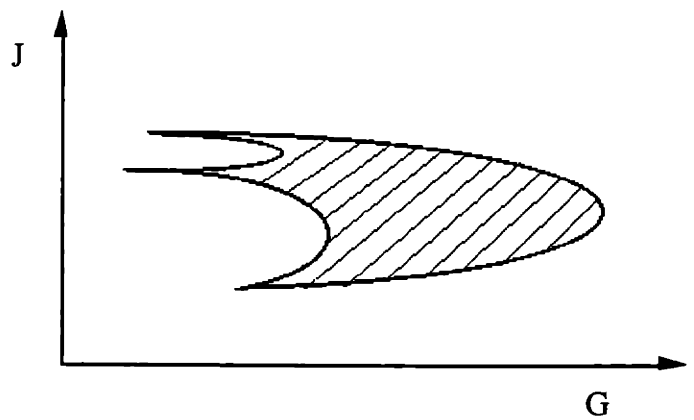


Figure 3.16 Performance-Workload Locus

Note that the mapping from strategy space to performance-workload space is not a one-to-one mapping. Several different strategies may have the same accuracy and workload values.

3.8 SUMMARY

In this chapter, the individual decision maker model and the interacting decision maker model have been described. The task can be modeled as a source which generates inputs to an organization. Workload is defined as the total cognitive activity during information processing and decision making. In the context of distributed decision making organizations that include human decision makers, bounded rationality is one of the critical parameters which affect a human's performance. The bounded rationality can be described by a maximum processing rate beyond which decision makers are overloaded, and consequently, their performance will degrade significantly. Measures of performance considered include accuracy, response time, and cognitive workload of decision makers in the organization. Finally, a performance-workload locus is used to describe the characteristics of organizational performance.

So far, the model of decision makers and the measures of performance have been described. Together with Chapter 2, a theoretical foundation has been established for the design of a model-driven experiment.

Chapter 4

A DESIGN METHODOLOGY FOR MODEL-DRIVEN EXPERIMENTS

In this chapter, a design methodology for model-driven experiment will be introduced. The major stages of the methodology are described. This methodology can be used as a guideline for designing a model-driven experiment; it will be applied to a specific problem in the following chapters.

4.1 OVERVIEW OF THE METHODOLOGY

The methodology is developed to guide the design of model-driven experiments to investigate the organizational behavior of distributed decision making (DDM) organizations. The task for a DDM organization is information processing and decision making. The operation of DDM organizations has the following features.

Distributed decision making organizations operate in an environment which changes dynamically. A change in the environment acts as a stimulus to a DDM organization; the organization senses the stimulus and processes it to infer what the situation is. Then, according to rules and procedures, the organization selects a response to the environmental change.

In many distributed decision making tasks, time constraints play an important role. For a given task, there is only a limited period of time during which the organizational response will be effective. This time period is called window of opportunity. A response produced too early or too late, i.e., outside of the window of opportunity, does not affect the environment. Therefore, in order to respond effectively, the tempo of operations has to adjust to the available time.

Human decision makers are a critical component in DDM organizations. Because of this bounded rationality in processing information and making decisions, organizational performance degrades if human DMs are overloaded.

These three features specify the context in which a model-driven experiment will be conducted.

There are four major stages in the methodology for designing model-driven experiments: 1) Theoretical analysis; 2) Experimental investigation; 3) Experimental data analysis; and 4) Comparison of the theoretical and experimental results. In addition, there is a step for time

scale calibration. This is not considered as a major stage because a small pilot experiment is sufficient to determine the time range for a given task. This time calibration gives the designer useful information about what time scale should be used in the experiment. Figure 4.1 shows the block diagram of the methodology.

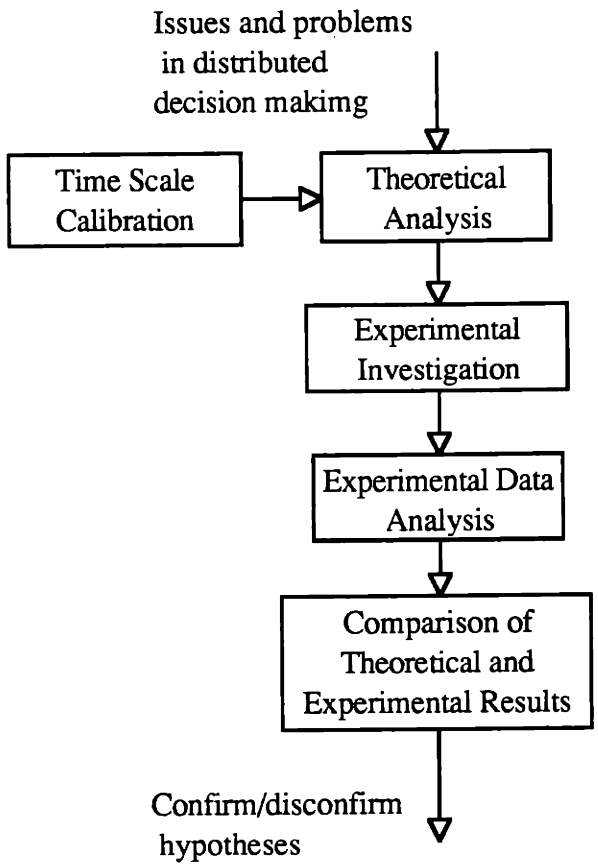


Figure 4.1 The Methodology for Experiment Design

4.2. DESCRIPTION OF THE DESIGN METHODOLOGY

The steps in each of the four design stages of the methodology are described in this section. The implementation of these steps will be shown through an application in the following chapters. It is assumed that the time scale calibration has been done.

Theoretical Analysis

This is the first stage in the experimental design. A task which will be performed by an organization is selected. This task should reflect the problems and issues to be investigated. The same task is used both in the theoretical analysis and in the experiment.

After defining the task, an organization is designed to carry out the task. The design of an organization includes to the determination of the protocol and the procedures to be used in carrying out the task. The organization is modeled using the Petri Net representation described in Chapter 2.

The Petri Net representation of the organization allows the use of computer simulation (MIT/SIM 4.0[1]) for reviewing the operational sequences in the organization. Possible errors in the design, such as deadlocks and conflicts, can be detected through the simulation.

The evaluation procedure, which will be described in the next chapter, is used to obtain performance measures analytically (CAESAR[2]). From these predictions, the Performance-Workload locus (P-W Locus) can be constructed. The characteristics of the P-W Locus lead to the generation of hypotheses on organizational behavior.

Experimental Investigation

This is the second stage in the experimental design. An experiment is designed to test the hypotheses. The complexity of the DDM organization results in a large number of parameters and in much uncertainty regarding their values. To determine the controlled and the measured parameters, dimensional analysis is applied after being extended to include the cognitive aspects of distributed decision making. Dimensional analysis, which will be introduced in the next chapter, is a scientific and engineering method for designing experiments.

In order to carry out the experiment, the range of the controlled parameters needs to be specified. The result from the time scale calibration can be used to estimate a range for the available time. The number of trials for each value of controlled parameters must also to be determined.

A pilot experiment is necessary to test the entire experimental design. Then, the actual experiment is carried out and experimental data are collected.

Experimental Data Analysis

In this stage, the collected data are analyzed and processed to obtain the measures of

---

[1] A noncommercial Petri Net Simulator developed by J.L. Grevet (1988) at MIT.
[2] CAESAR is a noncommercial software package developed at MIT.

performance. The procedures for testing the hypotheses are determined. These procedures are usually statistical ones. To apply the procedures, the hypotheses developed in the theoretical analysis may need to be transformed into an explicit form which can tested directly. All variables involved in the hypothesis testing are gathered and stored in an appropriate format for the test.

Comparison of Theoretical and Experimental Results

This is the last stage in which the hypotheses are tested using the experimental data. The theoretical and experimental results are compared to assess the model's ability to predict organizational behavior. Final conclusions are drawn.

In this chapter, a brief description of a methodology for designing model-driven experiment has been presented. In the following chapters, an application will be used to illustrate the methodology: a multi-person experiment will be designed for investigating the effect of organizational structure on the performance of DDM organizations.

Chapter 5

THEORETICAL ANALYSIS

In this chapter, the first stage of the methodology is applied. The analytical tools of the previous chapters will be applied to two organizational designs to carry out a naval outer air battle (OAB). The procedures and algorithms that will be used to perform the task are presented and explained, and analytical results are described. In Chapter 6, a model of the experiment will be established for the organization and the task described in this chapter.

5.1 A NAVAL OUTER AIR BATTLE

In general, decision making organizations perform large scale complex tasks. Some of these tasks are well defined and structured while others are somehow disordered and random. In order to conduct a rigorous study, one starts with a relatively simple problem to obtain some insight and only then advances to more complex problems. For this reason, defense in a naval outer air battle is chosen as the task to be performed by two small (three person) decision making organizations.

The objective of a naval outer air battle is to monitor incoming enemy aircraft and deploy interceptors to engage threats so as to prevent the enemy from entering the range where missiles can be fired at ships in the battle group.

In this environment, a team of DMs forms an outer air battle group to perform the above task. Specifically, the task of the DMs is to detect incoming enemy aircraft ("threats"); find out the type and the number of threats; then allocate their own aircraft ("resource") to intercept the threats.

Figure. 5.1 depicts a hypothetical naval outer air battle environment. The carrier is at the center of the circles. Airborne warning radar aircraft (E2C) patrol the area at a distance R_p from the carrier. Each E2C commands several squadrons of interceptors, which can then directly intercept the threats. The E2C is equipped with passive radar (ESM) and active radar. Passive radar receives the radar transmission of other aircraft while active radar receives the reflection of its transmission by other objects. The passive radar has a range of R_0 and the active radar has a range of R_a ($R_0 > R_a$). Assume that the range of the enemy's missile is R_m.

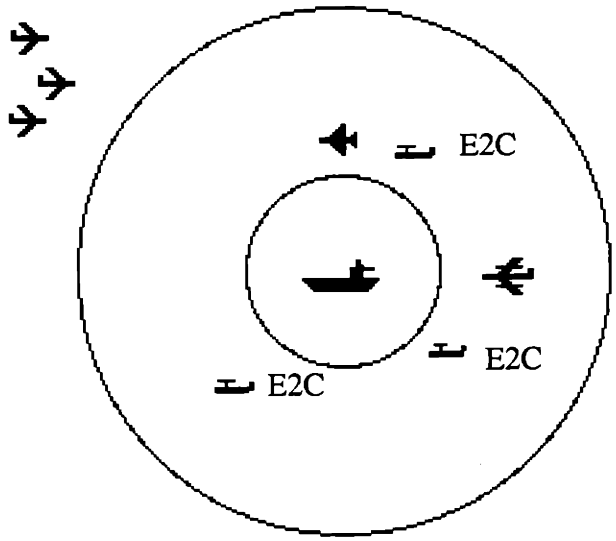


Figure 5.1 A Naval Outer Air Battle Environment

ESM (passive radar) has a larger range for detecting incoming threats but provides less specific data than the active radar: the presence and bearing (direction to) of the threats. The active radar provides more detailed data such as the position and speed of a threat. The signature of an aircraft is provided by ESM when the threat is closer. An emitter signature indicates the existence of an aircraft with its corresponding emitter.

The E2C initially operates only the passive radar to avoid being detected by the enemy's radar. When enemy aircraft approach the E2C, and are within a range R_a, the E2C turns on the active radar. When all of information, speed, emitter signature, and so on, about a threat is available, enemy aircraft can be identified. Correlation between the emitter signature and the speed of the aircraft can be used to classify the type of the aircraft with some level of certainty.

Based on the assessment of incoming threats, the E2C mission commanders allocate resources to intercept the enemy aircraft. The resources are Tomcat fighter aircraft (F14), Hornet fighter/attack aircraft (F18), and Prowler aircraft (EA-6B).

There are situations in which uncertainty and conflict exist. For example, there may be a threat detected by more than one E2C. Then, the question becomes one of determining who is going to deal with it. In a situation like this, coordination between organizational members is necessary. The coordination is done through communication. In addition, the E2C mission commanders may have to communicate with the outer air warfare commander on the carrier to report the situation or to ask him to launch more interceptors. The protocol for communication is different for different organizational structures.

When an enemy aircraft enters its missile range R_m without being engaged by interceptors, the nature of the task changes to the inner air battle, and the outer air battle is over. Therefore, if enemy aircraft are not intercepted before they enter the inner air battle region, the outer air battle defense is considered to have failed.

The above description of the naval outer air battle has been abstracted from actual operation for the purpose of the study. However, the abstraction and simplification are such that it may not reflect the reality of naval operations.

5.2 THE GAME: COMPUTER SIMULATION AND HUMAN OPERATORS

The proposed experiment is based on a simulation of a naval air defense environment. An organization with three decision makers is considered. The objective of the task is to intercept enemy aircraft before they penetrate the circle with a radius R_m, where missiles can be fired against the carrier. The area with radius R_p is defined as the defending area, in which the outer air battle (OAB) will take place. The carrier is located at the center of the coordinates. Decision makers play the role of the mission commanders of E2C patrolling the area to detect incoming threats. To intercept the threats, the speed and the type of the threats have to be determined first. Since one threat may include more than one aircraft, the number of aircraft and type of threat need to be determined for allocating resources, e.g., fighter aircraft, intruder aircraft, and so on. The task is considered to be completed when the interceptors are assigned to encounter the threats.

Human operators interact with the computer simulation of the OAB and make decisions for executing the task. The task is divided into subtasks which are carried out by different DMs in the organization. Each of DMs has a display to observe the OAB situation. The display consists of a simulated radar screen in which threats are displayed and a board for numerical values; a window showing the resource status, and a communications widow displaying the incoming and outgoing messages. In addition, several buttons are present that can be pressed by the DM. The detailed description of the display is in given in Appendix C. The interaction between decision makers necessary for completing the task is realized by communication through computer networks.

To investigate the effects of organizational structure on performance, two different structures are used in the experiment. In each organizational structure, decision makers are organized into teams and then these teams perform the task. The performance of the teams is measured. The following section describes the organizational structures and their models.

5.3 ORGANIZATIONAL STRUCTURES

To perform the task described in the previous sections, two three-DM organizational structures are considered: a parallel structure and a hierarchical structure. In the parallel structure, all DMs are at the same level of authority. They are working together in coordination. In the hierarchical structure, authority varies with the rank of a DM, that is, the position a DM holds in the organization. In both structures, the task is the same; and all members of the organization have to act as a team to perform the task.

In the parallel structure, each one of the three decision makers is the mission commander of an E2C. The defense area is divided into three sectors. Each E2C patrols in one sector and commands several interceptors. When a threat is detected in a sector, the DM who patrols the sector is expected to generate a response. Communication between the DMs is necessary when there is uncertainty about the detected threats. Figure 5.2 shows the block diagram of the parallel structure.

In the hierarchical structure, two DMs play the role of E2C mission commanders as subordinates while the third DM plays the supervisory role of being a commander on the carrier (AAW: Anti-Air Warfare commander). The task for E2C mission commanders is the same as in the parallel structure except that they do not communicate directly with each other, but both report to the AAW commander. The AAW commander does not observe the defense area directly, but receives reports from the E2C mission commanders, assesses the global situation, and then issues commands to the E2C mission commanders who do the local resource allocation. Figure 5.3 shows the block diagram of the hierarchical structure.

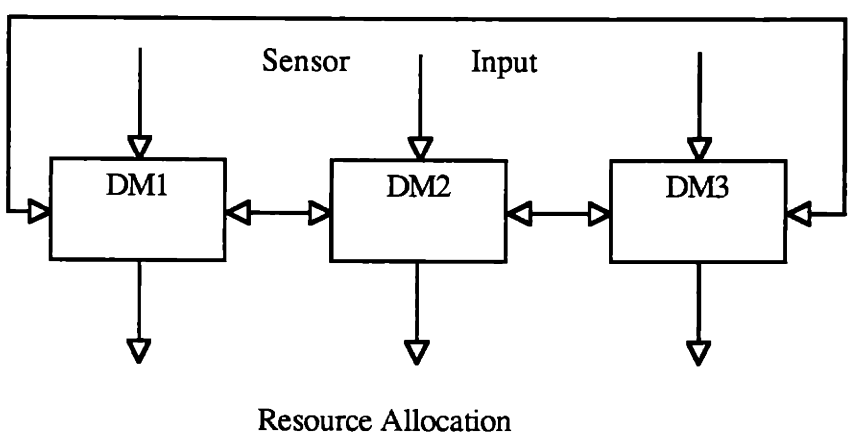


Figure 5.2 Parallel Organization

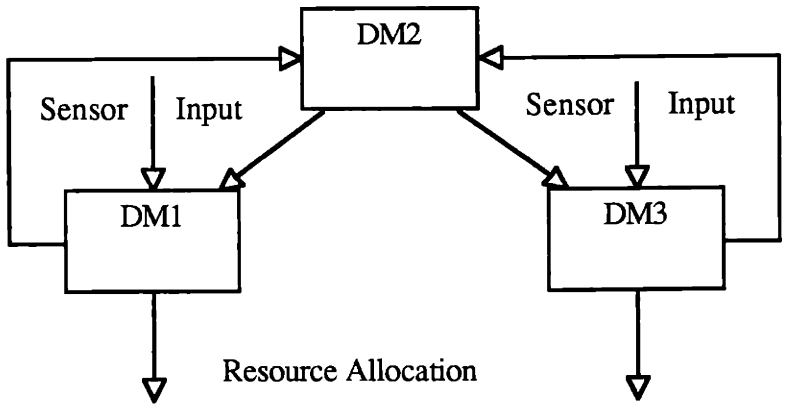


Figure 5.3 Hierarchical Organization

5.4 PROCEDURES FOR PERFORMING THE TASK

The general procedure for performing the task is the same for both organizations. However, there are several differences that depend on the structure. In this section, the general procedure is discussed first, then the specific procedures for the parallel and hierarchical structures are described.

The procedure is as follows. When threats are detected, the situation has to be assessed. The situation assessment (SA) function provides information such as the number of threats, the position and speed of the threats, the type of the threats, and the number of aircraft in each threat. The situation assessment involves data gathering and processing because some of the information can be directly obtained from the observed data while other information is available only after the raw data are processed. Depending on the particular situation, communication may be required after the situation assessment. The results of the communication are processed in the information fusion stage for the parallel structure and in the command interpretation for the hierarchical structure. Following the fusion stage is the response selection stage. On the basis of the fusion data results, resources can be allocated to counter the threats. The task is completed after resources are allocated. The following paragraphs describe in detail the procedures for each structure.

Procedure for Parallel Organizational Structure

In the parallel organizational structure, the defense area is divided into three sectors. Each DM is an E2C mission commander and is responsible for one sector, that is, this DM is

responsible for all and only threats in the sector. There are overlap areas between the sectors. In Fig. 5.4, the solid straight lines in the radar screen divide the defense area into three sectors. The area of responsibility of a DM is the white sector bounded by two solid straight lines. The areas bounded by dotted lines are the overlap areas between the sectors. Each DM can see a part of the other two sectors. Therefore, the area that can be seen by a DM, defined as the observation area, is shown by the area without gray shading. Any threat in the gray shaded area cannot be seen by this DM and is out of the region of responsibility of this DM.

It should be clear that there are two areas for each DM in which the responsibilities are different. One is the observation area, the white area in Fig. 5.4, which includes the sector and the overlap areas of other two adjacent sectors. Another is the sector, bounded by solid lines in Fig. 5.4. The threats in the observation area can be detected and the information about these threats can be obtained. However, a DM can only allocate resources and intercept the threats in *his* sector.

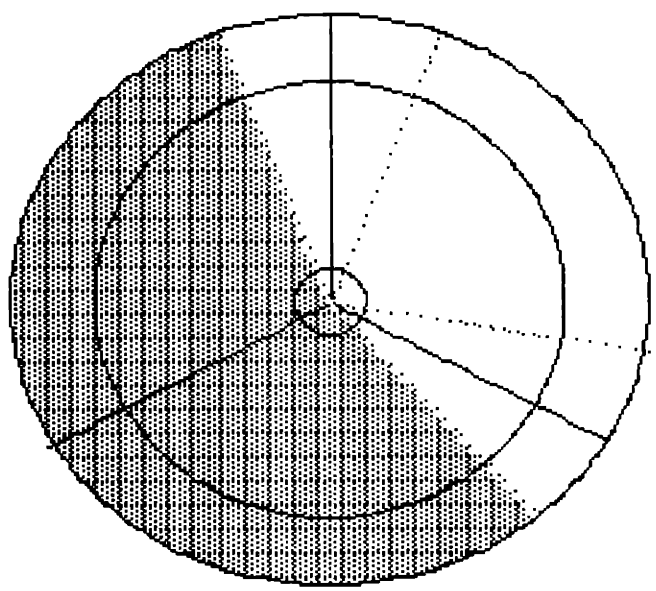


Figure 5.4 Defense Area Divided into Three Sectors

The threats which are not in the overlap area can be processed without exchanging information with the adjacent DMs because only the local DM can observe these threats. For the threats in the overlap area, partial information is received, therefore, the coordination with other DMs in the team is necessary. The coordination is through communication. In practice, the procedure for coordination is quite complex. In this experiment, the procedure is simplified so that it is controllable and so that it serves the purpose of the experiment. The two cases when threats are in the overlap area are discussed in the following paragraphs. To avoid

confusion, the decision makers who are responsible for sector 1, sector 2, and sector 3 are named DM1, DM2, DM3, respectively.

Case 1: For threats in the overlap areas *and* in sector i, DMi monitors the threats and assesses the situation according to the data available to him. But, the DMi cannot proceed with allocating resources and intercepting the threat until he receives information from the adjacent DMj who does the situation assessment according to the data available in sector j. In other words, DMi has to wait for the information from DMj who can also observe the threat. After fusing the results from DMj with his own, DMi can continue the processing of the threat.

Case 2: For threats in the overlap areas *but* not in sector i, the DMi does the situation assessment, then sends the result to the corresponding DM who can intercept the threat *and* who cannot allocate resources to the threat until he receives the information.

In both cases, coordination is required.

The task and the procedure for all of the DMs in the parallel structure are the same. All of the DMs can communicate with each other. However, the communication occurs only when there is a threat in the overlap area.

Procedure for Hierarchical Organizational Structure

In this organization, the defense area is divided into two sectors as shown in Fig. 5.5. One DM is the commander on the carrier; he performs a supervisory role and coordinates between the two sectors. The other two DMs are E2C mission commanders and play the role of subordinates. Only subordinates can observe the defense area directly. As in the parallel organization, each subordinate is responsible for monitoring and intercepting the threats in one sector. If there is any conflict, that is, threats are in the overlap area, subordinates have to report the situation from their perspective to the supervisor. The following paragraphs describe the roles for the subordinates and the supervisor.

Subordinates

The threats in a sector and not in the overlap area can be processed without communicating with the supervisor. When a threat is in an overlap area, whether in the sector or not, a subordinate has to perform the local situation assessment and report to the supervisor. Then he has to wait for commands from the supervisor before prosecuting this threat. The command input from the supervisor contains the type of threat and the resource selected for engaging the threat. After receiving the command, a subordinate interprets the commands (CI) by combining the local situation and the commands, then makes the final decision on the resource allocation (RS).

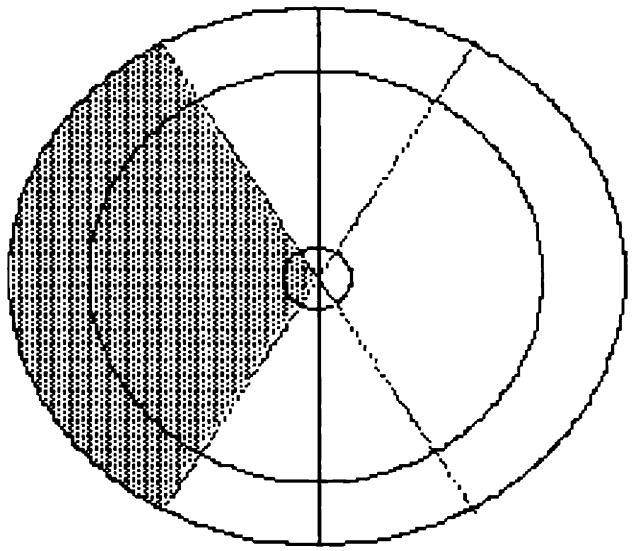


Figure 5.5 Defense Area Divided into Two Sectors

Supervisor

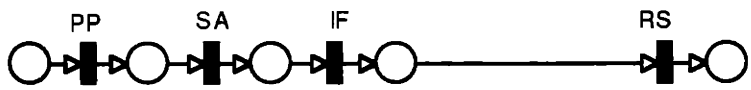
The supervisor receives the reports on the local situation from both sectors. Therefore, the supervisor has an overview of the entire situation, and he can do the global situation assessment (GSA) to determine the type of threats in the overlap area. The global resource allocation, or more generally, global response selection (GRS) is the selection of resources to counter the threats in the overlap area. When GRS is done, the supervisor assigns the threat to an appropriate sector from which the threat should be attacked. The result, then, is transmitted to the subordinate in the chosen sector as a command.

The interaction level in the hierarchical organization is higher than that in the parallel organization. In the next section, the execution of the OAB task will be modeled by a sequence of functions. The functions and algorithms for performing the functions will be described.

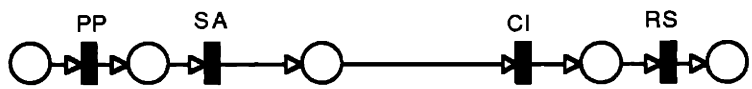
5.5 DESCRIPTION OF FUNCTIONS AND ALGORITHMS

The task of the organization is distributed to three organizational members. Each DM does a part of the task, called subtask, and contributes to the organizational performance. The subtask can be further decomposed into functions. The functions are defined so that they preserve the features of the naval outer air battle. Then, algorithms to implement these functions can be designed. There are five basic functions: preprocessing (PP), situation assessment (SA), information fusion (IF), command interpretation (CI), response selection (RS), and post processing (POP). The decision making process of each decision maker can be

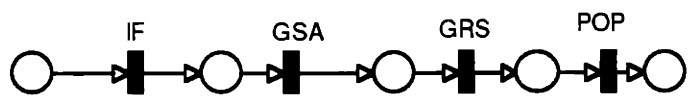
modeled by connecting some of these functions together. However, it is not necessary to have all of them in each DM. Figure 5.6 shows the basic models for DMs who play different roles in different organizations.



(a) DM in the Parallel Organization



(b) Subordinate in the Hierarchical Organization



(c) Supervisor in the Hierarchical Organization

Figure 5.6 Model for DMs in Different Organizations

Preprocessing

The first stage is the preprocessing stage (PP). At this stage, a DM looks at the overall situation in the area of responsibility. Specifically, a DM can assess the data to find out the number of threats and classify threats according to their speed. At this stage, a DM will have a rough idea about the task's difficulty. The input to this function is the speed, v, and the constant bearing, θ, of incoming threats. The input is denoted by a vector x,

$$x = \begin{bmatrix} x1 \\ x2 \end{bmatrix} = \begin{bmatrix} \text{Speed, } v \\ \text{Bearing, } \theta \end{bmatrix}$$

Although the number of threats in a *sector* is constant, the total number of threats in an *observation area* varies depending on the number of threats in the areas that overlap with the sectors of the adjacent DMs. The bearing of a threat determines whether the threat is in the

overlap area. Let N denote the number of threats in an observation area. Assume there are three possible values of N, that is, N can take a value of N1, N2, or N3, or,

$$N = \{ N1, N2, N3 \}$$

The class of a threat is defined by its speed. Three classes are considered: fast (f), medium (m), and slow (s). The flowchart of this function is shown in Fig. 5.7.

In Fig. 5.7, v_s denotes the upper limit of the speed of the slow threats, e.g., x2' = s, while v_f represents the lower limit of the speed of the fast threats, e.g., x2' = f. Speeds in between these two limits are classified as medium, e.g., x2' = m. The output of the function is the number and class of the threats. The output of the PP stage is represented by a vector x' which is

$$x' = \begin{bmatrix} x1' \\ x2' \end{bmatrix} = \begin{bmatrix} \text{Number of threats} \\ \text{Class of threats} \end{bmatrix}$$

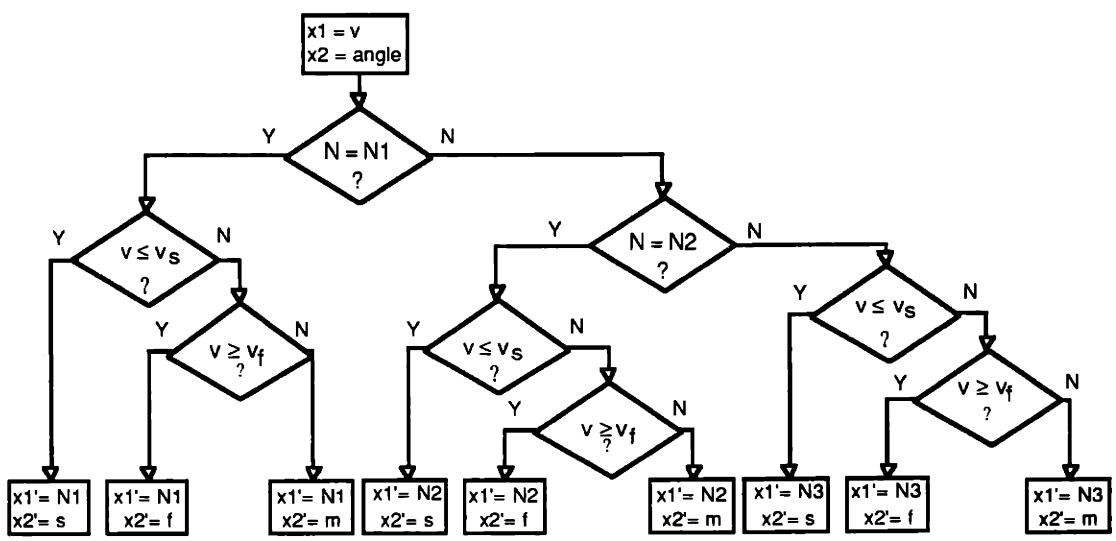


Figure 5.7 Flowchart for Preprocessing Stage (PP)

Situation Assessment

The type of the threats is determined at the situation assessment (SA) stage. The type is determined by the class and the emitter signature of the threat. To simplify the implementation, the number of aircraft in the threat substitutes for the emitter signature. Therefore, the type of the threat is determined by the class and the number of aircraft.

The inputs to this function are the class and the number of aircraft, n, in the threat. The relation between these two parameters and the type is shown in Table 5.1. The output of the function is the type of the threats, that is, fighter (F), bomber (B), or surveillance aircraft (S). As we can see in Table 5.1, there are cases in which the type of threat cannot be uniquely determined. This reflects the lack of adequate information. In this case, the type may be estimated according to the probability distribution of the type of threats. A more accurate way is to probe and acquire the information. However, data acquisition requires extra time. It follows that there are different algorithms that can be used to determine the type of the threat. The basic algorithms are: (a) quick estimation with attendant risk of errors and, (b) accurate acquisition but with time delay. Which one should be chosen depends on a particular situation, i.e., on the level of uncertainty, on time available, and so on.

Th SA stage can be divided into three subfunctions which are described in Table 5.2.

Table 5.1 Type of Threats

Number\class	x'2 = Fast	x'2 = Medium	x'2 = Slow
n < 5	F	F,S	S
n = 5	F, B	F, B, S	B, S
n > 5	B	B	B

Table 5.2 The Subfunctions in the SA Stage

Name	Function	Input	Output
f1:	Determines if the threat is in the overlap area and in the sector;	Entering bearing of the threat	1. If the threat is in overlap area, z1 2. If it is in the sector, z2 3. Whether to wait for message, z3.
f2:	Determines the type of the threat;	1.Class of the threat; 2.Number of aircraft in the threat.	Type of the threat, z4
f3:	Determines if interaction is required, that is, if the result of this stage should be sent to other DMs.	Output of f1	Whether communication should occur, z5

The values for the assessed situation z in Table 5.2 are as follows. These values are also used in the flowcharts representing the subfunctions.

$$z1 = \begin{bmatrix} 1, & \text{in overlap area;} \\ 0, & \text{not in overlap area.} \end{bmatrix}$$

$$z2 = \begin{bmatrix} 1, & \text{in this sector;} \\ 0, & \text{not in this sector.} \end{bmatrix}$$

$$z3 = \begin{bmatrix} 1, & \text{waiting for message;} \\ 0, & \text{not waiting for message.} \end{bmatrix}$$

$$z4 = [\text{ Bomber (B), Fighter (F), Surveillance aircraft (S) }]$$

$$z5 = \begin{bmatrix} 1, & \text{communicate;} \\ 0, & \text{do not communicate.} \end{bmatrix}$$

For example, when z1 = 1 and z2 = 1, the threat is in the overlap area and in the sector. A DM in the parallel organization has to wait for a message from the other DM before he can proceed to the next function. In this case, he has an option: he may want to do situation assessment first then wait for the message or he may wait for the message before doing the situation assessment. When z1 is equal to zero (z1 = 0), then the threat is not in the overlap area, and a DM is free to proceed without any interaction.

Figure 5.8 shows the flowcharts for f1.

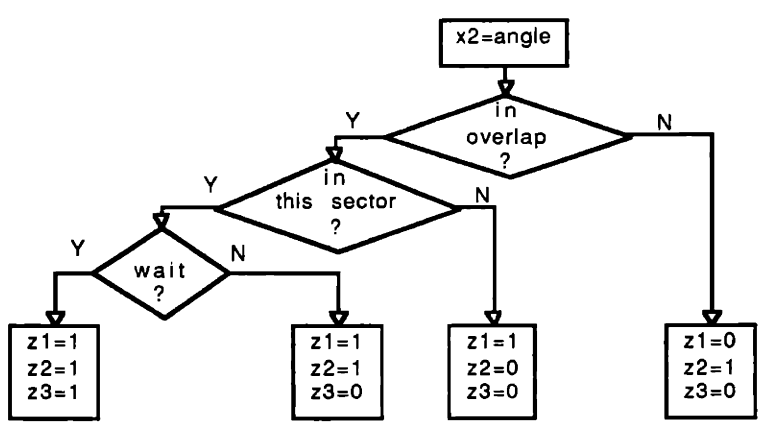


Figure 5.8 Flowchart for Situation Assessment : f1 - Location of Threat

f2: Algorithm for Estimating the Type of a Threat

Figure 5.9 shows the flowchart for this algorithm. Input to this algorithm is the class of a threat and number of aircraft in the threat. The type of threat is determined according Table 5.1. When there is not a unique choice, uncertainty exists. An estimate may be made on the basis of the probability that each type will occur.

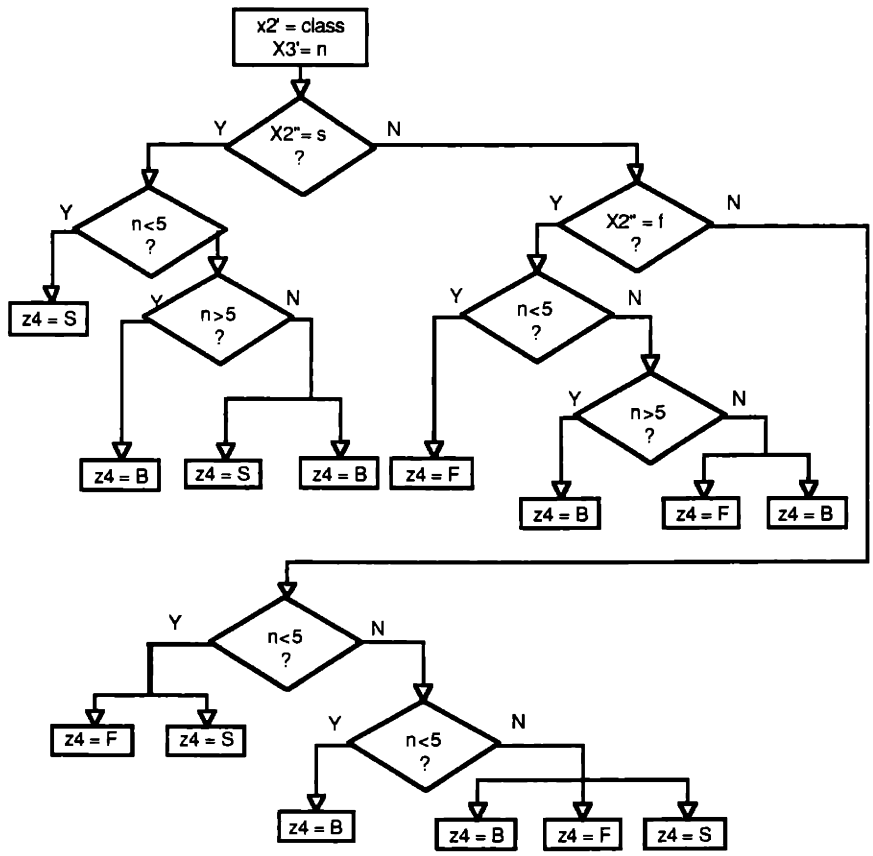


Figure 5.9 Flowchart for Situation Assessment : f2 - Estimation

f2: Algorithm for Probing the Type of a Threat

This algorithm allows to probe for accurate information on the type of threat. To initiate this algorithm, the probing command is selected. The input to the algorithm is the class and the number of aircraft in the threat. There is some time delay associated with this algorithm. The output is the type of the threat. Figure 5.10 is the flowchart for this algorithm.

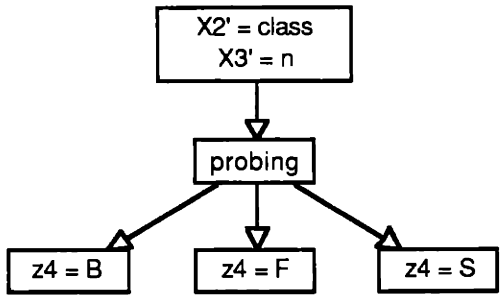


Figure 5.10 Flowchart for Situation Assessment : f2 - Probing of Threat

Figures 5.11 and 5.12 show the function f3 in the SA stage for the parallel organization and the hierarchical organization, respectively. The flowcharts depict the different rules for interaction in the parallel organization and in the hierarchical organization,

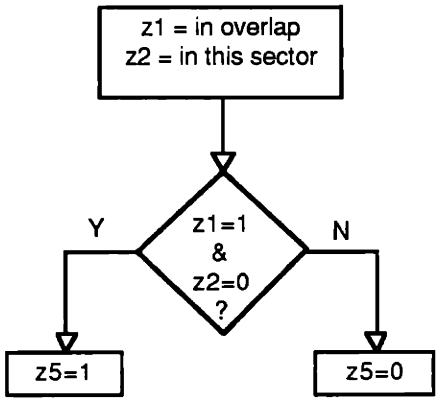


Figure 5.11 Flowchart for SA-f3 of a DM in the Parallel Organization

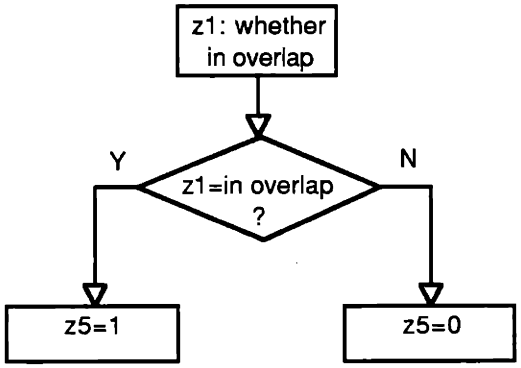


Figure 5.12 Flowchart for SA-f3 of a Subordinate in the Hierarchical Organization

Information Fusion

Information fusion (IF) is present only in the parallel structure. At the information fusion stage, a DM receives information about a threat which is also detected by other organization members. He combines it with his own information from his SA stage to obtain complete information about that threat.

The input to the fusion stage is the type of the aircraft determined in his own SA stage and the type determined by another DM in the organization. The output of the fusion function is the type of the aircraft and how an interaction has taken place. Figure 5.13 shows the flowchart for the IF function. The input is denoted by z

$$z = \begin{bmatrix} \text{type of a threat determined internally, } z4; \\ \text{type received from other DM, } z_m. \end{bmatrix}$$

The output is represented by z'

$$z' = \begin{bmatrix} \text{type of a threat, } z'1; \\ \text{interaction: } z'2 = \begin{bmatrix} 1, \text{ accept the threat's type as received;} \\ 0, \text{ do not accept.} \end{bmatrix} \end{bmatrix}$$

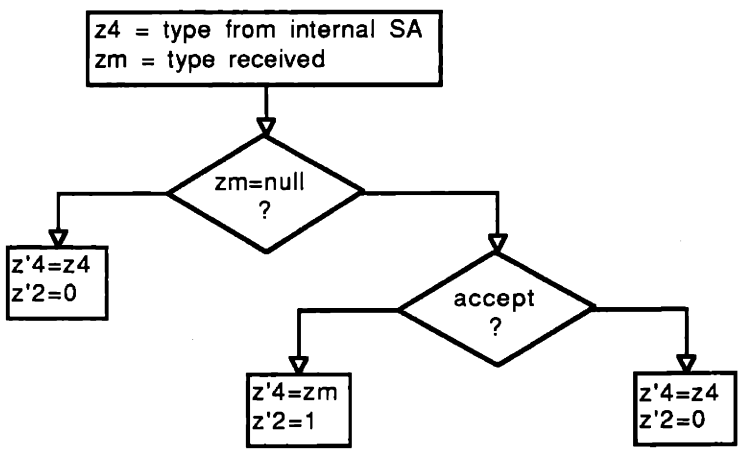


Figure 5.13 Flowchart for Information Fusion (IF)

Command Interpretation

The Command Interpretation (CI) function is present only in the subordinates in the hierarchical organization. At this stage, subordinates receive commands from the supervisor. The command contains guidance on the response selection for the threat.

The input to this function is a vector, z_C, which contains the type of the aircraft in the threat and the number of each kind of interceptor that may be allocated to the threat. On the basis of the local situation and availability of free interceptors, a subordinate can adopt the command exactly as received from the commander or he may modify it to suit the local situation. The output is whether the command has been adopted as is or whether it has been modified. Figure 5.14. shows the flowchart.

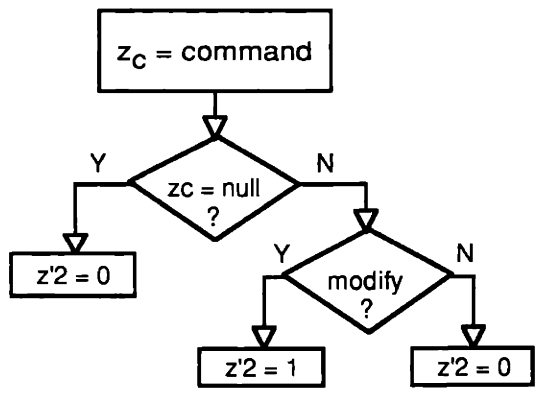


Figure 5.14 Flowchart for Command Interpretation (CI)

Response Selection

This is the final stage of the process (except for the supervisor in the hierarchical organization). At this stage, the resources (the interceptors) are allocated to the threat according to the type of threat and number of aircraft in the threat. The choice of resources is not unique. The objective is to allocate the correct type and number of resources to intercept the threat. After selecting the resources, directives are sent to the selected interceptors assigning them to threats. Table 5.3 is the Threat-Resource table which shows, for each type of threat, the number of enemy aircraft that each resource can intercept successfully.

There are two algorithms that can be used to allocate resources: "estimation" and "probing". Just like the algorithms in the SA stage, the tradeoff between the two algorithms is time and accuracy. The flowchart for the estimation algorithm is shown in Fig. 5.15. There are different resource-threat relations. The equations shown in the flowchart are based on allocations shown in Table 5.3 and are used to determine the actual resource allocation. The solution is not unique. The results are correct as long as they are satisfy the appropriate equation.

Table 5.3 Threat-Resource Table

Type of threat \ Resource	F14	F18	EA-6B
Bomber	3	2	1
Fighter	2	1	0
Surveillance aircraft	4	3	1

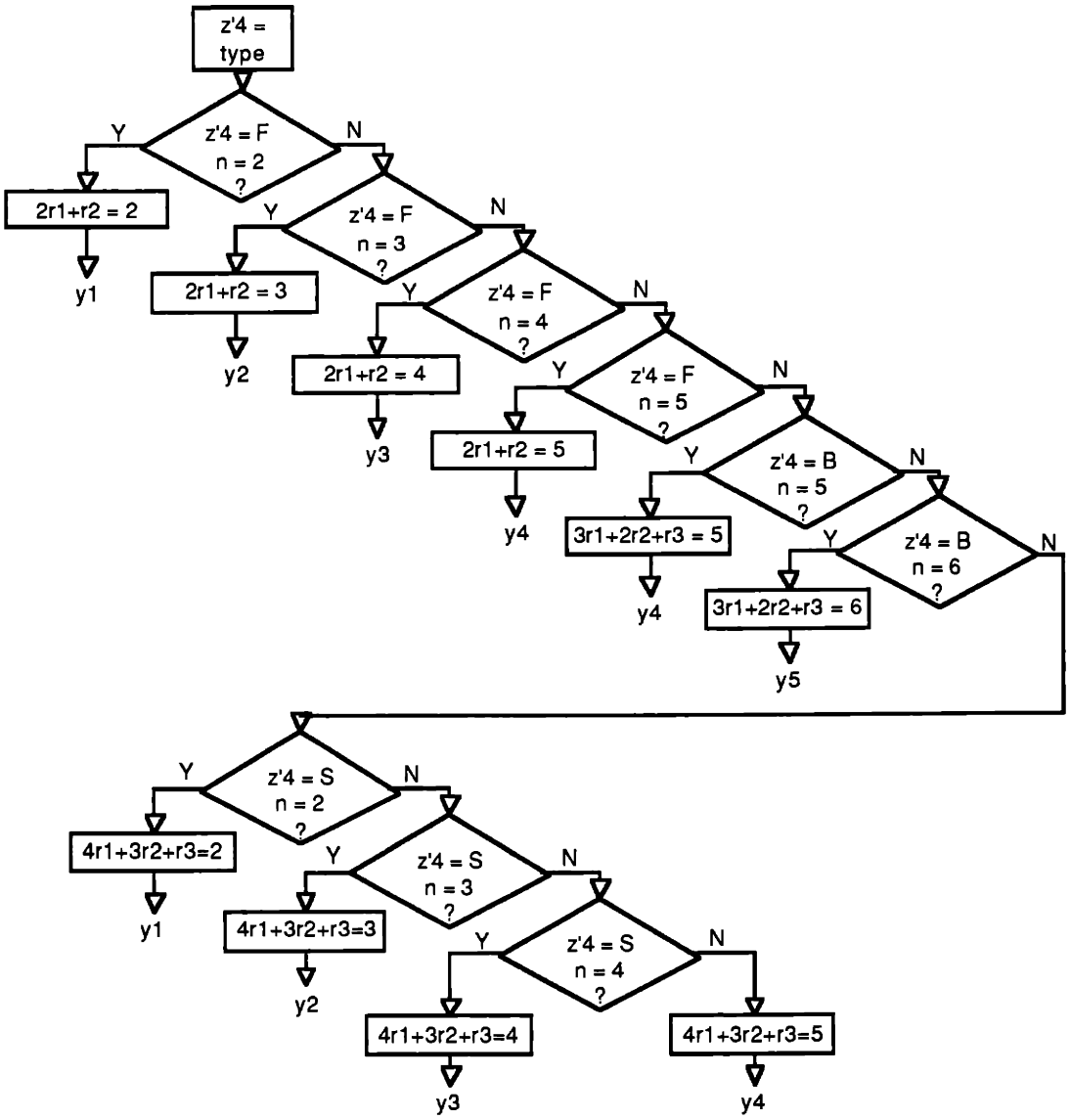


Figure 5.15 Flowchart for Resource Allocation: Estimation

The input to the algorithms are the type of a threat and the number of aircraft in the threat. The output of the algorithms is the result of resource allocation, denoted by y, which indicates the number of each type of the resources selected to intercept the threat.

Figure 5.15 shows the flowchart for the estimation algorithm. In Fig. 5.15, n is the number of aircraft belonging to a threat; r_1, r_2, and r_3 are the number of F14, F18, and EA-6B aircraft, respectively, allocated to the threat. For example, if the threat type is bomber (B) and the number of bombers is 5 (n = 5), then one possible resource allocation can be $r_1 = 1$, $r_2 = 1$, and $r_3 = 0$. Substituting these values in the equation corresponding to the bomber ($3r_1+2r_2+r_3 = n$), it can be seen that the selection of resources satisfies the equation.

Each y in Fig. 5.15 corresponds to a set of vectors of dimension three, which are the possible outputs of the RS stage. In general, the entire vector space of dimension three is the sample space from which a specific resource selection can be drawn. However, only a small portion of this vector space corresponds to the correct response selection. Therefore, the sample space is reduced by the following assumption. It is assumed that the total number of resources selected to intercept a threat cannot be larger than the number of aircraft in that threat. This assumption results in a sample space of 82 vectors.

Furthermore, this sample space is divided into subspaces which will satisfy the assumption for different values of n, the number of aircraft in a threat. Let n take one of five values:

$$n = \{ 2, 3, 4, 5, 6 \}$$

Then, there are five subspaces, denoted by yn, each corresponding to a value of n:

y1	corresponding to n = 2
y2,	corresponding to n = 3
y3,	corresponding to n = 4
y4,	corresponding to n = 5
y5,	corresponding to n = 6.

Figure 5.16 shows the flowchart for the probing algorithm. In this algorithm, a DM only needs to know the type and the number of aircraft of the threat to probe for the correct resource allocation.

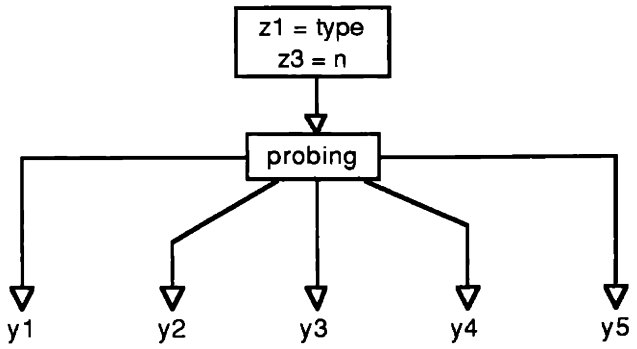


Figure 5.16 Flowchart for Resource Allocation: Probing

Post Processing

Post processing (POP) is only done by the supervisor of the hierarchical organization. After selecting the resource, the supervisor has to decide to which sector (DM) the task should be allocated. The input to this function is the position of the threat. The output is the sector number where the task is assigned. Figure 5.17 shows the flowchart of this function.

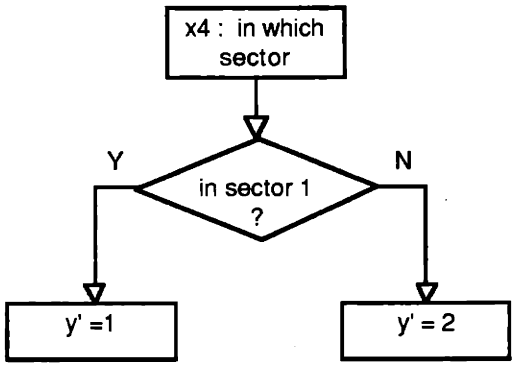


Figure 5.17 Flowchart for Post Process: POP

The functions in all stages of information processing and decision making have been described. Different algorithms for the SA and the RS stage have been introduced. These functions and algorithms will be used in the theoretical analysis and the simulation to predict the organizational performance and will be used for the experimental design.

5.6 PETRI NET REPRESENTATION OF THE ORGANIZATIONS

The models for individual decision makers are connected to create a model for the organization. The connections between DMs are made in accordance with the interaction protocol of the organization. There are different protocols for the parallel and hierarchical organizations. Figures. 5.18 and 5.19 represent the Petri Net models for the parallel and hierarchical organizations, respectively.

The input is partitioned into segments, as described in Section 3.3, each going to a corresponding DM in the organization in accordance with the location of the threat. Recall that each threat is characterized by a vector

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \text{speed, } v \\ \text{Bearing, } \theta \end{bmatrix}$$

Then the partition is represented by a set of partition matrices, π^i, one such 2x2 matrix for each sector i. Whether a particular threat x will appear in sector i is determined by

$$\underline{x}^i = \pi^i x.$$

For the parallel organization, there are three sectors. Each sector is 120 degrees (120°). But there are overlap areas between the sectors. The overlap areas are 40° (20° in each sector). The three partitions are

$$\pi^i = \begin{bmatrix} \pi^i_{11} & 0 \\ 0 & \pi^i_{22} \end{bmatrix}$$

where

$$\pi^i_{jj} = \begin{cases} 1, & \text{if } \frac{360}{3}(i-1) - 20 \le \theta \le \frac{360}{3} i + 20 \\ 0, & \text{otherwise} \end{cases}$$

for i = 1, 2, and 3 and j = 1, 2.

For the hierarchical organization, there are two equal sectors of 180° each. The overlap areas are 60° (30° in each sector).The partition matrices are

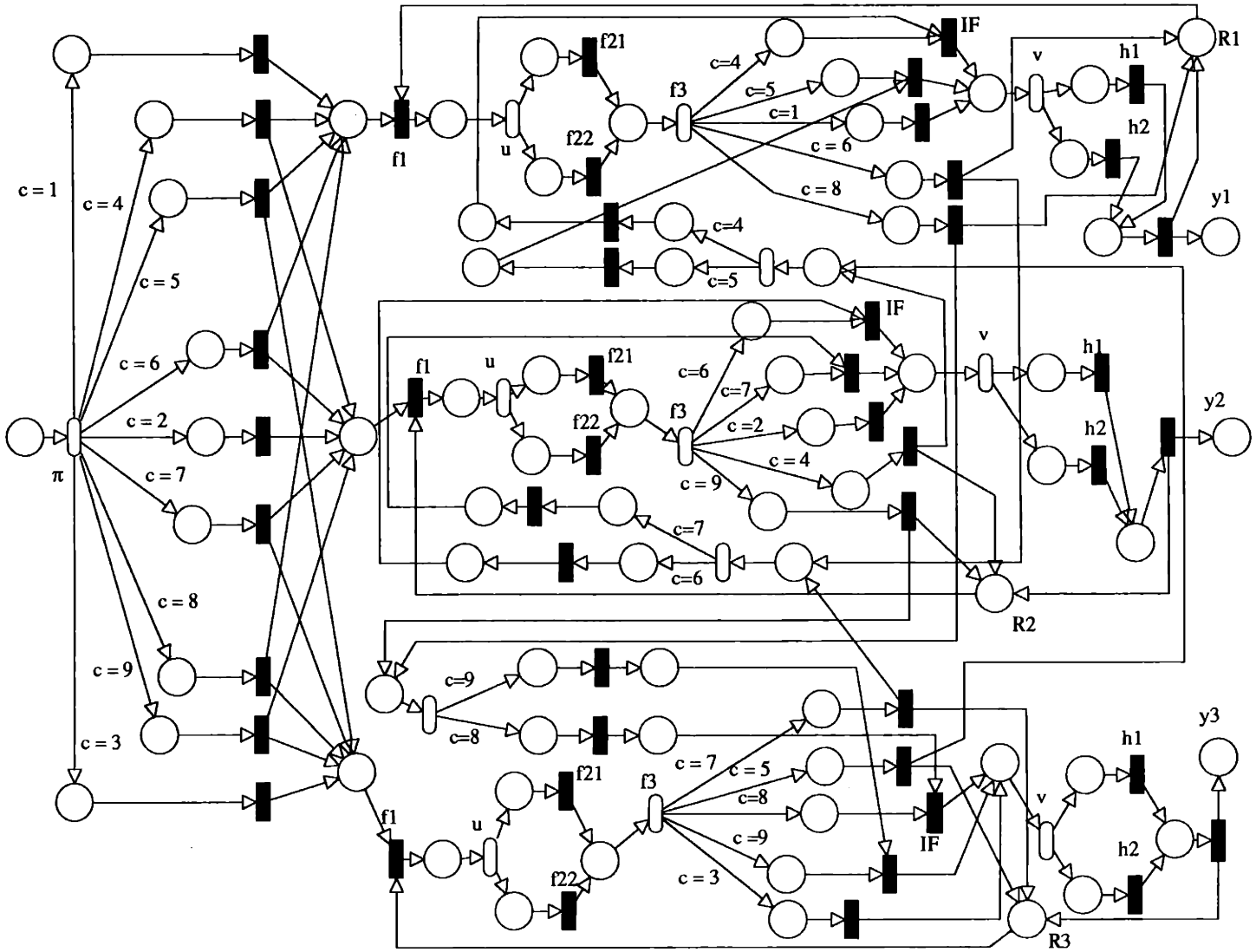


Figure 5.18 Petri Net Representation of the Parallel Organization

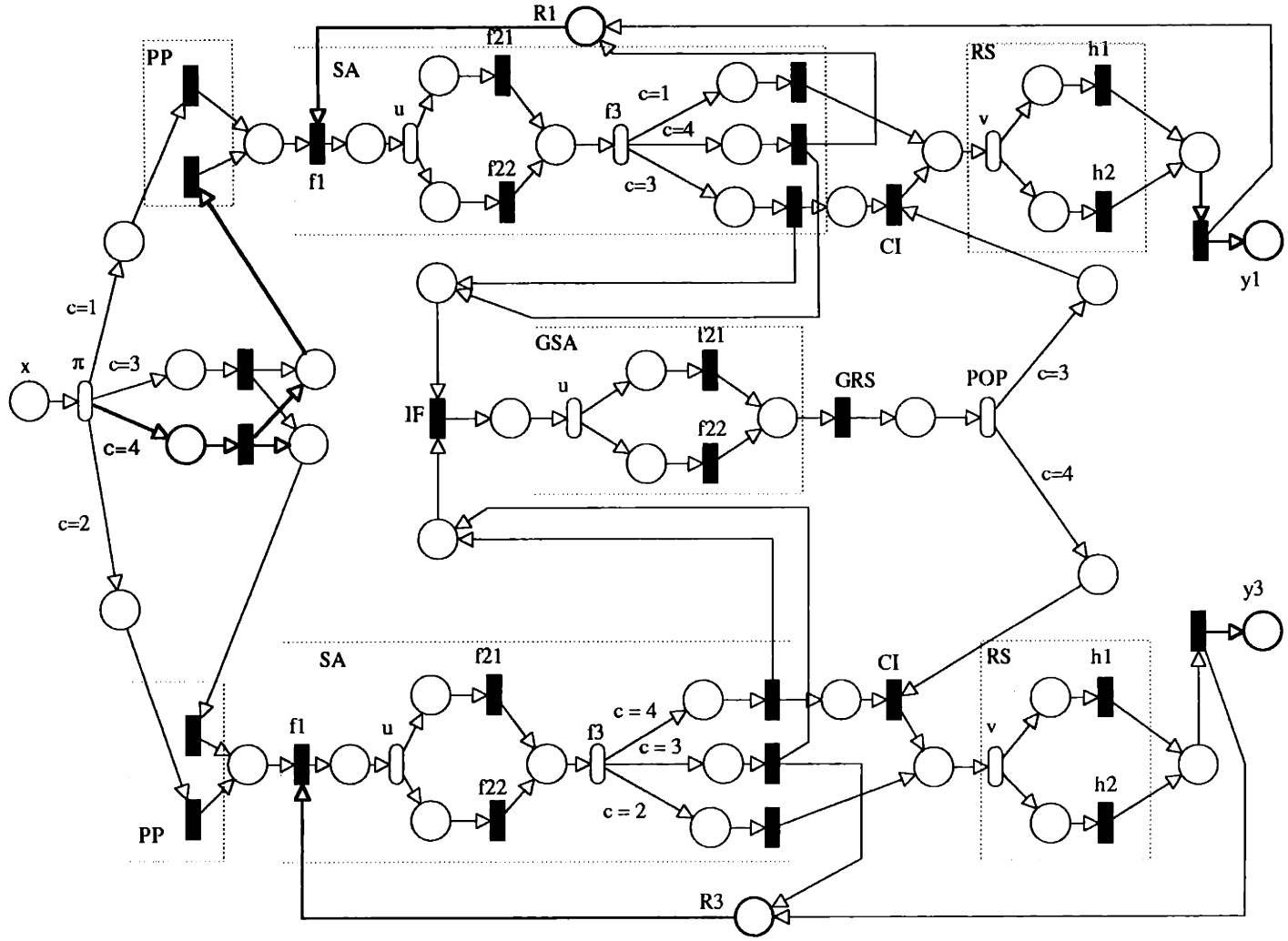


Figure 5.19 Petri Net Representation For the Hierarchical Organization

$$\pi^{i}_{jj} = \begin{cases} 1, & \text{if } \frac{360}{2}(i-1) - 30 \le \theta \le \frac{360}{2} i + 30 \\ 0, & \text{otherwise} \end{cases}$$

for i = 1 , 2 and j = 1, 2.

The supervisor in the hierarchical organization does not interact with the external data directly. He only receives data from the subordinates who observe changes in the environment. Therefore, he has no partition matrix. The supervisor does not react to threats directly, but issues the commands to direct the subordinates. All actions for intercepting the threats are taken by the subordinates. It can be seen from Fig. 5.19 that there is no connection between the supervisor and the input and output transitions.

Decision switches are used for two purposes. One is to represent the availability of different algorithms to process the task in both SA and RS stages. For example, the switches with labels u and v in Figs. 5.18 and 5.19 are called strategy switches. As the name suggests, a strategy switch indicates that different strategies can be used for doing a task: f21, f22, h1, and h2 are different algorithms that can be selected during the execution of the task. There are two strategy switches in each DM model in the parallel organization and in each subordinate's model in the hierarchical organization, while there is only one strategy switch in the supervisor's model.

Another usage of the switch is for modeling procedures. For example, the partition of the input is modeled by a switch π. Threats are sent to different sectors according to their bearing. The switches labelled f3 in Figs. 5.18 and 5.19 are the switches representing the procedure. The labels of the output arcs of these switches specify the path to be followed by a particular type of information. In terms of Petri Nets, a label indicates what type of tokens may go through the arc. Token type depends on the input attributes. For example, a label of c = 1 means that only type 1 tokens can pass through this arc.

The switches labelled COM in Fig. 5.18 and POP in Fig. 5.19 also model a procedure. Switch COM is used to process incoming message for information fusion while switch POP is to determine to which sector the command should be sent. The meanings of the labels shown in Figs. 5.18 and 5.19 are listed in Tables 5.4 and 5.5.

The output of the organizations is produced by individual decision makers and is denoted by y. Following the output transitions, there is a resource place added to each DM to allow DMs process more than one threat. These places must have an initial marking when the Petri Net model is simulated. The software package MIT/SIM is used to simulate the Petri Nets to assure the correctness of the model.

Table 5. 4 Token Type for Hierarchical Organization

Token Type	Meaning
1	a threat is in sector 1 and *not* in overlap area
2	a threat is in sector 2 and *not* in overlap area
3	a threat is in sector 1 and *in* overlap area
4	a threat is in sector 2 and *in* overlap area

Table 5.5 Token Type for the Parallel Organization

Token Type	Meaning
1	a threat is in sector 1 and *not* in overlap area
2	a threat is in sector 2 and *not* in overlap area
3	a threat is in sector 3 and *not* in overlap area
4	a threat is in sector 1 and can be seen in sector 2
5	a threat is in sector 1 and can be seen in sector 3
6	a threat is in sector 2 and can be seen in sector 1
7	a threat is in sector 2 and can be seen in sector 3
8	a threat is in sector 3 and can be seen in sector 1
9	a threat is in sector 3 and can be seen in sector 2

The output of the organizations is produced by individual decision makers and is denoted by y. Following the output transitions, there is a resource place added to each DM to allow DMs process more than one threat. These places must have an initial marking when the Petri Net model is simulated. The software package MIT/SIM is used to simulate the Petri Nets to assure the correctness of the model.

In the parallel organization, the interaction takes place at the information fusion stage. Each DM sends the result of his SA stage to the other DMs. The results from different DMs are then fused to obtain a more complete assessment of the situation. Response is selected locally.

In the hierarchical organization, there are two modes of interaction. First, the subordinates report the local situation to the supervisor. The supervisor does the global situation assessment and response selection and then sends the result to the corresponding subordinate. The subordinate waits for the command before proceeding with his resource selection. When he receives the command, the subordinate has to interpret the command and then make a decision regarding the local response selection.

The interaction level is higher in the hierarchical organization than in the parallel organization. A comparison of Figs. 5.18 and 5.19 shows that the path of an input going through the organization contains more processes (transitions) in the hierarchical organization than in the parallel one. In other words, the information flow paths involve more interaction between organizational members in the hierarchical organization than in the parallel organization.

5.7 DECISION STRATEGIES AND TASK WORKLOAD

5.7.1 Decision Strategies

As described in Section 5.4, both the Situation Assessment stage and the Response Selection stage contain two algorithms. Assume that the choice of the algorithm in the SA stage is independent of the input x. The selection of the algorithm in the response selection stage is independent of which algorithm is selected in the situation assessment stage. Then, there are four possible pure strategies.

Let u represent the decision variable for choosing an algorithm in the SA stage and v be the variable for choosing an algorithm in the RS stage. Since each stage has two algorithms, u and v are binary, that is, both can take the values of either 1, which means the first algorithm is chosen, or 2, which indicates the second algorithm is selected. Using these values of u and v, the pure strategies can be represented as follows:

Pure strategy D1:	(u=1, v=1);
Pure strategy D2:	(u=1, v=2) ;
Pure strategy D3:	(u=2, v=1);
Pure strategy D4:	(u=2, v=2) .

However, during the real operation, it is not the case that only one strategy is always used. DMs use different strategies to perform the task according to a specific situation. Let p1, p2, p3, and p4 denote the probability that the pure strategies D1, D2, D3, and D4 are selected during an operating period. The strategy space is a four dimensional space. However, the probability law requires

$$\sum_{i=1}^{4} p_i = 1$$

This reduces the dimension of the strategy space by one, since

$$p4 = 1 - (p1 + p2 + p3)$$

and p4 is determined by p1, p2, and p3.

The corresponding strategy space is three dimensional and shown in Fig. 5.20. The strategy space is the *volume* bounded by the p1 = 0, p2 = 0, and p3 = 0 planes and the plane p1+p2+p3=1. The origin is also included in the strategy space because it is corresponding to the pure strategy p4 = 1 when the others are zero. In the next section, the mapping between the strategy space and the performance space will be discussed. The computation of the task workload associated with the strategies will be described.

5.7.2 Task Workload

The mapping between the strategy space and the performance space is as follows. Each point in the strategy space represents a specific way to do the task. If a task is done in this specific way, then there are corresponding values for the workload G and accuracy J associated with it. A pair of values for G and J is a point in the performance space. Therefore, for each given strategy, G and J can be computed. If G and J are computed for all points in the strategy space, the performance space is constructed.

The number of possible strategies is infinite since the values of the probability that a pure strategy is selected is a real number. Therefore, it is not feasible to compute G and J for all points of the strategy space. However, because of the convexity of the both spaces described in Chapter 3, only a small portion of the points in the strategy space need to be computed. The following steps describe the computation of G and J.

Workload and Accuracy for Pure Strategies

Recall from Chapter 3 that the general expression for the total activity is

$$G = \sum_i H(w_i) \tag{5.1}$$

where w_i is a system variable and, $H(w_i)$ is defined by

$$H(w_i) = -\sum_{w_i} p(w_i) \log(p(w_i)) \tag{5.2}$$

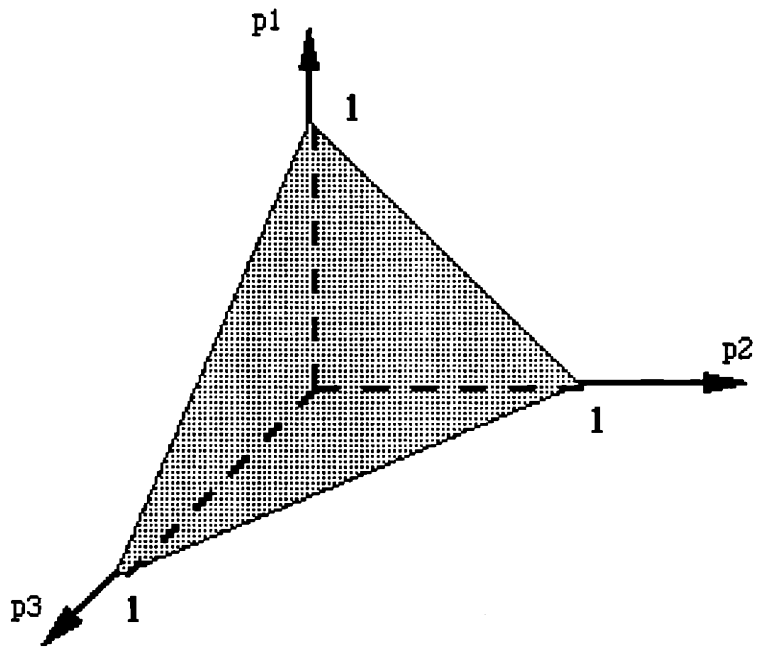


Figure 5.20 Strategy Space

The flowchart showing the sequence of operations for doing a task in a trial is shown in Fig. 5.21. Since there are N threats in each observation area, some of the system variables will occur several times. Let $\mu(N)$ denote the mean of N and M denote the number of threats in each sector. Since $\mu(N)$ and M are constant and what has been done to one of the threats in a trial does not affect what will be done to the next threat, the entropy of the system variables in the SA stage will be multiplied by $\mu(N)$ while the system variables in the IF or IC and RS stages will be multiplied by M. The $\mu(N)$ - M threats, which are in the overlap area but not in the sector, are not processed in the stages following SA and are sent to other DMs.

As shown in Figs. 5.18 and 5.19, there is one strategy switch for the supervisor and two for each DM in the parallel organization and each subordinate in the hierarchical organization. Since all strategy switches have two alternative algorithms, there are two pure strategies for the supervisor and four for the others. Detailed computation for pure strategies can be found in Appendix B.

Accuracy for a pure strategy is obtained by simulating the model in which all functions described in the previous section are implemented.

Workload and Accuracy for Behavioral Strategies

To compute accuracy and workload for a mixed strategy, it is necessary to introduce the organizational decision strategies (Levis and Boettcher, 1983). Recall from section 3.7, that the pure strategy for an individual DM is

$$D_k = \{ p(u = 1) = 1;\ p(v = j \mid z = z_k) = 1 \}$$

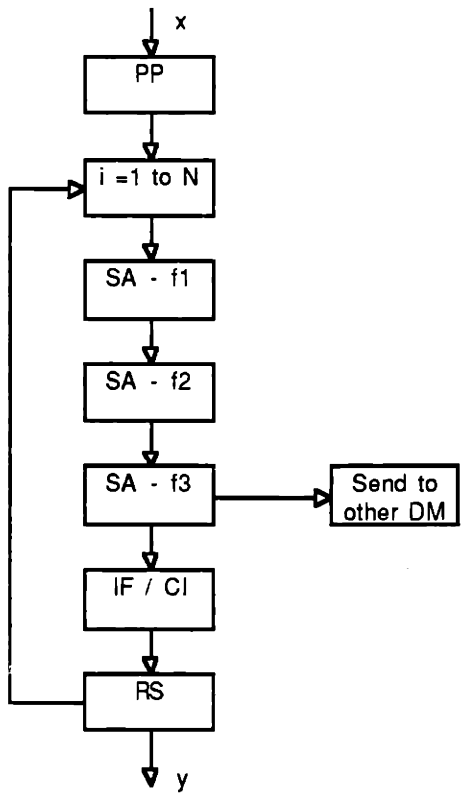


Figure 5.21 Block Diagram for Processing Sequence

A set of pure strategies , one for each DM, defines a pure strategy for organization. For a three-DM organization, a pure strategy is

$$\Delta_{k^1,\, k^2, k^2} = \{ D^1_{k^1},\ D^2_{k^2},\ D^3_{k^3} \} \tag{5.3}$$

where D^i is the pure strategy used by DM^i. There are

$$K = \prod_{i=1}^{3} k^i$$

pure strategies for the organizations.

When individual DMs do use mixed strategies, the organizational strategy is called behavioral strategy (Owen, 1968). Behavioral strategies for an organization can be expressed as follows (Levis and Boettcher, 1983):

$$\Delta = \{ D^1(p^1), D^2(p^2), \ldots \ldots, D^n(p^n) \}$$

$$= \sum_{k^1k^2k^3} \Delta_{k^1k^2k^3} \cdot p^1_{k^1} \cdot p^2_{k^2} \cdot p^3_{k^3} \tag{5.4}$$

where $D^i(p^i)$ is the strategy used by the ith DM.

The accuracy of an organization can be obtained as a function of Δ. Since any organizational strategy considered is a weighted sum of pure strategies equation (5.4), the organizational accuracy can be computed by

$$J(\Delta) = \sum_{k^1k^2k^3} J_{k^1k^2k^3} \cdot p^1_{k^1} \cdot p^2_{k^2} \cdot p^3_{k^3} \tag{5.5}$$

where $J_{k^1k^2k^3}$ is the accuracy when the DM_1, DM_2, and DM_3 use pure strategy k^1, k^2, and k^3, respectively.

The workload for each DM is also a function of Δ:

$$G = G(\Delta).$$

From the definition of G as the sum of the marginal entropies of each system variable and the fact that the probability distribution p(w) are elements of a convex distribution space determined by the organizational decision strategy, p(w) can be expressed as (Levis and Boettcher, 1983)

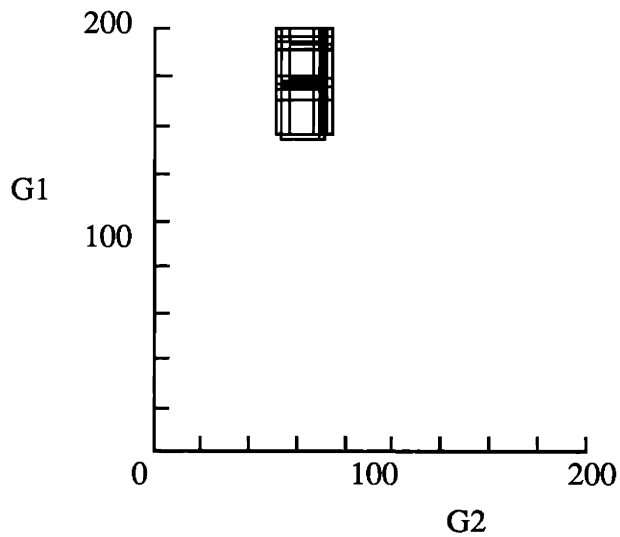
$$p(w) = \sum_{k^1k^2k^3} p(w \mid \Delta_{k^1k^2k^3}) \cdot p^1_{k^1} \cdot p^2_{k^2} \cdot p^3_{k^3} \tag{5.6}$$

By substituting equation (5.6) into equation (5.2), the marginal entropies for all system variables can be computed. Equation (5.1) then can be used to compute the workload G. Appendix B shows the details of this computation.

In the theoretical analysis, mixed strategies are implemented to compute J and G. Then, the performance-workload locus is constructed to predict the organizational performance.

In the experiment, mixed strategies used by DMs will be recorded. The workload associated with each strategy then can be computed and is expected to be inside the J-G locus computed analytically.

In Fig. 5.22, the projection of the Performance-Workload locus on the workload plane for decision makers DM1 and DM2 is shown. To each organizational strategy corresponds a point in the G1-G2 locus. In the next section, hypotheses will be generated by interpreting the results of theoretical analysis and aspects of the Performance-Workload locus.

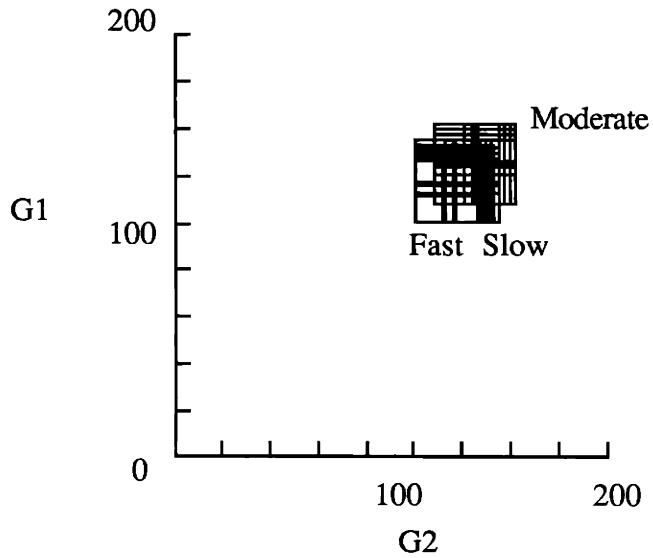


(a) Workload for Hierarchical Organization: Subordinate (G1); Supervisor (G2)

Figure 5.22 (a) Workload of Decision Makers

5.8 HYPOTHESES

From the G1-G2 locus shown in Fig. 5.22, the following properties are observed. The workload is different for DMs playing different roles. Subordinates in the hierarchical organization have the highest workload among all roles, while the supervisor has the least workload. The workload of the DMs in the parallel organization falls in between these values for the hierarchical organization. What is the effect of these properties on organizational performance?



(b) Workload for Parallel Organization: Two Human DMs

Figure 5.22 (b) Workload of Decision Makers

Let us consider what will happen when the available time decreases. When T_a decreases, the processing rate F increases while the task workload is kept constant (see Fig. 3.10). If T_a decreases continuously until the processing rate reaches the maximum value F_{max}, further decrease of T_a will force a reduction of workload which is accomplished by the DM selecting a strategy requiring less workload. This method of coping with time pressure works until the maximum rate F_{max} is attained using the strategy with the least required task workload, G_{min}. Then further decrease of T_a will result in a rapid degradation of performance since no strategy is available to do the task completely. The DM will fail to complete the task and may make random errors on the portion of the task that he completes.

Let T* denote the available time when G_{min} is chosen. Then, the maximum processing rate can be expressed as

$$F_{max} = \frac{G_{min}}{T^*} \tag{5.1}$$

For a DM in the hierarchical organization, the minimum G is denoted by G_{hmin}. For a DM in the parallel organization, the minimum workload is denoted by G_{pmin}. Since F_{max} has been assumed constant for an individual DM (Louvet et al., 1988), equation (5.1) results in

$$\frac{G_{hmin}}{T_h^*} = F_{hmax} \quad \text{and} \quad \frac{G_{pmin}}{T_p^*} = F_{pmax} \tag{5.2}$$

where T^*_h and T^*_p are the available times driving the DM to the maximum processing rate.

If the information theoretical model for workload were exact and captured all aspects of the cognitive tasks, then for a decision maker

$$F_{hmax} = F_{pmax}$$

In this case, the two parts of equation (8.2) can be combined to yield

$$\frac{G_{hmin}}{G_{pmin}} = \frac{T_h^*}{T_p^*} \tag{5.3}$$

Equation (5.3) indicates that if the values of the minimum workload for two structures are known, the ratio of available time at which performance degrades rapidly can be predicted exactly.

There is, however, another complication. While in the parallel organization the workload locus is symmetric - the two DMs shown have the same range for task workload (Fig. 5.22b) - this is not the case for the hierarchical organization (Fig. 5.22a). The question then arises as to which minimum workload should be considered, $G1_{min}$. or $G2_{min}$? It is argued now that in the hierarchical organization with a protocol requiring close interaction among DMs, when one DM's needed task processing rate exceeds his processing rate, the resulting individual degradation in performance will affect organizational performance. Consequently, the Gh_{min} in equation (5.3) is chosen as

$$G_{hmin} = \max_i \{ G^i_{hmin} \}$$

If the parallel organization's G1-G2 locus were asymmetric, then

$$G_{pmin} = \max_i \{ G^i_{pmin} \}$$

Finally, because the theoretical model for the cognitive workload is an approximate one, the exact relation represented by equation (8.3) may be expressed as an approximate relation:

$$\frac{G_{hmin}}{G_{pmin}} \sim \frac{T^*_h}{T^*_p} \tag{5.4}$$

From relation (5.4), it is observed that if G_{hmin} is larger than G_{pmin}, then, T^*_h will be larger than T^*_p, which indicates when the available time T_a decreases continuously, the rapid degradation of performance will occur in the hierarchical organization first. A hypothesis is established as follows.

Hypothesis 1. When the available time is decreasing, the organization with the highest minimum workload for a given set of strategies will exhibit a performance degradation at a larger value of available time than the organizations which have lower minimum workload.

The second hypothesis is derived by considering the possible strategies for doing the task. There are four pure strategies for each DM except for the supervisor in the hierarchical organization who has two pure strategies. According to equation (5.3), a pure strategy for an organization occurs when all DMs in the organization use a pure strategy. The number of pure strategies for the organizations is the number of combinations of all pure strategies used by DMs, which is computed by .

$$K = \prod_{i=1}^{3} k^i \tag{5.5}$$

where k^i is the number of pure strategies of the i-th DM.

Therefore, there are 64 pure strategies for the hierarchical organization and 32 pure strategies for the parallel organization. All the pure strategies and associated workload can be found in Appendix B. Table 5.6 shown the pure strategies which lead to the maximum or the minimum workload.

Table 5.6 (a) Maximum and Minimum Workload for Pure Organizational Strategies

Hierarchical

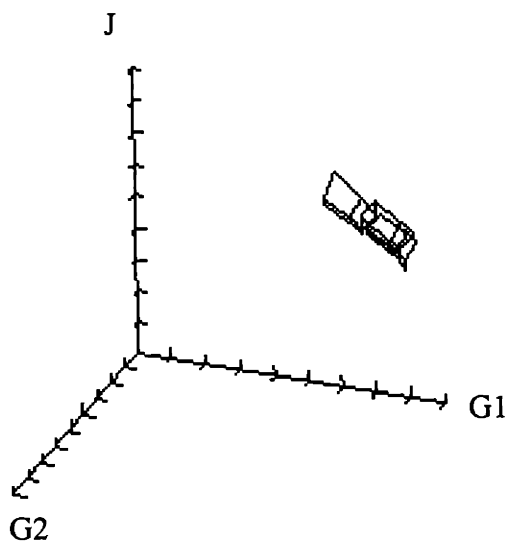
DM1	DM2	DM3	G1/G3	G2	J
D4	D4	D4	135.62	53.65	1.0
D1	D1	D1	182.52	72.06	0.82, 0.63, 0.47

Table 5.6 (b) Maximum and Minimum Workload for Pure Organizational Strategies

Parallel

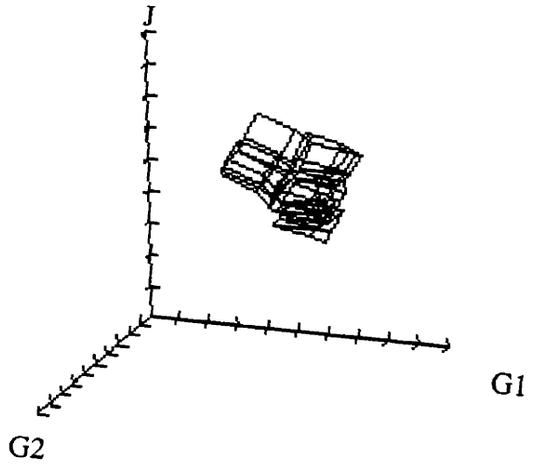
DM1	DM2	DM3	All G	J
D4	D4	D4	101.70	1.0
D1	D1	D1	135.12	0.76, 0.63, 0.58

In Table 5.6, strategy D1 corresponds to three values of the accuracy measure depending on the tempo of operations. The first value is the accuracy measure for slow operation; the second for moderate operation, and the third for fast operation. Figure 5.23 shows a set of Performance-Workload loci.

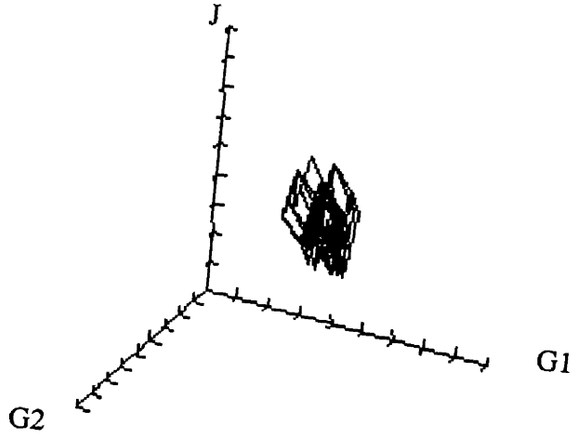


(a) Performance-Workload Locus for Hierarchical Organization: Moderate Speed

Figure 5.23 Performance-Workload Loci



(b) Performance-Workload Locus for Parallel Organization: Moderate Speed



(c) Performance-Workload Locus for Parallel Organization: Fast Speed

Figure 5.23 Performance-Workload Loci

From Table 5.6, it can be seen that the minimum workload in both hierarchical and parallel organizations is associated with D4 which is the strategy of probing. And as discussed in Section 5.5, the probing strategy results in the highest performance. To develop a hypothesis from this observation, let us consider the following.

Because of bounded rationality, DMs will change to strategies with less workload when the available time decreases. Given that in this experiment the minimum workload strategy yields the highest performance, there is no other strategy available for further reduction of the workload to accommodate a shorter available time when a DM reaches the maximum processing rate, F_{max}, when using the minimum workload strategy. Then, the ways to cope with the situation are either to reduce the number of communications or to reduce the number of threats being processed. Since the objective of the naval air battle is to process completely all threats, it is hypothesized that a decision maker will omit some required communication in favor of processing threats in his own sector. While this strategy may improve individual performance, it will cause a rapid degradation in organizational performance, Consequently, the onset of degradation of organizational performance should occur at the same time that the number of communications begins to be reduced.

This can be interpreted as selfish, local behavior. Each DM, under pressure, will attempt to respond to the threats in his sector at the expense of organizational performance. Essentially, this means that under pressure, individual DMs will tend to decouple from the organization by reducing coordination and operating in a decoupled mode. If this were not the case, then degradation of performance will begin before reduction in communications and the latter will be more gradual than performance degradation. For this argument, the following hypothesis is formulated.

Hypothesis 2 Since the minimum workload strategy yields highest performance, under increased time pressure decision makers will reduce communications (coordination) with an attendant reduction in organizational performance.

These two hypotheses will be tested by the experiment.

5.9 SUMMARY

In this chapter, the purpose of the experiment has been stated. In order to design a model-driven experiment, the naval outer air battle was chosen as the task to be performed. Three-decision-maker organizations, parallel and hierarchical, were selected to conduct the investigation on the effect of organizational structure on performance. The organizations were modeled using the Petri Net representation to show the protocol for interactions among the decision makers. An analysis has been carried out to predict organizational performance and

generate hypotheses. Two hypotheses were established that will be tested in an experiment. In the next chapter, the second stage - experimental investigation - will be described.

Chapter 6

EXPERIMENT DESIGN

In this chapter, the second stage of the methodology for designing model–driven experiments is implemented. The first section of this chapter states the purpose of the experimental investigation and describes the type of experiment to be designed. Then a model that will be used to design a multi-person experiment for testing the hypotheses established in Chapter 5 will be developed. Dimensional analysis will be used as the tool in developing the model. The parameters that are critical to the evaluation of measures of performance (MOPs) in this study will be discussed. The parameters to be controlled and those to be measured will be determined. The experiment design will be described.

6.1 INTRODUCTION TO THE EXPERIMENTAL INVESTIGATION

6.1.1 Purpose of Experiment and Approach

In the previous chapters, the methodology for analysis and evaluation of performance of decision making organization was described. However, it is not yet known how well the model predicts performance of an actual organization or whether theoretically predicted phenomena or behaviors can be observed during an actual operation of an organization. An experimental investigation, in which an information processing and decision making environment is simulated, will be helpful in clarifying these issues. The purpose of the experiment is to assess organizational performance experimentally so that the results can be compared with the analytically derived results. This comparison will provide information about the correctness of the current model and the accuracy of the methodology for the evaluation of organizational performance. It is necessary to emphasize that this study is not designed to answer questions such as if one organizational structure is better than another. Instead, the objective is to observe the behavior of different organizational structures in a controlled environment and use the information for model validation. If the model is validated - i.e., its predictions are experimentally verified - then it could be used as a design and evaluation tool and for comparing the performance of organizational structures.

6.1.2 A Multi-person Model-driven Experiment

The specific objective of the experimental investigation is to consider a multi-person model-driven experiment. Decision making organizations consist of more than one decision maker (DM). A group of DMs work together in coordination to perform information processing and decision making tasks. Besides the human decision makers, the organization includes a large number of other components , i.e., computers, networks, and various kinds of communication and data storage equipment.

One of the major difficulties in developing a model-driven experimental program is the large number of parameters that have to be specified and varied. This results in a two-part problem: (a) The parameterization of the experimental conditions leads to a very large number of trials, a situation that is not really feasible when human subjects are to be used, and (b) Not all experimental variables can be set at the values required by the experimental design because of the lack of direct control on the cognitive variables.

Consequently, an orderly procedure is needed that will allow the reduction of the number of experimental variables and, more importantly, that will lead to variables that are easier to manipulate. Such an approach, called dimensional analysis, has been used in the physical and engineering sciences (Hunsacker, 1947; Gerhart, 1985). In Chapter 5, dimensional analysis is extended to the problems that have cognitive aspects so that it can be used for the design and analysis of experiments on decision making organizations.

6.2 DIMENSIONAL ANALYSIS

As stated in the previous chapter, the complexity of the distributed decision making organization results in a large number of parameters affecting the performance. However, it is not practical to run the experiment under every condition created by varying each parameter. To reduce the number of experimental conditions and to assure that the experimental results will provide enough information, the parameters which will affect MOPs should be examined carefully. Dimensional analysis, a scientific and engineering method, will be extended to include cognitive aspects, and will be used to check the correctness and completeness of the model. The model of the experiment describes the functional relation between the parameters and the MOPs.

6.2.1 Introduction to Dimensional Analysis

Dimensional analysis is a method for reducing the number and complexity of experimental variables which affect a given physical phenomenon. A detailed introduction to dimensional analysis can be found in (Hunsacker, 1947) and (Gerhart, 1985).

Dimensions and Units. A dimension is the measure which expresses a physical variable qualitatively. A unit is a particular way to express a physical quantity, that is, to relate a value to a dimension. *Fundamental dimensions* are the primary dimensions which characterize all variables in a physical system. For example, length, mass, and time are fundamental dimensions in mechanical systems. A dimension such as length per time is a secondary or derived dimension. If the dimension of a physical variable cannot be expressed by the dimensions of others in the same equation, then this variable is dimensionally independent.

The foundation of dimensional analysis is the *Principle of Dimensional Homogeneity*, which states that if an equation truly describes a physical phenomenon, it must be dimensionally homogeneous, i.e., each of its additive terms should have the same dimensions. The basic theorem of dimensional analysis is the π theorem, also called Buckingham's theorem:

> **π theorem:** If a physical process is described by a dimensionally homogeneous relation involving n dimensional variables, such as
>
> $$x_1 = f(x_2, x_3,..., x_n) \tag{6.1}$$
>
> then there exists an equivalent relation involving (n-k) dimensionless variables, such as
>
> $$\pi_1 = F(\pi_2, \pi_3, ..., \pi_{n-k}) \tag{6.2}$$
>
> where k is usually equal to, but never greater than, the number of fundamental dimensions needed to describe all x's.

Each of the π's in equation (6.2) is formed by combining (k+1) x's to form dimensionless variables. Comparing equations (6.1) and (6.2), it is clear that the number of independent variables is reduced by k, where k is the maximum number of dimensionally independent variables in the relation. The proof of the π theorem can be found in (Gerhart, 1985).

The π theorem provides a more efficient way to organize and manage the variables in a specific problem and guarantees a reduction of the number of independent variables in a relation. Dimensionless variables, also called dimensionless groups, are formed by grouping primary variables with each one of the secondary variables.

6.2.2 Construct the Experiment Model Using Dimensional Analysis

To apply dimensional analysis to decision making organizations, the fundamental dimensions of the variables that describe organizational behavior must be determined. A system of three dimensions is shown in Table 6.1 that is considered adequate for the purpose in modeling cognitive workload and bounded rationality.

Table 6.1. Fundamental Dimensions for Systems with Cognitive Aspects

DIMENSION	SYMBOL	UNITS
Time	T	second
Information	I	bit
Task	S	symbol

These fundamental dimensions are related to the measures of performance as follows. Consider the accuracy and response time of an organization. Let J denote accuracy and T_f denote response time. The description of the task in Chapter 5 shows that the parameters that may affect this MOP are:

1) available time to do a task, T_a;
2) number of threats in a task, N;
3) total number of enemy aircraft in a threat, m.
4) uncertainty of input, H;
5) task workload of individual decision makers, G; and
6) number of communications required by the task, N_{rc}.

The significance and effect of these variables are explained in the following paragraphs.

T_a: Available Time to Do a Task

In the experiment, a task is performed in a trial in which N threats are involved. T_a is a critical parameter which determines the tempo of operations. There are three regions from which T_a can take its value. In each of these regions, the performance of the organization and the behavior of the decision makers have somewhat different characteristics.

The first region is the region in which T_a is "small," that is, decision makers are under severe time pressure in performing the task. Therefore, the required processing rate is high and may exceed the maximum rate F_{max} possible for each DM. The given task may be not completed and accuracy may be poor. The second region can be called the high activity region. In this region, a small decrease of T_a forces decision makers to increase their processing rate, but not beyond their capabilities. In the third region, T_a is relatively large so that there is no time pressure. Decision makers can take time to perform the task, and the performance of the organization is virtually independent of time.

The focus of this study is on performance in the middle region of T_a because this is where the onset of rapid degradation of performance occurs. Since the objective is to design organizations that can execute given tasks within the available time and at a desired performance level, it is necessary to be able to predict the onset of performance degradation as a function of available time so that it can be avoided.

N: Number of Threats in a Trial

The number of threats in a trial determines the amount of work that needs to be done. It is a positive integer.

m : Number of Aircraft in a Threat

The number of enemy aircraft in a threat determines how many resources are needed. The decision on resource allocation depends on this variable.

H: Uncertainty of Input

The entropy of the input x, H(x), is used to characterize its uncertainty. The input x contains the attributes of the threats: number of threats, N, and number of aircraft in a threat, m.

N and m have probability distributions, denoted by p(N) and p(m) respectively. When p(N) and p(m) are given, the input entropy H(x) can be computed using

$$H(x) = -\sum_{x} p(x) \log_2 p(x)$$

where p(x) is the probability of input which can be expressed by the joint probability distribution of N and m. Since N and m are independent of each other, the joint probability can be written as p(N)p(m). Then, H(x) is obtained from

$$H(x) = -\sum_{N,m} p(N)\, p(m) \log_2 [\, p(N)\, p(m)\,] = H(m) + H(N)$$

The values of N and m will be discussed later in this chapter.

H(x) will vary with different probability distributions of N and m. The higher the entropy, the more uncertainty in the task. Different experimental conditions can be specified by varying the probability distributions of the input attributes.

G^i: Task Workload of the i-th Decision Maker

There are several ways to execute a task. The different ways can be characterized by the strategies used, since each strategy corresponds to a different algorithm used to perform the task. Each algorithm requires a certain amount of cognitive activity. Task workload is then defined as the amount of cognitive activity required when a particular algorithm is used to do the task. If the choice of algorithm is known, then the task workload can be used to estimate the amount of cognitive activity involved. (The term "workload" will mean "task workload" unless otherwise specified.) Let D^i denote the strategy used by the i-th DM. The actual task workload can be expressed by

$$G^i = G(D^i)$$

where function G relates a particular strategy to the cognitive workload.

N_{rc}: Number of Required Communications

This variable characterizes the level of interaction between the organization members. The assumption is that the team members are trained to work together toward a common goal, i.e., they are trying their best to perform the task. Therefore, the number of actual communications

is proportional to the required number of communications. The more communications, the higher the level of interaction. This simplified model of interactions is only intended to provide insight in what we believe to be one of the most important variables; it does not reflect fully the complexity of the actual operation.

The value of N_{rc} depends on the protocols and procedures established in the analysis stage. Specifically, since communications are required only when threats are in the overlap areas between sectors, N_{rc} is determined by the bearing of the threats.

In summary, independent variables are:

$$T_a,\ N,\ m,\ H,\ G^i, \text{ and } N_{rc}$$

where G^i is the task workload of the i-th DM. i = 1, 2, 3 for a three-DM organization.

Dependent variables are the two MOPs: accuracy and response time. However, neither of these MOPs are direct measurements. They are computed from other variables which can be directly measured from the experiment. Accuracy can be represented by the correctness in determining the type of the threat and in allocating resources. The response time is related to the individual DM's processing time. Then the MOPs can be computed for the naval outer air battle. Three relations are established as follows.

$$Tp^i = f1(T_a, H, N, G^i, N_{rc}) \tag{6.3}$$

$$N_c = f2(N_{rc}, T_a, Tp^1, Tp^2, Tp^3) \tag{6.4}$$

$$Y^i = f3(T_a, H, N, G^i, M) \tag{6.5}$$

where Tp^i, N_c, Y^i, and M are defined as follows:

Tp^i : Processing Time of the i-th Decision Maker in a Trial

The processing time is the time period during which a DM is actually carrying out the task. The processing time is usually less than the available time except when the available time is so short that the task cannot be completed, in which case Tp^i is the same as T_a. Tp^i depends on the algorithm chosen, the available time, and the individual skill of a DM. This variable can be measured directly during the experiment.

N_C: The Actual Number of Communications in a Trial

Each task requires communication. However, the actual number of communications depends on several other factors including time available to communicate, the strategy adopted by the DMs, and so on. This variable is directly measured in the experiment and is used to indicate the actual level of interaction between the team members.

Y^i: The Resource Allocation of the i-th DM in a Trial

The dimension of Y^i depends on the task that an organization performs. For the naval outer air battle, Y^i is the total number of resources allocated to threats. This variable is used to compute the accuracy of the organizational response. Because the number of resources has the dimension of "symbol", the dimension of Y^i is S. In general, the output of the organization can have different dimensions depending on a particular application.

M: Total Number of Enemy Aircraft in N Threats for a Trial

Because Y^i is computed for a trial and not for each individual threat, it follows that the number of enemy aircraft in a trial should be used in equation (6.5). M is defined as

$$M = \sum_{k=1}^{N} m_k$$

where m_k is the number of aircraft in the k-th threat.

The question of correctness and completeness of equations (6.3) to (6.5) will be answered using dimensional analysis. The following paragraphs show the application of dimensional analysis step by step.

Step 1 *Write a dimensional expression*

Equations (6.3) to (6.5) are the dimensional expressions for the system of interest.

The first step in the application of dimensional analysis is to check whether this functional relation could describe the relation between the MOPs and other variables. The dimensions of the variables in equations (6.3) to (6.5) are as follows:

$[T_a] = T$	$[H] = I$	$[N] = S$	$[N_{rc}] = S$	$[m] = S$
$[G^i] = I$	$[Tp^i] = T$	$[Y^i] = S$	$[N_c] = S$	

To check if equations (6.3) to (6.5) are homogeneous, consider the dimensions on both sides of these equations. Since the dimension of Tp^i is time, the right hand side of equation (6.3) must have the dimension of time. Similarly, the dimension of N_C in equation (6.4) requires that f2 result in a combination of variables having the dimension of symbol. Finally, equation (6.5) requires both sides of the equation to have the dimension of symbol. Because all fundamental dimensions for the decision making system are present in the right hand side of equations (6.3) to (6.5), it is possible to combine them to obtain the dimensions required by the left hand side.

There are six dimensional variables in each of these equations, that is, n = 6.

Step 2 *Determine the number of dimensionless groups*

The number of dimensionless variables is equal to n-k, where k is the maximum number of dimensionally independent variables in equations (6.3) to (6.5). The maximum number of dimensionally independent variables is three. Therefore k is equal to three. Then, the number of dimensionless groups is:

$$n - k = 6 - 3 = 3.$$

There will be three dimensionless groups in the dimensionless equation corresponding to equations (6.3) to (6.5).

Step 3 *Construct the dimensionless groups*

While the choice of primary variables is essentially arbitrary, consideration should be given to making the dimensionless groups meaningful. If T_a, N, and H are selected as the three (because k = 3) primary variables, three dimensionless groups are constructed on the basis of the remaining variables Tp^i, G^i, and N_{rc} in equation (6.3). As an example, a dimensionless group π_1 is formed by combining T_a, N, H and Tp^i. Using the power-product method, π_1 can be determined by the following procedure. Write π_1 as

$$\pi_1 = T_a{}^a \; N^b \; H^c \; Tp^{i\,d}$$

where a, b, c, and d are constants that make the right hand side of the equation dimensionless, so that the equation is dimensionally homogeneous. In terms of the dimensions of π_1, T_a, N, H and Tp^i we have

$$[\pi_1] = [S^0 \, I^0 \, T^0] = [T]^a \, [S]^b \, [I]^c \, [T]^d = T^{a+d} \, S^b \, I^c$$

By the principle of dimensional homogeneity, the following set of simultaneous algebraic equations must be satisfied.

For T:	$a + d = 0$
For S:	$b = 0$
For I:	$c = 0$

There are three equations and four unknowns, so the solution is not unique. In general, the choice of solution depends on the particular problem. For our purposes, the secondary variable, in this example Tp^i, is chosen to appear in the first power, that is, d is set equal to unity. Thus by solving the set of algebraic equations, we obtain:

$$a = -1, \quad b = 0, \quad c = 0, \quad d = 1.$$

Then substituting a, b, c, and d into the expression for π_1 gives

$$\pi_1 = Tp^i/T_a$$

Similarly, using the same power-product method, the other dimensionless variables are:

For equation (6.3):

$$\pi_2 = G^i / H$$
$$\pi_3 = N_{rc} / N$$

For equation (6.4):

$$\pi_1 = N_c / N_{rc}$$
$$\pi_2 = T_p{}^1 / T_a$$
$$\pi_3 = T_p{}^2 / T_a$$
$$\pi_4 = T_p{}^3 / T_a$$

For equation (6.5):

$$\pi_1 = Y^i/m$$
$$\pi_2 = G^i\, T_a / HTp^i$$
$$\pi_3 = m/N$$

Then, the dimensionless form of equations (6.3) to (6.5) is

$$\frac{T_p^i}{T_a} = \Phi_2\left(\frac{G^i}{H}, \frac{N_c^i}{N^i}\right) \tag{6.6}$$

$$\frac{N_c}{N_{rc}} = \Phi_1\left(\frac{T_p^1}{T_a}, \frac{T_p^2}{T_a}, \frac{T_p^3}{T_a}\right) \tag{6.7}$$

$$\frac{Y^i}{m} = \Phi_3\left(\frac{G^i T_a}{H T_p^i}, \frac{m}{N^i}\right) \tag{6.8}$$

Equations (6.6) to (6.8) show all variables which are directly involved in the experiment. This is the model that will be used to design the experiment.

Figure 6.1 shows the structure and relationships for deriving these dimensionless equations. All the variables involved in the experiment are partitioned into two sets: controlled variables and measured variables. The combination of the variables results in three dimensionless equations which will be used in the experimental design.

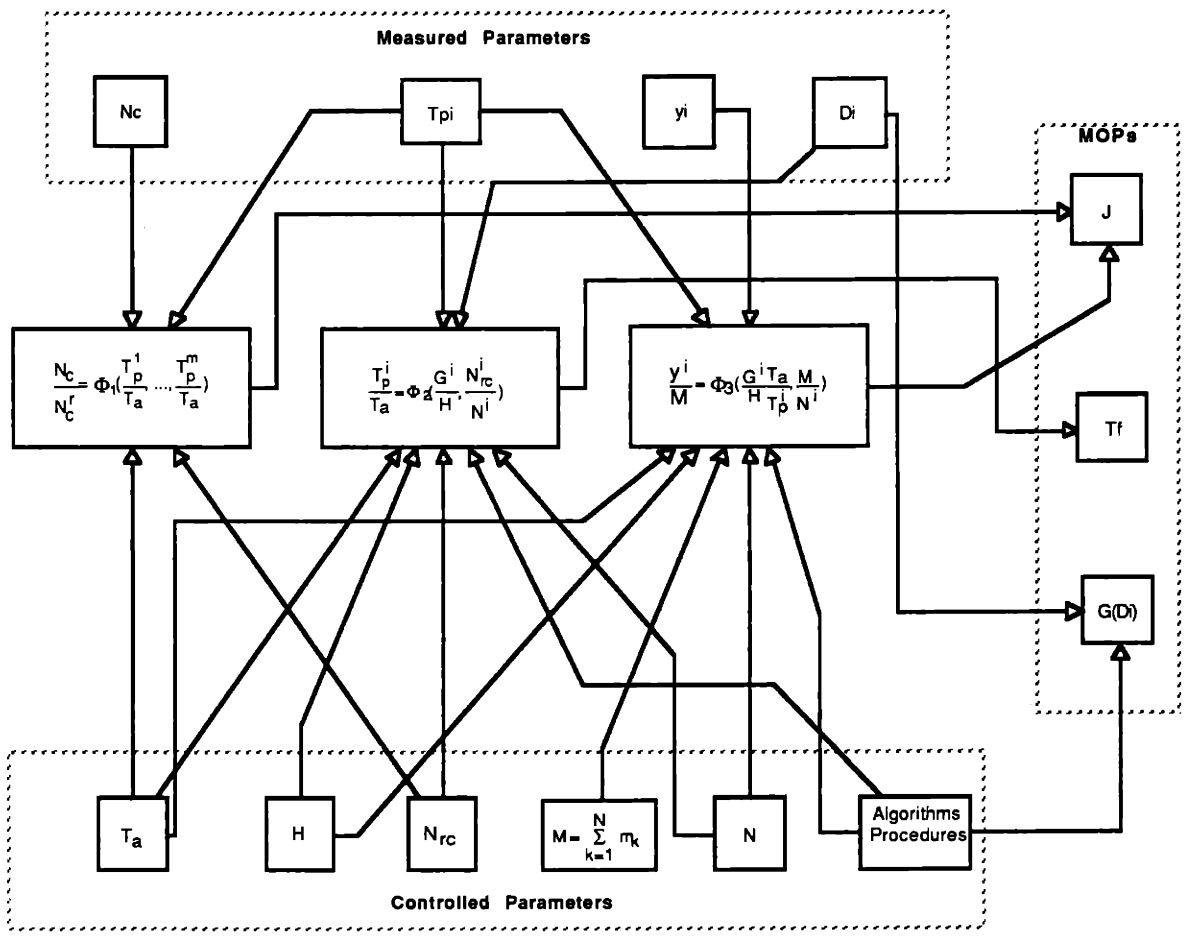


Figure 6.1 Framework for Obtaining MOPs

The values of the parameters on the right hand side of equations (6.6) to (6.8) can be controlled or measured directly or indirectly in the experiment. The actual task workload G^i can be computed when the strategy used in doing the task is known. Since the algorithms for performing a task depend on the organizational protocols and procedures, partial and indirect control of the task workload is possible by varying organizational protocols and the procedures for doing the task. Input uncertainty H can be varied by changing the probability distribution of the input attributes. The processing time T^ip can be bounded from above by the available time T_a. T^ip can be measured directly. Strategy D^i is explicitly shown in Fig. 6.1 because it will be directly measured in the experiment and will be used to compute G^i. Different values of the controlled parameters will create different experimental conditions for testing the hypotheses. In a later section of this chapter, the ranges of the values for these parameters will be derived.

Comparing equations (6.3) to (6.5) and equations (6.6) to (6.8), one finds that the number of independent variables is reduced from seven to four in equations (6.4) and (6.7); and from six to three in the other pairs of equations. This reduces the complexity of the equations and facilitates experiment design and analysis. Properly designed experiments using dimensional analysis provide similitude of experimental conditions for different combinations of dimensional variables with the same value of π's. Similitude reduces the number of trials that must be run in order to define the three Φ's. This is a major advantage especially when the physical (dimensional) experimental variables cannot be set at arbitrary values.

6.3 COMPUTATION OF PERFORMANCE

Given the experiment model represented by Equations (6.6) to (6.8), how should performance be evaluated? This section describes the parameters being measured directly from the experiment and their relation to MOPs. The following computations are just one way to obtain measures of performance and are appropriate for the naval outer air battle task being simulated in this experiment. There are many other formulas that can be used to compute MOPs which may be more appropriate for other applications.

Accuracy

Accuracy is represented by the accuracy index J. J is the measure of the discrepancy between the desired output, Y_d, and the actual output, Y, for a given input x. Y_d is computed according to the input while Y is observed from the experiment. Figure 6.2 shows the block diagram for determining J. x is the input to the experiment, and the mapping function L maps

the input to the desired output. $C(Y_d,Y)$ is the cost function for the deviation of actual output from the desired output. The detailed computation of accuracy is shown in Appendix D.

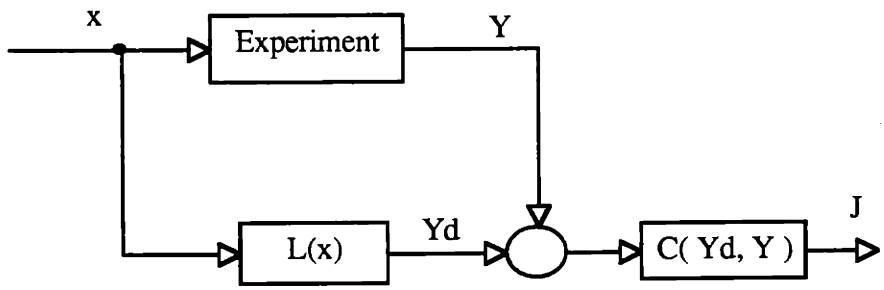


Figure 6.2 Block Diagram for Evaluating Accuracy

Individual processing time, Tp^i

Tp^i is all the time used by DMi to process the threats in the i-th observation area during a single trial. Recall that the observation area consists of a sector plus overlap areas as described in Chapter 5. The time that the first threat is detected is recorded as t_0. The time that the last threat has been attacked is recorded as t_f. The difference between t_f and t_0 is the processing time, that is,

$$Tp^i = t_f - t_0$$

Response Time

The response time, T_f, of the organization is not directly measured in the experiment. It is computed from the processing times of individual DMs. T_f measures the time period between the starting time and the ending time of each engagement of the organization. The starting time of an engagement is defined as the time at which the first threat is detected. The ending time is defined as the time when there is no active threat (either all threats are attacked or some are attacked and others penetrate into the center) in the outer air battle region. The value of T_f is the maximum processing time of individual DMs in the organization, that is,

$$T_f = \max_i (T_p^i)$$

where i = 1, 2, 3 for the parallel organization;
i = 1, 3 for the hierarchical organization.

Tp^i is the processing time of DMi and is measured during the experiment.

Workload

The frequency with which each algorithm is used to perform the task is recorded. The probability that each of the algorithms is selected is obtained. The decision strategy is determined from these probabilities. Then the cognitive workload for carrying out the task during the experiment is computed. The algorithms have been discussed in Chapter 5. The detailed computation of the workload is shown in Appendix B.

Number of Communications

In addition to the MOPs defined in Chapter 3, the number of communications, N_C, is measured directly in the experiment. N_C reflects the interaction between the DMs in an organization. For a given task, the amount of communication can be specified by the protocols and the procedures. However, the actual communications depend on the strategy chosen to perform the task and the attributes of the input. Therefore, the actual number of communications needs to be recorded to obtain information on the interaction between DMs.

When all of the variables discussed above are obtained, the experimental values of MOPs are available for testing the hypotheses and comparing with the theoretical predictions of organizational performance.

6.4 CONTROLLED PARAMETERS

Controlled parameters can be divided into two groups: fixed and varied. Fixed parameters are those whose values do not change during the experiment. The fixed parameters include the procedures, protocols, probability distributions of input attributes, and constants in the implementation of the experiment. Varied parameters are the speed of threats, which specifies the available time, the number of threats in a trial, and the number of aircraft in a threat. Varied parameters can be manipulated to create different experimental conditions to test the hypotheses.

There are three controlled parameters in the experiment. The time available to do the task, T_a, is the parameter that controls the tempo of operation. Different values of this parameter are obtained by setting the speed of the threats: the faster the threats, the less the time available to perform the task. When T_a becomes shorter and shorter, the time pressure in doing the task is higher and higher.

The second controlled parameter is the number of threats, N. N is varied stochastically. The number of aircraft, m, in a threat is the third controlled parameter. This parameter is related to the type of aircraft in a threat. The dimensions of N and m are symbol.

Since the values of the controlled parameters specify the experimental conditions, determination of the ranges of the controlled parameters is critical to the success of the experiment. In this section, the parameter ranges of available time, number of threats, and number of aircraft in a threat will be derived.

Available Time T_a of a Trial

The available time, T_a, depends on the speed and the initial range of threats. The initial range is the farthest distance at which a threat can be detected. This range is defined as the range of the passive radar, which has been described in Chapter 5 and is denoted by R_p. Because the initial range for all threats is the same, the available time is determined only by the speed. Let V denote speed; the formula used to compute the available time is

$$T_a = \frac{R_p}{V} \tag{6.11}$$

The maximum range of the E2C radar is about 300 miles (in the implementation, R_p = 345 miles is used). There are two critical values for T_a: the maximum value and the minimum value. The maximum value specifies the longest interval during which the task can be processed while the minimum time is the shortest time for the task to be completed.

The speed of threats ranges from 300 miles per hour (mph) to 1200 miles per hour. To find the maximum value, the minimum speed is used in Equation (6.11). The maximum available time, T_{max}, is

$$T_{max} = \frac{R_p}{V_{min}}$$

Since

$$R_p = 345 \text{ miles}, \qquad V_{min} = 300 \text{ mph},$$

then $$T_{max} = \frac{R_p}{V_{min}} = \frac{345}{300} = 1.15 \text{ hours}$$

Similarly, using the fastest speed, V_{max}, in equation (6.11), the minimum available time, T_{min}, is

$$T_{min} = \frac{R_p}{V_{max}} = \frac{345}{1200} = 0.288 \text{ hours}$$

In the experiment, it is impractical to use the actual time elapsed. A scale factor is introduced to speed up the experiment. A constant scale factor, ß, is computed as follows.

Assume the desired maximum experiment time is $T_{exp.max}$, then

$$\beta = \frac{T_{max}}{T_{exp.max}} \tag{6.12}$$

Finally, T_a for the experiment can be computed by

$$T_a = \frac{R_p}{\beta V} \tag{6.13}$$

for each value of speed.

The determination of $T_{exp.max}$ is the first step in finding the range of T_a. $T_{exp.max}$ is the longest time available to do the task. Considering the feasibility of the experiment which involves human subjects, the time of the experiment should be appropriate. If $T_{exp.max}$ is chosen to be 120 seconds, then, Equation (6.13) gives

$$\beta = \frac{1.15 \times 3600}{120} = 34.5$$

Using this ß in Equation (6.12), $T_{exp.min}$ is computed to be

$$T_{exp.min} = 30 \text{ seconds.}$$

Six T_a values were selected for the experiment:

$$T_a = \{ 30, 40, 60, 80, 100, 120 \} \text{ in seconds.}$$

The appropriateness of these values is determined as follows. The minimum value of T_a determines the fastest tempo of operation. The bounded rationality constraint requires that the information processing rate for the task should not exceed the maximum value (F_{max}). Therefore, the $T_{exp.min}$ should drive the information processing rate to the bounded rationality boundary but not exceed it. On the other hand, the longest available time results in the slowest operation. However, it should not be so long that the DMs become inattentive. A pilot experiment is needed to verify these range.

In Chapter 7, the results from the pilot experiment will be discussed, and the values of $T_{exp.max}$ and $T_{exp.min}$ will be examined.

Number of Threats, N in a Trial

The number of threats in a trial partially specifies the amount of work needed to be done during the experiment. In the experiment, the total number of threats in the entire *defense area* and in *each sector* are constant. However, the number of threats that a DM has to process varies. In other words, the number of threats in an observation area is not a constant. Recall that the observation area is the assigned sector plus the overlap areas. The total number of threats to be processed by a DM is equal to the number of threats in the observation area.

Since the purpose of the experiment is to investigate *organizational* performance, coordination between DMs is a necessary feature. Therefore, the experiment is designed to require coordination by including threats in the overlap areas.

Whether a threat is in the overlap area depends on its initial bearing on the radar screen. Therefore, the number of threats in each observation area is determined by the position of the threats. In the experiment, the possible number of threats in each observation area are:

for the parallel structure: $N = \{ 4, 5, 6\}$;
for the hierarchical structure: $N = \{ 5, 6, 7\}$.
The detailed description in obtaining these values of N is in Appendix A.

Number of Aircraft in a Threat, m

The number of aircraft in a threat is related to the type of threats. As stated in Chapter 5, there are three types of threats: bombers, fighters, and surveillance aircraft. In order to determine the type of a threat, two pieces of information are required: speed of the threat and the number of aircraft in the threat. The variability of number of aircraft results in uncertainty in determining the type during the experiment. The more uncertainty, the higher the workload.

Table 6.2 repeats Table 4.1 to show the relation between the speed, number of aircraft, and type of aircraft. From Table 6.2, it can be found that when the number of aircraft is five (m = 5) either two or all three of the types are possible. This implies higher uncertainty in determining the type. By varying the probability distribution of m, the uncertainty level during the experiment can be changed. Consequently, the input entropy, H(x), and the workload required for the task are also changed.

The values of m are

$$m = \{ 2, 3, 4, 5, 6 \}.$$

Table 6.2 Type of Threats

Number \ class	= Fast	= Medium	= Slow
m < 5	F	F,S	S
m = 5	F, B	F, B, S	B, S
m > 5	B	B	B

Number of Required Communications in a Trial, N_{rc}

This is determined by the input and the protocol for the organization. This variable is related to the number of threats in an observation area N. For the hierarchical and parallel organizations, N_{rc} is different. For detailed information about the range and probability of N_{rc}, see Appendix A.

Entropy of the Input, H

Input entropy depends on the probability distribution of the input attributes: T_a, N, and m. H is calculated using the task model introduced in Chapter 3. In Appendix A, the probability distributions are described and the computation of H is shown.

All the controlled and measured experimental parameters are shown in Table 6.3

6.5 THE EXPERIMENT DESIGN

In the experiment, the naval outer air battle discussed in Chapter 5 is simulated. To test the hypotheses, the experiment is designed to be consistent with the theoretical model. The two

organizational structures described in Chapter 5 are used: the parallel structure and the hierarchical structure. In the parallel structure, all DMs are at the same level of authority. They are working together in coordination In the hierarchical structure, authority varies with the ranks of DMs, that is, the positions in which DMs are in the organization. In both structures, the task is the same; and all members of the organization have to act as a team to perform the task.

Table 6.3 Summary of Controlled and Measured Parameters

Controlled Parameters	Measured Parameters
T_a: Available time	J: Accuracy index (Y, N, m)
N: Number of threats	T_f: Response time
m: Number of aircraft	G^1: Workload of DM1
H: Input uncertainty	G^2: Workload of DM2
N_{rc}: Number of required communication	G^3: Workload of DM3
Algorithms and procedures	T_p^1: Processing time of DM1
	T_p^2: Processing time of DM2
	T_p^3: Processing time of DM3
	N_c: Number of communications

Each DM is provided a computer screen which displays the task situation. DMs process the data available to them, coordinate with each other, and make decisions on what to do about the situation. The objective is to respond accurately to the input. All DMs understand the goal clearly: to assign interceptors to every threat.

As described in section 6.2, MOPs are computed from observed variables in the experiment. The input is generated by a computer. The input attributes are designed and controlled to simulate different experimental conditions. The main attribute is time pressure which can be varied by controlling the speed of threats. The major interest is to find whether the phenomena predicted by the model can be observed in the experiment.

Figure 6.3 shows the block diagram for the experiment design. In Fig. 6.3, X is the input vector containing all controlled parameters that can be varied to create different experimental conditions. B is the vector of the controlled parameters which take constant values. Y is the vector which contains all measured variables in the experiment. The function f(Y; X, B) is the mapping between the measured variables and MOPs. After obtaining the MOPs from the experimental data, the hypotheses can be tested.

The elements of X, B, and Y are:

$$X = \{ V, N, m \}$$
$$B = \{ R_p, p(N), p(m), p(\theta), p(type \mid m, V) \}$$
$$Y = \{ t_0, t_f, r_1, r_2, r_3, u, v, type, N_c \}$$

where

in X, the variables refer to a trial:

V:	speed of threats
N:	number of threats;
m:	a vector of dimension N in which each element is the number of aircraft in a threat;

in B, the parameters are constant in all trials and for both organizational structures:

R_p:	the farthest distance that a threat can be detected;
p(N):	probability that the number of threats is N;
p(m):	probability that the number of aircraft in a threat is m;
$p(\theta)$:	probability that entering bearing of a threat is θ;
p(type| m,V):	probability that a threat is a particular type for given speed and number of aircraft;

in Y, all the following measures are for a threat except N_c which is measured for each trial:

t_0:	time that a threat is being processed;
t_f:	time the process for a threat is completed (when interceptors are assigned);
r_i:	number of the type i resource allocated to a threat;
u:	algorithm selected in the situation assessment stage;
v:	algorithm selected in the response selection stage;
type:	type assigned to a threat;
N_c:	number of communications.

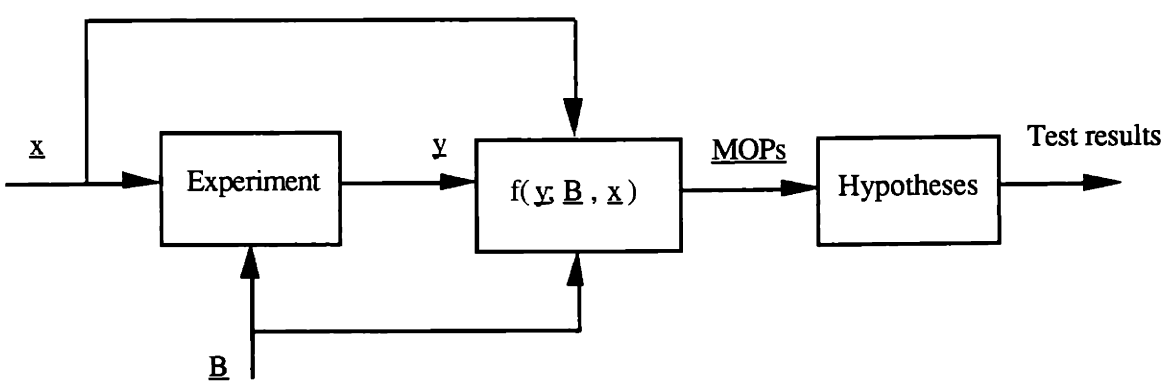


Figure 6.3 Block Diagram of Experiment

When MOPs are computed from the experimental data, the hypotheses can be tested. Appendix C shows the computer displays for all DMs in the hierarchical and in the parallel organization.

6.6 EXPERIMENT SETUP

6.6.1 Hardware and Software

The experiment is implemented on Macintosh computers. Three Macintosh computers are connected to form a network (Fig. 6.4). Each computer is a node in the network.

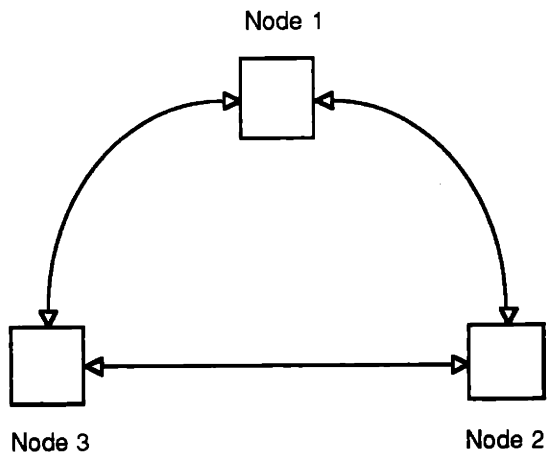


Figure 6.4 A Computer Network for Communication

Because there is no central controller, a software synchronizer is used to control the progress of the experiment. At the end of each trial, a node sends an end signal to the other nodes and checks if it has received the end signals from the others. A new trial will start only when all of the nodes in the network finish the previous trial and receive end signals from all of other nodes in the network. Figure 6.5 shows the

Net representation of the synchronizer. A transition (black bars in Fig. 6.5) on the right hand of Fig. 6.5 serves to synchronize the three nodes. As described in Chapter 2, only when each of the input places of a transition has a token, the transition is enabled and can be fired instantaneously. In Fig. 6.5, S1 denotes the source place; it contains the token which represents each trial. S2 is a resource place which implements the synchronization condition. Only when the synchronization transition is fired, will S2 have a token. Therefore, unless all nodes complete a trial, no new trial can start on any computer.

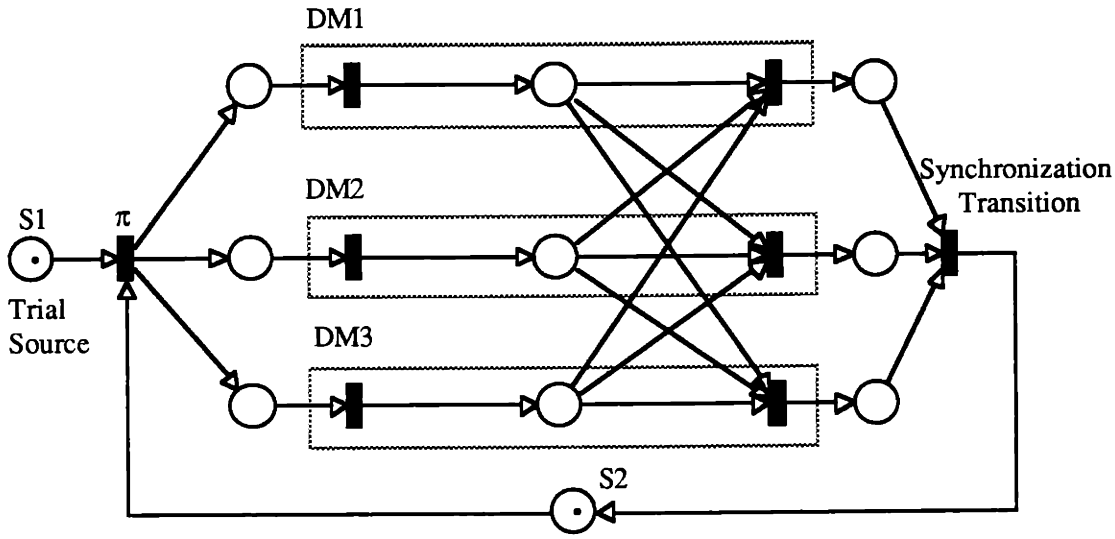


Figure 6.5 Synchronization of the Network for Each Trial

DMs are in physically separated locations to prevent visual contact and direct communication. Each DM works on one computer and performs his task. Coordination between DMs is realized by communication. The communication consists of transmission of written messages over the network. The messages are in a standard format. There is no voice communication.

6.6.2 One DM in the Organization is Played by a Computer

Although the experiment is designed for organizations with three decision makers, only two human subjects are involved in each organization. The third DM is played by a computer. The computer follows procedures and algorithms exactly and literally as designed from the model, and may show what happens if human error or adaptability is absent. The computer is programmed for this purpose. In general, the interactions in a decision making organization are not only between human decision makers, but also with computers or intelligent machines. For this reasons, it is not unrealistic to have a computer playing a role in the organization. In addition, this reduces by one third the number of human subjects required to run the experiment.

6.7 SUMMARY

In this chapter, the experimental model has been developed and described. The measured and controlled variables have been defined and discussed. Before running the experiment, a pilot experiment is necessary to test the experimental design on all the parameter values and to test the hardware and the software. When the pilot experiment is completed, the number of trials for each input condition and the number of teams needed to run the experiment will be determined. In the next chapter, the results of the pilot experiment will be discussed.

Chapter 7

PILOT EXPERIMENT

In Chapter 6, the experiment was described and the controlled and measured variables in the experiment were determined. To test the experiment design and determine the range of the controlled parameters, a pilot experiment was conducted. In this chapter, the goal of the pilot experiment is discussed first. Then, the results are represented.

7.1 THE GOAL OF THE PILOT EXPERIMENT

As described in Chapter 6, the experiment is designed on the basis of the theoretical model. However, it is necessary to determine experimentally the range of some design parameters so that the experimental data will be useful for testing the selected hypotheses. Consequently, a pilot experiment was conducted to test the experimental design.

Check the range of controlled parameters

As discussed in Chapter 6, time pressure is a major factor in organizational performance. Because of the bounded rationality of the decision makers, there exists a limitation on the human's capability for processing information and making decisions. It is important that the range of available time T_a include values of T_a which drive the processing rate of DMs to the maximum value so that organizational behavior under time pressure can be observed.

Check the correctness of the data collection

Since MOPs are computed from the experimental data, the timely, accurate, and complete recording of the observables is critical to the success of the experiment.

Test the software

The software implementing the experiment design is reasonably large-scaled and complex. Since there is no central controller, the coordination between the computers is controlled by the simple software synchronizer described in Chapter 6. Although the task procedure and possible algorithms are defined, there are variations between different human DMs on how they will do the task. Therefore, the actual behavior of this distributed system cannot be established until the experiment is run with human DMs.

The above check list is used to analyze the data of the pilot experiment. The next section discusses the results of the pilot experiment which was run with two teams, each one consisting of two DMs. Twenty trials were run by each team.

7.2 RESULTS OF THE PILOT EXPERIMENT

The data from the pilot experiment is analyzed to obtain the preliminary results. Figures 7.1 to 7.3 show plots of the accuracy versus available time, response time versus available time, and accuracy versus response time, respectively.

In Fig. 7.1, one can see that accuracy (J) decreases when the available time (T_a) decreases. When T_a is reasonably large, J does not change much, that is, the curve tends to be "flat". The flat region indicates the absence of time pressure; available time is not a critical factor affecting accuracy. Figure 7.2 shows that the response time decreases when the available time decreases and that it is always less than the available time. The difference between the two becomes smaller as T_a decreases. The plot of accuracy versus response time (Fig. 7.3) shows that higher accuracy is achieved when the response time is longer. The phenomena shown in Figs. 7.1 to 7.3 are consistent with the model predictions of the organizational behavior. Therefore, the pilot experiment validates the experiment design in general.

On the other hand, the results of the pilot experiment also led to several modifications of the original design. The following paragraphs describe these modifications.

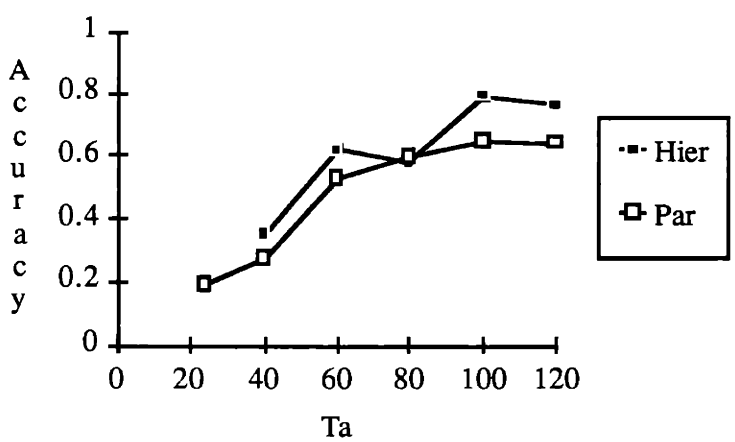


Figure 7.1 Accuracy versus Available Time

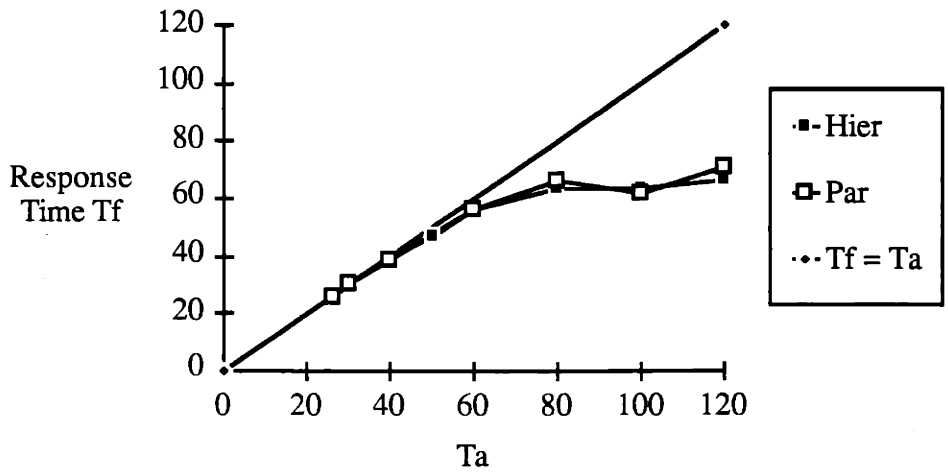


Figure 7.2 Response Time versus Available Time

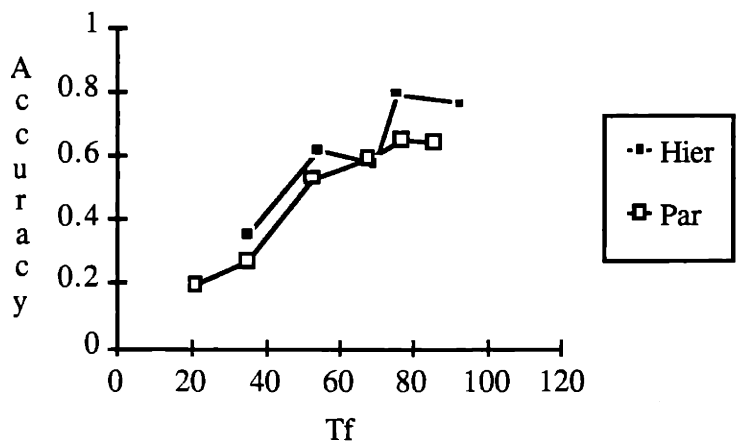


Figure 7.3 Accuracy versus Response Time

The Range of Available Time T_a

The values of the available time used in the pilot experiment are

for the hierarchical organization	T_a = { 30, 40, 60, 80, 100, 120 };
for the parallel organization	T_a = { 24, 40, 60, 80, 100, 120 }.

The difference between the values of T_a for different organizations is due to the level of required interactions in the hierarchical organization being higher than in the parallel organization.

The result shows that these ranges of T_a capture the characteristics of organizational behavior when available time changes. From Figs. 7.1 to 7.3, it is observed that there are more changes in the region of shorter available time than in the region of longer available time. Therefore, more experimental points are desired for the shorter available time to assure accurate observations. Specifically, available times of 30 seconds and 50 seconds are added for the parallel and hierarchical organizations respectively.

On the other hand, the longest available time is also checked. The period of 120 seconds seems to be appropriated for both structures. At this time condition, the slopes of both the J-T_a and J-T_f curves are smaller which indicates a reduction of time pressure.

In accordance with the result of the pilot experiment, the values for the available time T_a to be used in the experiments are:

For the hierarchical organization: $T_a = \{ 30, 40, 50, 60, 80, 100, 120 \}$ in seconds;
For the parallel organization: $T_a = \{ 26, 30, 40, 60, 80, 100, 120 \}$ in seconds.

The minimum T_a for the parallel organization is 26 seconds instead of the 24 seconds used in the pilot experiment . Figure 7.1 shows that at the shortest available time for the parallel organization (24 seconds) accuracy is only 20% which indicates that the available time is too short to do the task.

Data files

In the original design, there are two sets of data files associated with each team: a measurement data file and a history file for each structure. In the measurement data file, the experiment data which will be used to compute MOPs are stored. The history file is the file in which all the actions that occurred during the experiment are stored. For example, whenever a mouse click occurs it is recorded in the history file. The history file may be useful in the post experiment study. The measurement data file is the main data file which contains all the measures involved in the computation of the parameters in the experiment model. The main data file will be directly used for the data analysis. The pilot experiment indicated two defects in the original data collection procedure.

First, the number of communications, N_c, should be recorded explicitly in the main data file. N_c was originally recorded in the history file. However, since the number of communications is an important parameter in characterizing the coordination between the members of an organization, it needs to be in the main data file.

Second, the starting time of each trial should be recorded in the main data file. Time is critical data collected in the experiment. Because there is no central controller, each computer

records the data corresponding to its own clock. It is almost impossible to synchronize the clocks of all three computers precisely. Therefore, the starting time of each trial must be recorded as a reference time so that the processing time can be computed. The starting time was originally stored in the history file only.

Software test:

Several network collisions occurred during the pilot experiment. Because the experiment is a real time simulation and a simple communication protocol has been adopted, network collisions cannot be prevented completely. The best approach is to add more software control to keep the probability of collision to a minimum. To accomplish this, the communication portion of the software was modified.

After these improvements were done, the sample sizes needed to be determined.

7.3 DETERMINATION OF SAMPLE SIZES

The sample sizes are the number of trials for each time condition and the number of teams. The determination of sample sizes are described in this section.

Number of Trials

The number of trials required for each input condition needs to be determined so that the sample mean can be used to estimate the population mean within a desired confidence interval. The experimental data set must be large enough to support conclusions about the hypotheses. On the other hand, if the sample size is larger than necessary, there will be a waste of time and money. Because human subjects are involved in the experiment, a large number of trials is not feasible. Therefore, the sample size should fit both statistical and practical considerations.

The sample size required depends on the input attributes. The input attributes are: 1) speed of threats, V; 2) number of threats, N; 3) number of aircraft in a threat, m (there are N m's for each trial). Since the speed of the threats uniquely determines the available time for a trial, T_a, it is more convenient to use T_a instead of V. Writing the input attributes in vector form, we have

$$x = \{ T_a, N, \underline{m} \}$$

where $\underline{m}$ is a vector of dimension N because each threat there has an m value associated with it.

Since the available time T_a is chosen deterministically and N and m are generated stochastically, the estimation of the sample size depends only on N and m. Table 7.1 shows the sample spaces and probabilities of N.

Table 7.1 Values and Probability Distributions of N

Hierarchical organization		Parallel organization	
N	P(N)	N	P(N)
5	0.33	3	0.17
6	0.49	4	0.49
7	0.18	5	0.34
Mean N: $\mu(N)$	5.84		4.18

The values of m are selected to be

$$m = \{ 2, 3, 4, 5, 6 \}, \text{ and}$$

the probability distribution of m is uniform. Since there are five possible values for m, the probability that m takes one of the five values is:

$$p(m) = \frac{1}{5} = 0.2$$

The determination of the sample size is described as follows.

Assume that the limiting error (maximum error) in estimating the population mean is c and the confidence interval desired is 95%. Let the sample mean be X_n computed from n samples and let the population mean be μ. Then, these specifications can be written as

$$P(| X_n - \mu | < c) = 95\% \tag{7.1}$$

where X can be either N or m.

The Law of Large Numbers states that X_n converges to μ in probability when n is increases. Therefore, X_n is close to μ when the sample size n is very large, that is,

$$P(| X_n - \mu | > c) \rightarrow 0$$

where c is any number larger than zero.

The Central Limit theorem allows an approximation on how close that X_n is to μ: when n is very large,

$$p(|X - \mu| < c) = \Phi(c\sqrt{\frac{n}{\sigma^2}}) = \Phi(z) \tag{7.2}$$

where Φ is a normal cumulative distribution function.

Applying the Central Limit theorem (with 95% confidence interval and using the normal table), the z value is

$$z(1-\alpha/2)=1.96,$$

where $\alpha = 0.05$.Then, from equation (7.2),

$$c\sqrt{\frac{n}{\sigma^2}} = 1.96$$

or

$$n = (\frac{1.96\,\sigma}{c})^2 \tag{7. 3}$$

Equation (7.3) is used to determine the sample size. Table 7.2 shows how the sample sizes vary with c, the error.

Table 7.2 Number of Trials (n) with Corresponding Error

error	n (Parallel: all DMs)	n (Hierarchical: DM1 & 3)	n (Hierarchical: DM2)	n (m)
0.1	186	186	403	769
0.2	47	47	101	193
0.3	21	21	45	86
0.4	12	12	25	49
0.5	8	8	17	31
0.6	6	6	12	22
0.63	5	5	11	20
0.7	4	4	9	16
0.8	3	3	7	12
0.9	3	3	5	10
1	2	3	5	8

The error in the vicinity of 0.5 is a reasonable choice because the mean values of N are 4.18 and 5.84 for the parallel and hierarchical organizations, respectively. When practical consideration regarding the availability of human subjects are taken into account, then the following sample sizes are obtained:

With 95% confidence level:

Parallel organization:	n_p (# of trial) = 8, error $\leq$ 0.5;
Hierarchical organization:	n_h (# of trial) = 10, error $\leq$ 0.63;

where the number of trials is for each value of the available time.

The reason for the number of trials in the hierarchical organization being larger than in the parallel organization is that the distribution of N corresponding to the supervisor (DM2) needs more trials (Table 7.2) to satisfy the constraint on the maximum error. For a maximum error of 0.5, the number of trials is 16, which is double the number for the parallel one. Then, the total number of trials for all time conditions will be 58 for the parallel organization and 112 for the hierarchical organization. The sample sizes for two organizational structures are so different that it may cause inconsistency or bias in experimental results. On the other hand, 168 (112+56) of total trials will take three hours and eight minutes for each team to run the experiment, which is quite difficult for an experiment involving a large number of humans. By relaxing the error criterion slightly (the maximum error is selected as 0.63), the number of trials for the hierarchical organization will be reduced to 10 which is compatible with the parallel organization; the total number of trials will be 70, about one third reduction from 112 trials. Therefore, the number of trials for the hierarchical organization was set at 10.

In the multi-parameter case, the sample size should be determined by choosing the largest number required by the parameters involved. However, since there are N threats in each trial, there are N values of m. The total expected number of aircraft in a trial can be computed as

$$\mu * n_p = 4.18 * 8 = 33.4 \quad \text{for the parallel organization;}$$
$$\mu * n_h = 5.84 * 10 = 58.4 \quad \text{for the hierarchical organization.}$$

Table 7.2 shows that the required sample sizes for m corresponding to the selected error are 30.73 and 19.36 for the parallel and hierarchical organizations, respectively. Therefore, the required sample size of m is satisfied when the sample sizes are chosen according to the requirement for N. As a result, the number of trials for parallel organization is 8 while for the hierarchical organization is 10.

After the sample size is determined, the experiment structure is established. Figure 7.4 shows the structure.

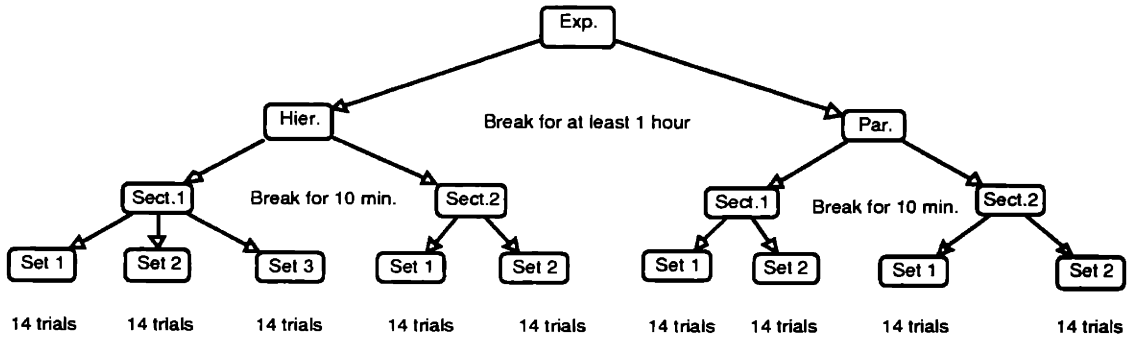


Figure 7.4 The Experiment Structure

The experiment consists two parts: the trials for the hierarchical organization and the trials for the parallel organization.The total numbers of trials is 70 for the hierarchical organization and 56 for the parallel organization. It takes two hours and 20 minutes for each team to run the experiment for both organizations. The experiment is divided into sections. There are breaks between sections to avoid fatigue effects on subjects.

Number of Teams

There are two considerations in determining the number of teams. The first one is to collect enough data to test the hypotheses. The second consideration is feasibility. Since there are human subjects involved in the experiment, a large number of teams is not feasible. However, the number of teams required cannot be decided exactly using the same method as for the number of trials because the probability distributions of measured variables are unknown.

To estimate the number of teams, MOPs are considered. As described in Chapter 4, MOPs consist of three measures: accuracy and response time for the organization, and cognitive workload for individual DMs. While there are no data available to estimate the probability distribution of accuracy and response time, a previous experiment provides information related to the cognitive workload. A single-person experiment has been run to investigate the bounded rationality constraint of human cognitive processes (Louvet, Casey and Levis, 1988).

From the single-person experiment, the bounded rationality of human DMs represented by the maximum processing rate has been observed. The probability distribution of the maximum rate has been found to be a normal distribution. Although the specific value of the maxi-

mum rate may depend on the task, the variance of the maximum rate reflects the bounded rationality which is independent of tasks. Therefore, the variance of the maximum rate from that experiment is used to estimate the number of subjects needed to capture the individual differences.

Although the main interest in this thesis is team performance, the behavior of the individual DMs is essential. Because there is no other data available, the use of individual information will provide insights on the number of teams required.

Table 7.3 shows the average maximum processing rate, F_{max}, and its standard deviation as found in the single-person experiment.

Table 7.3 Mean and Standard Deviation of F_{max}
(Unit: bits per second)

Mean	40.83
Standard Deviation	13.013

To use this information to determine the number of teams needed for the multi-person experiment, the following analysis is conducted. Let k denote the number of subjects needed to run the multi-person experiment. Using the same method as for determining the number of trials, the following equation can be used to compute k:

$$k = \frac{z\left(1 - \frac{\alpha}{2}\right)\sigma^2}{c^2} \tag{7.4}$$

where $(1-\alpha/2)$ is the confidence interval; z is $(1-\alpha/2)\%$ percentile of the normal distribution since the average value of the maximum rate was found to have a normal distribution; σ^2 is the variance of the maximum rate; and c is the maximum error. Because the sample size can not be very large, the t distribution is substituted for the normal distribution. Then equation (7.4) becomes

$$k = \frac{t\left(1 - \frac{\alpha}{2}, \nu\right)\sigma^2}{c^2} \tag{7.5}$$

where t represents the t distribution; ν is the degree of freedom which is equal to the number of sample minus the number of parameters estimated. In this case, two parameters are estimated: the mean and the variance of the maximum rate. Therefore, ν equals to k - 2. When a 95% confidence interval is desired, α is 0.05 and

$$t(1-\frac{\alpha}{2}, \nu) = t(0.975, k-2) \tag{7.6}$$

From the t distribution table (Neter et al, 1978), if $\nu = 20$ is used in equation (7.6), the value of t is found to be 2.086 and of k is 22. From equation (7.5), the limiting error c can be computed by

$$c = \sqrt{\frac{t(1-\frac{\alpha}{2})\,\sigma^2}{k}} = \sqrt{\frac{2.086 \times 13.013^2}{22}} \tag{7.7}$$

Table 7.4 shows the values of the maximum error c related to the degree of freedom ν calculated using equation (7.7) with 95% confidence interval.

Table 7.4 Maximum Error and Number of Degrees of Freedom

ν	t(1-a/2)	c
20	2.086	4.0061
22	2.074	3.8245
24	2.064	3.6656
26	2.056	3.5254
28	2.048	3.3992
30	2.042	3.2865
40	2.021	2.8539
60	2	2.3367
120	1.98	1.6574

From Table 7.4, it can be seen that error decreases with an increase in the degree of freedom. However, when the degree of freedom increases from 20 to 120, the error is reduced from about 4.0 bits per second to 1.7 bits per second. The reduction of error is not very large compared with the mean value for F_{max} which is about 40 bits per second (Table 7.3). Therefore, considering the maximum error and the feasibility of running experiments with human subjects, the degree of freedom was set at 28. The corresponding maximum error is 3.4.

As a result, k is 30, or, the number of subjects needed to run the experiment is 30. Since there are two human DMs in each team, 15 teams can be formed.

In summary, the pilot experiment served as a complete test of the experiment design. Insights gained from the pilot experiment led to improvements of the main experiment design, especially in the selection of sample sizes and ranges for the controlled parameters.

Chapter 8

EXPERIMENTAL RESULTS

The experiment was run with fifteen two-person teams, for a total of 30 subjects. Among the subjects, 28 were students and two were MIT employees. Seven of the 28 students were graduate students, nineteen were undergraduate students, and two were middle school students. Both MIT employees had college or graduate degrees. The teams were numbered 3 to 17; teams 1 and 2 were the ones that participated in the pilot experiment. The experiment was carried out during the winter Independent Study Period 1990 at MIT, which is about one month long.

In Section 1 of this chapter, the data recorded in the experiment are described. The computation of MOPs is presented in Section 2. In Section 3, the experimental results are presented and discussed. In Section 4, the results are used to test the two hypotheses formulated in Chapter 5.

8.1 DATA COLLECTION

The experimental data for each team consists of three data files generated during the experiment, one file for each DM. Only the main data files, as described in Chapter 7, are used in the data analysis. History files serve as a reference when needed. In the following discussion, the phrase "data file" refers to the main data file unless otherwise specified. Each data file contains information about the actions that a DM has taken during the experiment. The content of the data files varies with the organizational structure and the position of the DM in the organization. The following paragraphs describe these data files in detail.

Data Files for the Parallel Organization

For the parallel structure, the format of the data files for all team members is the same.The data stored in the data file is listed in Table 8.1. Figure 8.1 shows a part of a data file. The meanings of the heading of the columns in Fig. 8.1 is also shown in Table 8.1.

Table 8.1 Data Listed in Data Files for Parallel Organization

A: trial number	L: trail starting time*
B: threat ID	M: # of F14 allocated
C: speed of the threat in miles per hour	N: # of F18 allocated
D: bearing of the threat in degrees (θ)	O: # of EA-6B allocated
E: number of aircraft in the threat (m)	P: strategy for SA (u)
F: threat type	Q: strategy for RS (v)
G: assigned type after IF	R: Algorithm used by other DM
H: received type through communication	S: accepting the type in IF: 1: no; 2: yes
I: class of threat	T : estimate type at 1:inner; 0:outer circle
J: start time of processing a threat*	U: to whom the message is sent
K: finishing time of processing a threat*	V: waiting time*

* unit is in seconds

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
1	4	300	100	3	S	S	x	s	43.9	50.6	37.55	0	0	0	1	0	0	1	0	1	0.0
1	3	300	85	3	S	S	S	s	60.3	83.5	37.55	0	1	0	1	1	1	2	0	0	0.0
1	9	300	325	5	B	B	x	s	52.9	101.0	37.55	0	0	0	1	0	0	1	1	2	0.0
1	1	300	355	6	B	B	0	s	101.3	110.5	37.55	2	0	0	1	1	0	1	1	0	0.0
1	2	300	25	3	S	S	0	s	112.1	119.3	37.55	0	1	0	1	1	0	1	1	0	0.0
2	4	360	100	2	S	S	x	s	170.1	176.4	163.43	0	0	0	1	0	0	1	0	1	0.0
2	9	360	325	4	S	S	x	s	177.8	182.8	163.43	0	0	0	1	0	0	1	0	1	0.0
2	3	360	85	2	S	S	S	s	184.2	199.8	163.43	0	0	2	1	1	1	1	1	0	0.0
2	2	360	40	4	S	S	0	s	201.3	207.6	163.43	1	0	0	1	1	0	1	1	0	0.0
2	1	360	355	2	S	S	0	s	208.9	224.0	163.43	0	0	2	1	1	0	1	1	0	0.0
3	4	450	100	3	F	F	x	m	275.6	285.4	269.20	0	0	0	1	0	0	1	0	1	0.0
3	9	450	325	6	B	B	x	m	286.9	301.5	269.20	0	0	0	1	0	0	1	0	2	0.0
3	3	450	85	2	S	S	F	m	302.5	322.6	269.20	0	0	2	1	1	1	1	1	0	0.0
3	2	450	40	4	F	F	0	m	323.3	340.4	269.20	2	0	0	1	1	0	1	1	0	0.0
3	1	450	355	3	F	F	0	m	341.6	348.5	269.20	2	1	0	1	1	0	1	1	0	0.0

Figure 8.1 Data File Format for the Parallel Organization

Data Files for Hierarchical Organization

For the hierarchical organization, data files for subordinates have a different content from the data file of the supervisor. Figures 8.2 and 8.3 illustrate the data files for subordinates and the supervisor.

In Fig. 8.2, all the headings mean the same as in Table 8.1 except for:

H: assigned type before Command Interpretation (CI),

R: accepting command exactly (0: no; 2: yes),

S: waiting time.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	5	300	15	2	S		x	s	19.9	23.9	13.30	0	0	0	0	0	0	0.0
1	8	300	245	5	B		x	s	24.6	29.6	13.30	0	0	0	0	0	0	0.0
1	4	300	85	3	S	S		s	15.4	50.0	13.30	0	1	0	0	0	2	0.0
1	3	300	45	4	S	S	0	s	33.0	54.8	13.30	0	0	4	1	0	0	0.0
1	2	300	25	5	S	B	0	s	36.3	59.4	13.30	0	0	5	1	0	0	0.0
1	1	300	5	6	B	B	0	s	41.4	63.7	13.30	1	1	1	1	0	0	0.0
2	5	360	115	2	S		x	s	148.4	53.3	140.60	0	0	0	0	0	0	0.0
2	2	360	45	2	S	S	0	s	162.8	178.4	140.60	0	0	2	1	0	0	0.0
2	1	360	25	3	S	S	0	s	179.9	184.2	140.60	0	0	3	1	0	0	0.0
2	4	360	85	6	B	B		s	154.4	186.9	140.60	2	0	0	0	0	2	0.0
2	3	360	65	6	B	B		s	158.3	189.1	140.60	2	0	0	0	0	2	0.0
3	8	450	245	2	S		x	m	261.4	265.4	248.47	0	0	0	0	0	0	0.0

Figure 8.2 Data File Format for Hierarchical Organization: Subordinate

In Fig. 8.3, all the headings have the same meanings as in Table 8.1 except for:

H: received type through communication. In the experiment, this parameter is not used because the subordinates do not consider the type when sending a message to the supervisor;

R: to whom the command is sent.

S: waiting time, in seconds, for messages from both sectors.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	5	300	205	2	S	S	u	s	26.05	46.70	21.45	0	0	2	1	1	2	10.03
1	5	300	205	2	S	0	u	s	36.18	0.00	21.45	0	0	0	0	0	0	10.03
1	4	300	175	3	S	S	u	s	41.97	62.57	21.45	0	1	0	1	1	1	10.03
1	4	300	175	3	S	0	u	s	25.30	0.00	21.45	0	0	0	0	0	0	10.03
1	8	300	335	5	B	B	u	s	49.23	78.83	21.45	1	1	0	2	1	2	10.03
1	8	300	335	5	B	0	u	s	26.70	0.00	21.45	0	0	0	0	0	0	10.03
2	5	360	205	2	S	S	u	s	154.43	169.03	148.57	0	0	2	1	1	2	2.40
2	5	360	205	2	S	0	u	s	155.00	0.00	148.57	0	0	0	0	0	0	2.40
2	4	360	175	6	B	B	u	s	161.08	176.03	148.57	2	0	0	1	1	1	2.40
2	4	360	175	6	B	0	u	s	153.75	0.00	148.57	0	0	0	0	0	0	2.40
2	3	360	155	6	B	B	u	s	165.35	184.60	148.57	2	0	0	1	1	1	2.40
2	3	360	155	6	B	0	u	s	153.02	0.00	148.57	0	0	0	0	0	0	2.40

Figure 8.3 Data File Format for Hierarchical Organization: Supervisor

8.2 COMPUTATION OF MOPs

The Measures of Performance are accuracy (J) and response time (T_f) for the organization and cognitive workload (G) for the individual DMs. The following paragraphs explain the computation. Detailed computations can be found in Appendices B and D.

Accuracy

Accuracy, J, is computed using directly measured data in the experiment. Data used for the computation are type and resources assigned to a threat. The decision is accurate only if both the type and the resource allocation are correct. Values of J are between zero and one. J equal to one indicates that there is no error. For detailed information on computing accuracy, see Appendix D. The result of this computation is, for each trial, one index representing the organizational accuracy and three individual indices for each of the three decision makers.

Response Time

Response time is computed in two steps for each trial. First, processing time of DMi, Tp^i, is computed for a trial.

$$T_p^i = \max_j t_{ej} - t_0$$

where t_{ej} is the time that the process corresponding to the jth threat in a trial is completed (or the jth threat is attacked); t_0 is the time the trial starts. The maximum difference between t_e and t_0 is the processing time for the trial.

Then, according to the organizational structure, the organizational response time T_f is computed using the individual processing time. Specifically,

for the hierarchical organization: $T_f = \max_i Tp^i$, $i = 1, 3$ (DM1 and DM3);

for the parallel organization: $T_f = \max_i Tp^i$, $i = 1, 2, 3$ (All DMs).

Workload

The algorithms used in the SA and RS stages of the experiment are recorded in the data file (columns 16 and 17 in Figs. 8.1 to 8.3). The probability that a particular algorithm is used can be computed and, therefore, the strategy can be determined. The average workload G of each DM for each of the time conditions can be computed if the strategy used is known. The computation is the same as described in Chapter 5 (Appendix B).

Number of Communications

The number of communications, N_c, is the actual number of communications that take place in each trial. Only outgoing messages are recorded because communication is always initiated by a sender. This information is directly obtained from the data file. In Figs. 8.1 and 8.2, the seventh column indicates communication. Specifically, "x" in the seventh column of Fig. 8.1 and a non-zero value in the seventh column of Fig. 8.2 indicate the occurrence of communication. For the case of the supervisor in the hierarchical organization (Fig. 8.3), the presence of a non-zero value in column K indicates that communication occurred.

The next two measures are the time ratio and the communication ratio. These ratios are suggested by dimensional analysis. They can be computed from the experimental data.

Time Ratio

The time ratio is computed as

$$t = \frac{T_f}{T_a}$$

For each time condition, the average time ratio is computed for each team.

Communications Ratio

The ratio of the actual number of communications and the number of task-required communications is computed as

$$n_c = \frac{N_c}{N_{rc}}$$

The total number of communications for each time condition is computed by adding the number of communications across trials. This ratio is computed for each time condition and each team.

The results are stored in the new data files which are ready to be used for the hypothesis testing.

8.3 EXPERIMENTAL RESULTS

In this section, the observations from the experimental data are discussed. Two important results are drawn from these observations: (a) interaction between organizational members compensated for individual differences; and (b) coordination by the supervisor in the hierarchical organization reduced the variance of organizational performance.

General Observations

The relation between accuracy J and the available time T_a was investigated for each team and each organizational structure.

The accuracy measure for all teams and both organizational structures as a function of the available time is presented in Tables 8.2 and 8.3.

There are no significant differences in accuracy for the different organizations (A test in Appendix E shows this conclusion). However, the interesting phenomenon is that the behaviors of the organizations changes under different tempos of operations. This is illustrated in Fig. 8.4 where J versus T_a is plotted for team #10. Accuracy of the parallel organization is better than the hierarchical organization's when the available time is short while the situation is

Table 8.2 Accuracy for All Teams: Hierarchical Organization

Team	Ta = 30	Ta = 40	Ta = 50	Ta = 60	Ta = 80	Ta = 100	Ta = 120
3	0.49	0.75	0.81	0.83	0.88	0.96	0.96
4	0.43	0.63	0.72	0.81	0.91	0.95	0.98
5	-	0.85	0.98	0.92	0.97	0.99	0.99
6	-	0.72	0.93	0.89	0.99	0.99	1.00
7	-	0.69	0.92	0.91	0.98	1.00	1.00
8	-	0.49	0.59	0.73	0.87	0.97	0.98
9	-	0.60	0.84	0.70	0.87	0.98	0.98
10	0.50	0.76	0.92	0.91	0.98	0.99	0.99
11	-	0.63	0.79	0.82	0.99	0.97	0.98
12	-	0.67	0.93	0.92	1.00	0.98	0.99
13	0.42	0.64	0.74	0.81	0.96	0.99	0.95
14	0.50	0.70	0.86	0.95	0.99	1.00	0.97
15	0.49	0.87	0.87	0.94	0.96	0.95	1.00
16	0.50	0.80	0.86	0.85	1.00	0.98	0.95
17	0.56	0.69	0.83	0.80	0.94	0.99	0.93
Mean	0.48	0.70	0.84	0.85	0.95	0.98	0.98
St. Dev.	0.04	0.10	0.10	0.08	0.05	0.02	0.02

opposite when the available time becomes moderate. Furthermore, at very slow tempo of operations, there is no difference between the accuracy of the two organizations. Therefore, in general, it is not possible to state that a fixed organizational structure will have better performance for all possible operating conditions.

Since differences in average performance between the hierarchical and the parallel organizations are insignificant, a study of the variance is conducted. The results are as follows.

Table 8.3 Accuracy for All Teams: Parallel Organization

Team	Ta = 26	Ta = 30	Ta = 40	Ta = 60	Ta = 80	Ta = 100	Ta = 120
3	-	0.64	0.75	0.82	0.80	0.82	0.95
4	0.48	0.60	0.77	0.86	0.93	1.00	1.00
5	-	0.84	0.93	0.92	0.96	0.99	0.98
6	-	0.67	0.93	0.87	0.94	1.00	0.99
7	-	0.71	0.86	0.92	0.95	0.98	0.99
8	-	0.42	0.58	0.67	0.79	0.94	0.98
9	-	0.53	0.72	0.73	0.85	0.99	0.93
10	-	0.60	0.83	0.87	0.89	0.98	0.98
11	-	0.55	0.78	0.77	0.88	1.00	0.99
12	-	0.60	0.81	0.89	0.93	0.99	0.99
13	0.56	0.66	0.71	0.85	0.91	0.95	1.00
14	0.68	0.63	0.79	0.83	0.88	0.98	0.98
15	0.65	0.75	0.92	0.95	0.94	0.97	0.99
16	0.55	0.71	0.74	0.85	0.91	0.99	0.99
17	0.41	0.48	0.52	0.81	0.78	0.99	0.99
Mean	0.55	0.63	0.78	0.84	0.89	0.97	0.98
St. Dev.	0.10	0.11	0.12	0.07	0.06	0.05	0.02

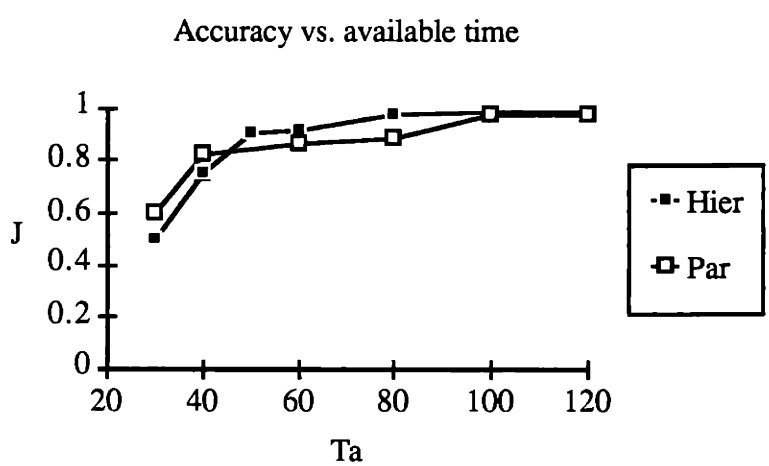


Figure. 8.4 Accuracy and Available Time for a Team: Team #10

Organization versus Individuals

When the amount of work required by a task is such that it cannot be handled by a single person, an organization is formed. A properly designed organization will maintain performance at a desired level. Furthermore, organization performance should not be sensitive to variations in individual skills. Figures 8.5 and 8.6 show a comparison of the standard deviation of the accuracy measure for each organization and for individual DMs in the two structures.

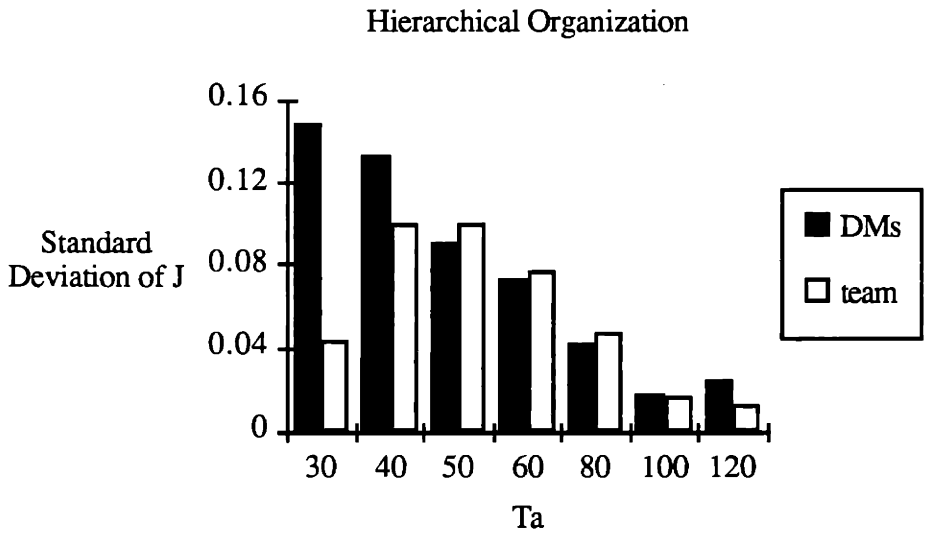


Figure 8.5 Standard Deviation of J for Teams and Individuals: Hierarchical

When the available time is long enough to do the task, the standard deviations between the teams and individuals are very close because the task can be completed accurately and the error is random. However, when time is decreased, individual differences in skills, experience, and capabilities are revealed. The standard deviation of individual performance is high. On the other hand, the organizational performance is more stable. This observation can be explained as follows.

An organization is designed so that the task is divided into subtasks and allocated to all organizational members. Each of the members in the organization will interact with other organizational members and contribute a part of the effort to perform the task. The decisions of one member will affect the decisions of the other. Therefore, compensatory behavior between the organizational members reduces the variance of organizational performance. As a consequence, organizational performance is less sensitive to individual difference. Both Fig. 8.5 and Fig. 8.6 show that the variance of the accuracy measure for teams is much smaller than that of individuals during the fast tempo of the operations.

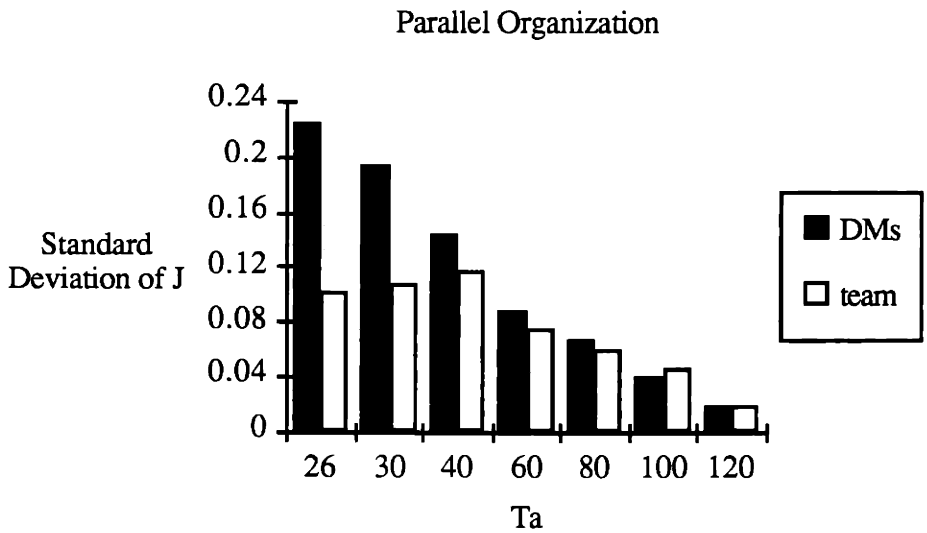


Figure 8.6 Standard Deviation of J for Teams and Individuals: Parallel

Although the actual variance is different in the different structures, the phenomenon is observed for both structures. Therefore, it can be concluded that team work reduces the effect of individual differences on performance.

Effects of Organizational Structures

Figure 8.7 shows the comparison of the standard deviation of the accuracy J for the two structures. Table 8.4 presents the numerical data of Fig. 8.7. The standard deviations are computed across 15 teams for both organizational structures. Tt can be seen from Fig. 8.7 that for most values of T_a, J has smaller standard deviation in the hierarchical organization than in the parallel organization. This implies that J of the hierarchical organization is more robust with respect to the individual differences than that of the parallel organization.

The difference in the standard deviation reflects the organizational effects. As already stated, the interaction level in the hierarchical organization is higher than that in the parallel organization. In terms of making decisions, DMs in the parallel organization have more "freedom" to choose what to do than those in the hierarchical organization. Therefore, it is expected that individual difference will have more influence on performance in the parallel

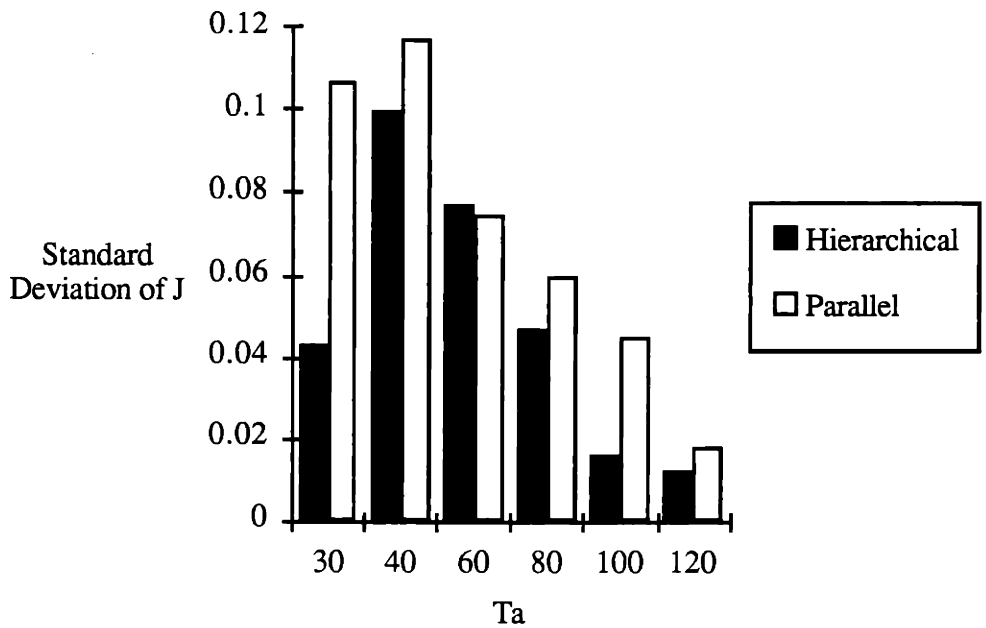


Figure 8.7 Standard Deviation of J for the Two Structures

Table 8.4 Variance and Standard Deviation of J

	Hierarchical	Parallel	Hierarchical	Parallel
Ta	Variance	Variance	Standard Dev.	Standard Dev.
30	0.00187	0.01136	0.0432	0.1066
40	0.00994	0.01374	0.0997	0.1172
60	0.00596	0.00552	0.0772	0.0743
80	0.00222	0.00357	0.0471	0.0598
100	0.00025	0.00204	0.0160	0.0451
120	0.00010	0.00032	0.0122	0.0180

organization. On the other hand, the interactions in the hierarchical organization restrict the choices of the decision makers and couple individual decisions with the decisions of other organization members. As a result, individual characteristics tend to be suppressed in the organizational performance.

The confidence level for variance is used to infer the population variance from the sample variance. The following theorem provides the basis of the test (Neter et al, 1978).

If a random sample of size n is selected from a normal population with variance σ^2, then:

$$\frac{(n - 1)\, s^2}{\sigma^2} = \chi^2(n - 1) \tag{8.1}$$

where χ^2 is the Chi-square distribution; s^2 is sample variance.

A two-side confidence interval for the population variance σ^2 with confidence coefficient $(1 - \alpha)$ is

$$L \leq \sigma^2 \leq U$$

where

$$L = \frac{(n - 1)s^2}{\chi^2(1 - \frac{\alpha}{2}; n - 1)} \quad ; \qquad \text{and}$$

$$U = \frac{(n - 1)s^2}{\chi^2(\frac{\alpha}{2}; n - 1)}$$

For this test, the confidence level is selected to be 95%. Then, for α set at 0.05. and for n, the number of teams, set at 15, the results of the test are shown in Table 8.5.

In Table 8.5, variances are sample variances computed by using the experimental data. All sample variances satisfy equation (8.1). Therefore, the variances of accuracy computed from the experimental data can satisfy the confidence level of 95%.

The general conclusion is that when there are more interactions among the organizational members, individual differences have a less pronounced effect on overall performance. In contrast, at low level of interaction, each individual's performance affects more directly the organizational output.

Table 8.5 Test Result for the Variance of Accuracy

	Hierarchical organization			Parallel Organization		
Ta	L	U	Variance	L	U	Variance
30	0.001	0.0046	0.0019	0.0061	0.0283	0.0114
40	0.005	0.0247	0.0099	0.0074	0.0342	0.0137
60	0.003	0.0148	0.0060	0.0030	0.0137	0.0055
80	0.001	0.0055	0.0022	0.0019	0.0089	0.0037
100	0.0001	0.0006	0.0003	0.0011	0.0051	0.0020
120	8E-05	0.0004	0.0001	0.0002	0.0008	0.0003

Next, the experimental results will be compared with the model predictions from Chapter 5.

Comparison of Model Prediction and Experimental Results

The theoretical model and evaluation procedure described in Chapter 5 will be applied now to predict organizational performance. Given an organizational strategy, performance can be computed. Specifically, accuracy corresponding to a behavioral strategy is computed by

$$J(\Delta) = \sum_{k^1k^2k^3} J_{k^1k^2k^3} \cdot p^1_{k^1} \cdot p^2_{k^2} \cdot p^3_{k^3} \tag{8.5}$$

where J_{k1k2k3} is the accuracy measured indexed for a pure organizational strategy (Appendix B); p^i is the strategy used by DMi.

To compare the model prediction and the experimental results, the strategies recorded in the experiment are used in equation (8.5) to obtain the accuracy J_{pred}. The accuracy computed from experimental data, J_{exp}, is compared then with J_{pred}. Let ΔJ denote the absolute value of the difference between J_{pred} and J_{exp}:

$$\Delta J = | J_{pred} - J_{exp} | \tag{8.6}$$

ΔJ is computed for all teams and all values of available time. Table 8.6 shows that the predicted and experimental values of J are very close for most of the teams. Since the range of J is between zero and one:

$$0 \leq J \leq 1.0,$$

an average error of prediction of 0.1 is quite small. Therefore, the model and the evaluation procedure predict organizational performance at least for the class of tasks which are highly structured, well-defined and under a time pressure.

Table 8.6 Comparison between Model Prediction and Experimental Result

	Hierarchical			Parallel		
	fast	med	slow	fast	med	slow
Team	ΔJ	ΔJ	ΔJ	ΔJ	ΔJ	ΔJ
3	0.06	0.02	0.06	0.03	0.03	0.07
4	0.12	0.02	0.06	0.07	0.09	0.13
5	0.17	0.10	0.09	0.13	0.10	0.09
6	0.08	0.06	0.07	0.09	0.08	0.11
7	0.06	0.09	0.10	0.08	0.11	0.11
8	0.20	0.03	0.06	0.21	0.06	0.08
9	0.02	0.07	0.07	0.09	0.01	0.08
10	0.01	0.07	0.09	0.06	0.13	0.10
11	0	0.04	0.09	0.04	0.01	0.12
12	0.08	0.09	0.09	0.01	0.09	0.11
13	0.11	0.02	0.08	0.07	0.06	0.10
14	0.02	0.10	0.10	0.01	0.03	0.10
15	0	0.12	0.06	0.06	0.13	0.11
16	0.02	0.06	0.06	0.04	0.07	0.12
17	0.04	0	0.06	0.24	0	0.12
Mean	0.066	0.0593	0.076	0.082	0.0667	0.1033
Variance	0.0038	0.0013	0.0003	0.0044	0.0019	0.0003

8.3.2 Response Time

Tables 8.7 and 8.8 show the numerical values of the response time T_f that correspond to each value of available time T_a for two different teams.

Table 8.7 Response Time of Team #7

Ta	Tf(Hierarchical)	Tf (Parallel)
30	-	30.00
40	39.33	38.86
50	43.91	-
60	56.93	57.35
80	63.36	69.56
100	61.03	65.87
120	60.09	69.13

The relationship between response time and available time for the two teams are shown in Figs. 8.8 and 8.9.

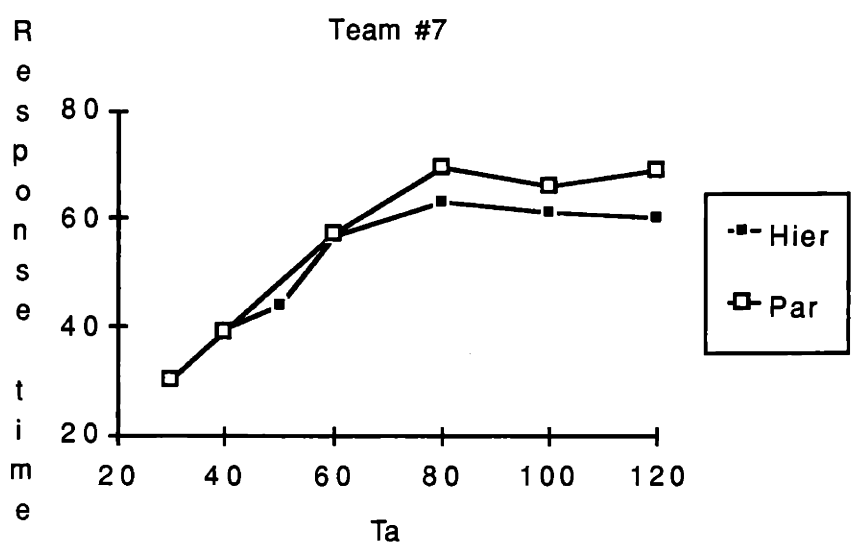


Figure 8.8 Response Time and the Available Time for Team #7

Table 8.8 Response Time of Team #10

Ta	Tf(Hierarchical)	Tf(Parallel)
30	29.86	30.00
40	37.76	39.70
50	44.31	-
60	55.51	56.51
80	60.34	66.74
100	56.97	65.48
120	60.74	77.43

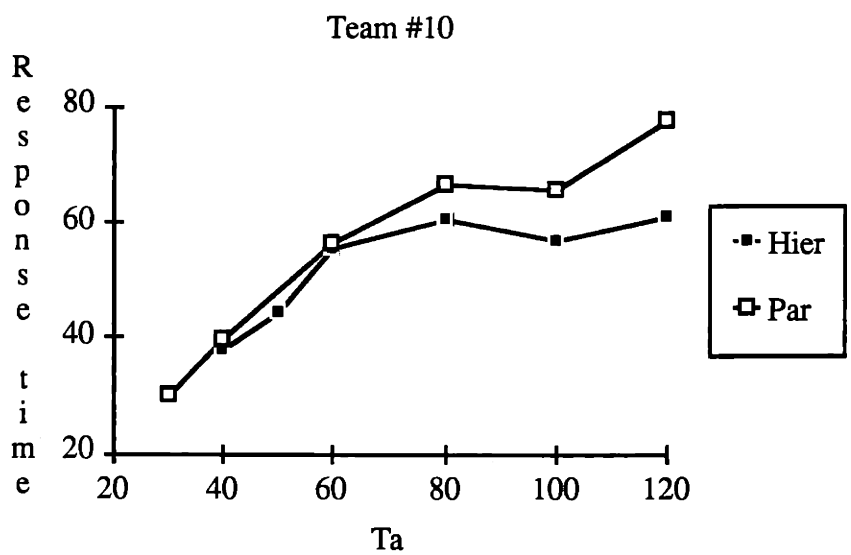


Figure 8.9 Response Time and the Available Time for Team #10

The experimental results for the response time for all teams and both organizations are summarized in Tables 8.9 and 8.10.

Table 8.9 Response Time For All Teams: Hierarchical Organization
(unit: in seconds)

Team	Ta=30	Ta= 40	Ta = 50	Ta = 60	Ta = 80	Ta =100	Ta = 120
3	30.00	39.50	48.11	57.19	57.60	57.13	63.19
4	30.00	39.83	48.01	55.84	70.51	74.99	71.56
5	-	37.52	39.01	49.87	52.82	49.24	50.35
6	-	39.88	43.55	56.15	62.12	60.71	56.38
7	-	39.33	43.91	56.93	63.36	61.03	60.09
8	-	40.00	50.07	57.02	70.35	75.43	74.50
9	-	40.00	47.76	58.27	70.50	61.46	61.72
10	30.00	37.91	44.45	55.57	60.69	57.42	60.74
11	-	40.00	49.86	58.98	69.61	74.51	78.29
12	-	40.00	44.77	55.40	61.55	69.39	70.32
13	30.00	39.29	48.30	58.01	67.85	60.52	71.49
14	30.00	38.69	47.85	53.88	60.56	62.01	68.76
15	30.00	37.46	47.05	52.22	60.17	55.76	64.19
16	30.00	39.26	46.54	56.73	62.82	57.46	64.75
17	30.00	39.70	48.68	57.41	69.25	62.72	75.17
Mean	30.00	39.23	46.53	55.96	63.98	62.65	66.10
St.Dev	0.00	0.91	2.92	2.40	5.44	7.68	7.68

By definition, the response time is always less than the available time. However, at small values of the available time, the response time is almost equal to T_a. The difference between T_f and T_a becomes larger as T_a increases. Let D denote the difference in the means of the response time for the two organizations. Then, the following hypothesis can be formulated that addresses the question of difference in speed of response between the two organizations.:

H0: $D \geq \delta$, structure x is faster than structure y.
H1: $D < \delta$, structure x is not faster than structure y.

where $D = T_f(x) - T_f(y)$ and δ is a threshold value for the difference. The confidence level required is 95%.

Table 8.10 Response Time for All Teams: Parallel Organization
(unit: in seconds)

Team	Ta=26	Ta= 30	Ta = 40	Ta = 60	Ta = 80	Ta =100	Ta = 120
3		30.00	37.52	50.75	59.30	59.12	63.88
4	26.00	30.00	39.69	56.75	64.44	60.98	72.70
5		29.28	32.29	49.29	56.09	47.66	59.60
6		30.00	37.15	57.63	66.39	51.74	59.74
7		30.00	38.86	57.35	69.56	65.88	69.13
8		30.00	40.00	60.01	75.33	88.96	83.51
9		30.00	39.81	57.63	71.71	60.80	80.21
10		30.00	39.70	56.50	66.74	65.48	77.43
11		30.00	39.61	60.00	73.15	64.64	76.86
12		30.00	39.90	58.09	64.39	59.80	65.25
13	26.00	30.00	40.00	58.35	68.15	57.29	73.29
14	26.00	30.00	40.00	58.61	63.81	60.31	79.28
15	26.00	30.00	38.69	49.40	55.13	58.39	60.70
16	26.00	30.00	39.84	55.83	67.14	59.81	72.21
17	26.00	30.00	40.00	55.68	73.64	64.15	74.00
Mean	26.00	29.95	38.87	56.12	66.33	61.67	71.19
St.Dev	0.00	0.19	2.04	3.51	6.08	9.00	7.81

The details of the test can be found in Appendix E. The result shows that there is no difference in the response time when the available time is 30, 40, and 60 seconds. When T_a is short, the entire available time is spent by both organizations because the amount of work needed to be done and the bounded rationality constraint require a processing time that is as long as possible. The difference, although very small, exists at the longer available times, that is, at T_a = 80 and 120 (at T_a = 100 the difference is negligible). However, this small difference highlights the difference in protocols between the two structures. The hierarchical organization tends to use more time to accommodate the interactions between the DMs.

8.3.3 Workload

The workload for each decision maker can be computed if the strategy used during the experiment is known. Tables 8.11 and 12 show the values of the workload for all decision makers when performing the task in both organizations.

Table 8.11 Workload for All DMs in the Hierarchical Organization
(unit : bits)

	Slow operation			Moderate operation			Fast Operation		
team	G1	G2	G3	G1	G2	G3	G1	G2	G3
3	182.1	71.58	135.6	190.2	61.95	144.1	180.6	70.02	135.7
4	181.6	71.63	135.6	189.2	69.48	144.1	180.6	70.14	160.6
5	182.2	71.68	135.6	189.9	62.36	144.1	180.9	70.29	135.9
6	179.9	71.81	135.6	178	64.83	144.1	179.9	70.14	135.7
7	181.7	71.68	135.6	179.3	65.75	144.1	180.6	70.07	135.7
8	182.5	71.68	135.6	190.4	66.04	144.1	180.6	70.36	135.7
9	182.2	71.84	135.6	182.1	70.48	144.1	180.6	70.32	135.7
10	181.7	71.77	155.6	174.3	64.17	144.1	180.3	70.3	156
11	181.9	71.68	160.6	183.4	64.17	144.1	180.6	70.14	160.6
12	180.9	71.63	160.6	178.7	62.36	144.1	180.6	69.98	160.6
13	181.7	71.63	160.6	171.3	63.48	144.1	180.6	70.26	160.6
14	181.5	71.68	160.6	179	61.95	144.1	180.4	70.26	160.6
15	182.5	71.72	135.6	190.4	65.14	144.1	180.6	70.26	135.7
16	181.9	71.63	135.6	178.3	64.83	144.1	180.6	70.36	135.7
17	181.4	71.29	135.6	174.7	65.45	144.1	180.4	70.23	135.7
Mean	181.7	71.66	143.6	181.9	64.83	144.1	180.5	70.21	145.3
St. Dev.	0.67	0.125	11.76	6.592	2.505	3E-06	0.224	0.12	12.32

Table 8.12 Workload for All DMs in the Parallel Organization

Team	G1	G2	G3	G1	G2	G3	G1	G2	G3
3	113	113	113	119.5	119.5	119.5	113	113	113
4	113	113	113	117.7	117.7	117.7	112.7	112.7	112.7
5	113	113	113	118.7	118.7	118.7	112.6	112.6	112.6
6	112.8	112.8	112.8	117.5	117.5	117.5	112.5	112.5	112.5
7	112.9	112.9	112.9	117.8	117.8	117.8	112.4	112.4	112.4
8	112.7	112.7	112.7	118.5	118.5	118.5	112.6	112.6	112.6
9	112.9	112.9	112.9	118.6	118.6	118.6	112.9	112.9	112.9
10	113	113	113	118.3	118.3	118.3	112.6	112.6	112.6
11	112.8	112.8	112.8	118.4	118.4	118.4	112.6	112.6	112.6
12	112.9	112.9	112.9	117.6	117.6	117.6	112.7	112.7	112.7
13	112.8	112.8	112.8	117.6	117.6	117.6	112.7	112.7	112.7
14	113	113	113	118.4	118.4	118.4	112.7	112.7	112.7
15	113	113	113	117.5	117.5	117.5	112.8	112.8	112.8
16	113	113	113	118.2	118.2	118.2	112.3	112.3	112.3
17	113	113	113	118.8	118.8	118.8	112.4	112.4	112.4
Mean	112.9	112.9	112.9	118.2	118.2	118.2	112.6	112.6	112.6
st.dev	0.08	0.08	0.08	0.57	0.57	0.57	0.19	0.19	0.19

8.3.4 Critical Time Ratio

The time ratio is a dimensionless variable which indicates what proportion of the available time is taken to do the task. For a given task, the amount of work needed to be done is a constant if the algorithm used to perform the task is fixed. Because the bounded rationality constraint imposes a maximum processing rate, there exists a minimum time for the completion of the task. When the available time is much longer than the minimum required time, the fraction of T_a used to do the task depends on the processing rate which is generally less than the maximum rate. When the available time is equal or close to the minimum time, the time ratio is equal or close to unity:

$$t = \frac{T_f}{T_a}$$

where T_f is the response time and T_a is the available time. It can assumed that for t near unity, the maximum possible processing rate has been achieved.

The time ratio can be computed from the experiment data. Then, the accuracy J can be plotted versus the time ratio to study its behavior. Figures 8.10 and 8.11 show two such plots for one team, team #13. In the plots, instead of the time ratio, one minus the time ratio ($1-T_f/T_a$) is used. This can be interpreted as the fraction of the available time remaining when the task is completed. Obviously, the larger the available time the larger the remaining time. From these figures, it is observed that accuracy decreases sharply when the time ratio is equal or close to one, that is, when ($1-T_f/T_a$) is equal or close to zero. All teams exhibit similar behavior.

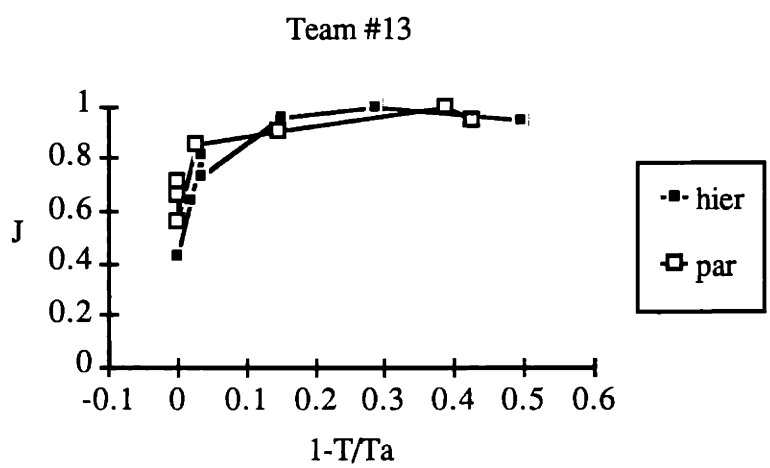


Figure 8.10 Accuracy and Time Ratio

The sudden change in the slope of the curves indicates that there exists a value of the time ratio, denoted by t*, at which the required processing rate is close to the maximum value F_{max}. To find the value of t*, a piece-wise linear fit is performed. Two asymptotes are found. The intersection point of the two asymptotes can be used as an estimate of the t* value.

To find the asymptotes, the Least Square (LS) fit is used. As an example, Fig. 8.11 shows the original curve for the hierarchical organization and the asymptotes found by using LS fit for the same team (#13) used in Fig. 10. Table 8.13 displays t* values for all teams.

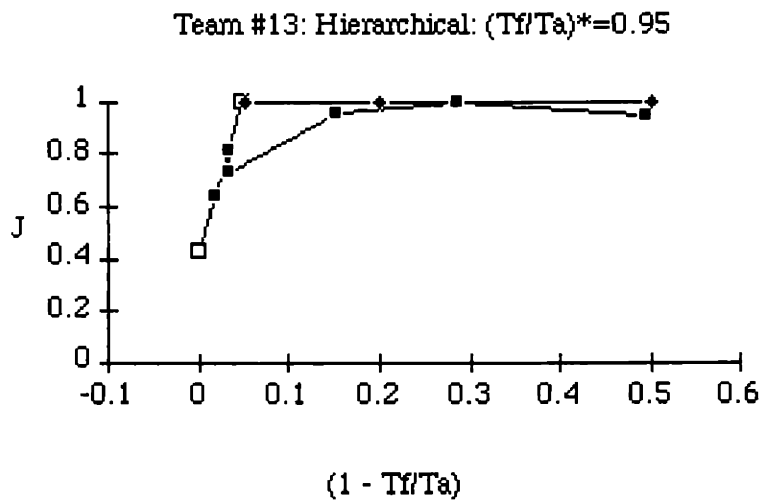


Figure 8.11 Asymptote for J and (1-Tf/Ta)

Table 8.13 Critical Time Ratio
(unit: in seconds)

	Hierarchical	Parallel
team	t*(J)	t*(J)
3	0.93	0.69
4	0.89	0.92
5	0.75	0.68
6	0.84	0.92
7	0.85	0.94
8	0.88	0.87
9	0.92	0.88
10	0.88	0.91
11	0.96	0.97
12	0.88	0.95
13	0.95	0.95
14	0.93	0.95
15	0.92	0.96
16	0.91	0.88
17	0.96	0.89
Mean	0.89	0.89
St. Dev	0.06	0.10

8.3.5 Communication Ratio

The communication ratio, also introduced by dimensional analysis, is defined as follows:

$$n_c = \frac{N_c}{N_{rc}}$$

Interaction between organizational members is an important feature of distributed decision making organizations. The experimental data suggest a relation between performance and interaction through communication.

The required number of communications, shown in Table 8.14, is very different in the two organizations. There are two sets of the values for each organizational structure. There are seven teams which had six T_a values in the experiment while the other eight teams had seven T_a values. The extra value of T_a was added to ensure that sufficient data were collected when the organization was under time pressure.

Table 8.14 Number of Required Communication

	Hierarchical		Parallel	
Ta	Nrc	Nrc	Nrc	Nrc
26	-	-	-	27
30	-	120	26	27
40	123	108	27	29
50	102	111	-	-
60	117	111	27	27
80	117	111	27	28
100	126	111	26	27
120	99	120	30	29

From Table 8.14, it is clear that the required communication level is much higher in the hierarchical organization than in the parallel organization.

Communication Ratio and Organizational Structure

The communication ratio was plotted as a function of the available time for each team. Tables 8.15 and 8.16 show the number of communications and the communication ratio for two teams, while Figs. 8.12 and 8.13 are the plots for these two teams.

Table 8.15 Communication Ratios of Team #13

	Hierarchical			Parallel		
Ta	Nc	Nrc	Nc/Nrc	Nc	Nrc	Nc/Nrc
26	-	-	-	10	24	0.42
30	78	120	0.65	11	26	0.42
40	76	108	0.70	23	29	0.79
50	96	111	0.86	-	-	-
60	100	111	0.90	24	27	0.89
80	108	111	0.97	27	28	0.96
100	105	111	0.95	27	27	1.00
120	111	120	0.93	29	29	1.00

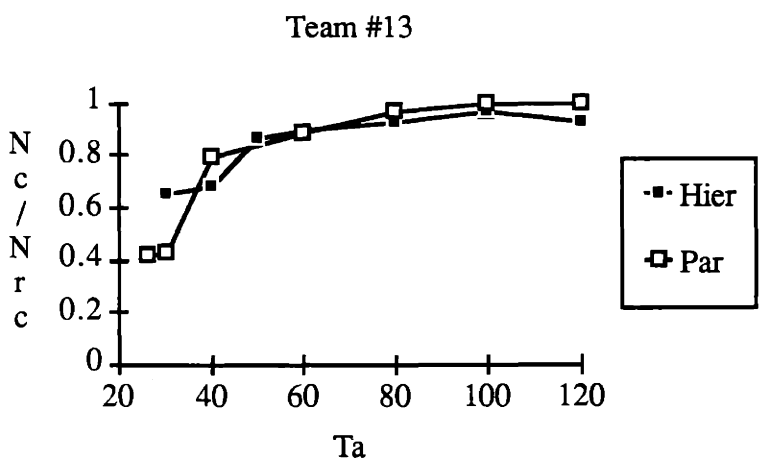


Figure 8.12 Communication Ratio versus Available Time

Table 8.16 Ten Communication Ratios of Team #14

	Hierarchical			Parallel		
Ta	Nc	Nrc	Nc/Nrc	Nc	Nrc	Nc/Nrc
26	-	-	-	14	27	0.52
30	80	120	0.67	13	27	0.48
40	81	108	0.75	20	29	0.69
50	100	111	0.90	-	-	-
60	109	111	0.98	24	27	0.89
80	105	111	0.95	26	28	0.93
100	107	111	0.96	27	27	1.00
120	116	120	0.97	29	29	1.00

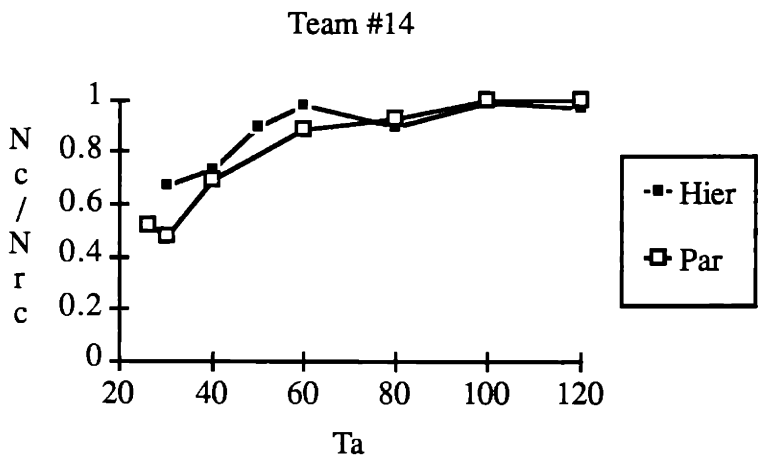


Figure 8.13 Communication Ratio versus Available Time

The following observations can be made.

- The communication ratio decreases as the available time decreases. This is expected.
- There is a larger difference in the communication ratio between two organizational structures when the available time is short than when the available time is long.

- The number of communication is much higher in the hierarchical organization than in the parallel organization.
- However, while the number of communications is very different between the hierarchical and parallel organizations, there is not much difference in the ratio of actual number of communications and the number of task-required communications.

The average values and standard deviations of the communications ratio, n_c, are shown in Table 8.17. Figure 8.14 shows the standard deviation as a function of the available time.

Table 8.17 Average and Standard Deviation of n_c

	Hierarchical		Parallel	
	Average	St. dev	Average	St. dev
Ta	Nc/Nrc	Nc/Nrc	Nc/Nrc	Nc/Nrc
26	-	-	0.64	0.22
30	0.66	0.05	0.72	0.23
40	0.80	0.11	0.87	0.14
50	0.91	0.07	-	-
60	0.93	0.05	0.93	0.08
80	0.96	0.04	0.97	0.04
100	0.98	0.03	0.99	0.02
120	0.98	0.03	1.00	0.01

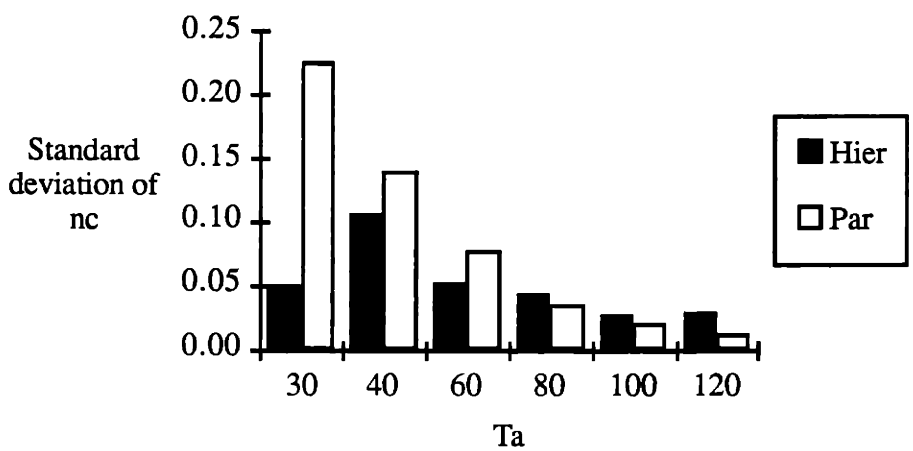


Figure 8.14 Standard Deviation of the Communication Ratio

Figure 8.14 shows clearly that for low values of T_a the variance of the communication ratio is less in the hierarchical organization than in the parallel structure. This reflects the effect of the structure. The hierarchical organization is designed in such a way that interaction is necessary to perform the task. It is natural to expect that a certain amount of communication has to be done regardless of the tempo of the operations. On the other hand, in the parallel organization, the operation is more autonomous than in the hierarchical one so that the interaction can be reduced if the available time is very short.

Next, the relation between the communication ratio and accuracy is studied.

Since both J and n_c are functions of T_a, accuracy versus communication ratio can be plotted. Figure 8.15 shows the plot of the mean values of J and n_c for all 15 teams.

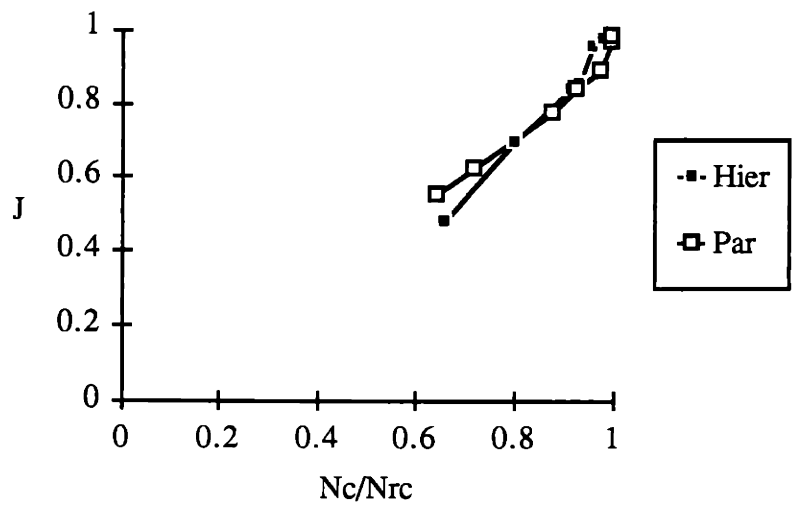


Figure 8.15 Accuracy and Communication Ratio

As a first approximation, a linear model is proposed to express the relation between accuracy J and communication ration n_c. Then, the coefficient of simple determination (Neter, et al, 1977) is computed to check the existence of the linear statistical relation between the two variables. The slope of the linear model indicates how sensitive accuracy is to changes in the communications ratio.

A linear model is assumed as

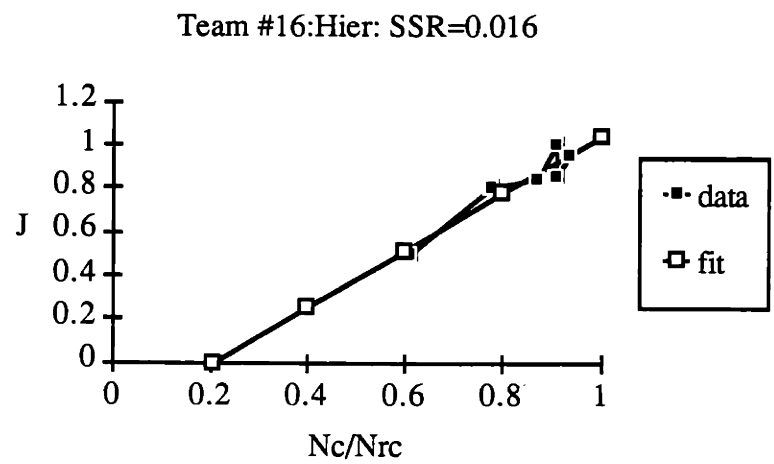
$$J = a + b\, n_c \tag{8.4}$$

where a and b are the parameters of the model. The Least Square method is used to determine a and b The values of a and b across all teams are listed below along with the average value and the standard deviation of the coefficient of simple determination (Table 8.18).

Table 8. 18 Parameters a and b in J and n_c Relation

	Hierarchical		Parallel	
team	a	b	a	b
3	0.02	0.92	0.00	0.80
4	-0.85	1.74	-0.87	1.85
5	-1.18	2.15	0.66	0.28
6	-1.10	2.08	-1.37	2.31
7	-1.34	2.34	-0.38	1.35
8	-0.21	1.17	0.19	0.74
9	-1.05	2.02	-1.04	1.99
10	-0.36	1.37	-0.38	1.33
11	-0.61	1.58	0.01	0.96
12	-1.07	2.05	0.18	0.76
13	-0.51	1.54	0.34	0.59
14	-0.37	1.39	0.35	0.60
15	-0.44	1.48	-1.86	2.82
16	-0.27	1.31	0.31	0.62
17	-0.03	1.02	-2.27	3.15
Mean	-0.62	1.61	-0.41	1.34
Standard Dev.	0.44	0.44	0.88	0.88

Figure 8.16 shows observed points and the model prediction for team #16. The sum of square residual for these curves is 0.016 for the hierarchical organization and 0.025 for the parallel organization.



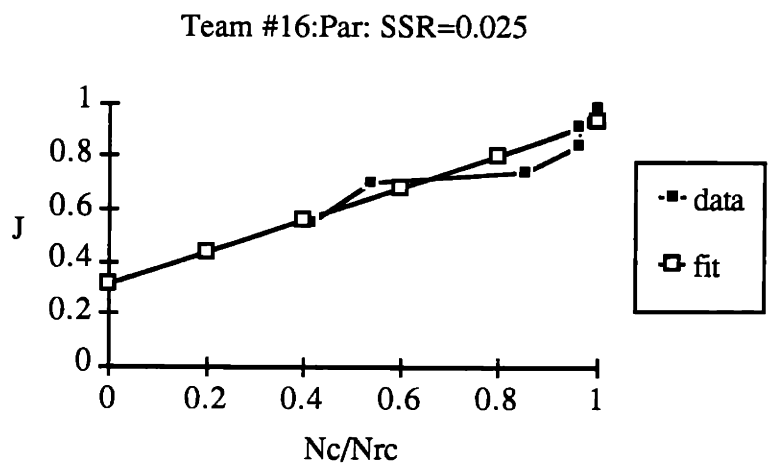


Figure 8.16 Linear Relation Between J and N_c/N_{rc}

The slope of the linear model indicates how sensitive accuracy is to changes in the communications ratio. Since the slope b of the hierarchical organization is higher than that of the parallel organization, the accuracy of the hierarchical organization is more sensitive to changes in the number of communications than that of the parallel organization.

8.4 HYPOTHESIS TESTING

In this section, the two hypotheses introduced in Chapter 5 are tested.

8.4.1 Testing Hypothesis 1.

Hypothesis 1 predicts that the organization with the highest minimum workload will show performance degradation prior to those organizations which have lower minimum workload. From Chapter 5, it is shown that the subordinates in the hierarchical organization have the highest minimum workload.

Let T^*_h and T^*p be the available times at which the performance of the hierarchical organization and the parallel organization degrade sharply. Hypothesis 1 can be expressed as follows.

H0:	$T^*_h > T^*_p$;	Hypothesis 1 is accepted;
H1:	$T^*_h \leq T^*_p$;	Hypothesis 1 is rejected.

To test Hypothesis 1, the mean values of the critical available times, which is the available time at which performance begins to degrade rapidly, for both organizational structures, need to be computed from the experimental data.

Accuracy versus available time (J-T_a plot) are plotted for each team. Figure 8.4 in Section 8.3 is a such plot for one team. The observation from the J-T_a plots indicates that there exists a region in which J starts to degrade rapidly. To estimate the T_a at which such degradation occurs, a piece-wise linear fit is performed. Two asymptotes are found. The intersection point of the two asymptotes can be used to estimate the T* values.

To find the asymptotes, the Least Square (LS) fit is used. As an example, Fig. 8.17 shows the original curves and the asymptotes found by using the LS fit. Table 8.19 displays T* values for all teams.

Although the individual teams show some variation in the behavioral characteristics, the mean value of T_h^* is 62.49 seconds and T_p^* is 58.55 seconds, Hypothesis 1 is confirmed.

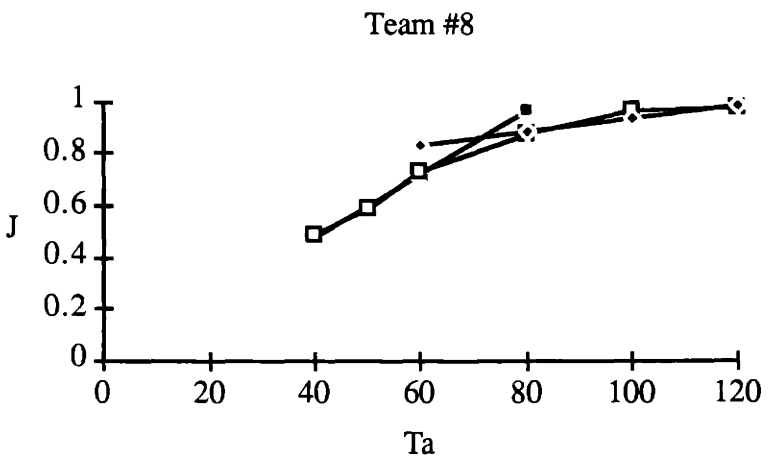


Figure 8.17 Accuracy and Available Time: Hierarchical

Table 8.19 T_a* for Both Organizational Structures

Team	T_h*	T_p*
3	59.82	61.35
4	68.48	50.88
5	51.15	48.08
6	53.18	42.65
7	53.29	49.04
8	82.13	109.89
9	56.73	55.04
10	53.18	47.55
11	76.79	49.17
12	63.02	49.14
13	80.22	69.17
14	61.24	64.23
15	60.35	43.91
16	55.62	60.95
17	62.19	77.27
Mean	62.49	58.55
St. Dev.	10.08	17.28

Ten of the teams verify the hypothesis directly There are also five teams which do not behave in accordance with the hypothesis. The explanation of this discrepancy is as follows. As described in Chapter 5, the workload is allocated differently among the DMs in the hierarchical organization while the workload is the same for all DMs in the parallel organization. During the experiment, a subject played a role in both organizations. Recall from Chapter 5 that in the hierarchical organization, subordinates have the higher workload and the supervisor has the lower workload. Since individual DMs have different skills and capability to process information and make decisions, consider a DM in a team who is a very slow player. His effect on organizational performance will depend on which role he is playing. If a slow player plays the supervisory role in the hierarchical organization, his effect on organizational performance will be very limited since the supervisor has the least workload. However, when this same decision maker plays in the parallel organization, he may reach his maximum processing rate at an earlier stage of decreasing available time. In this case, his performance will significantly affect organizational performance. Therefore, organizational performance of the parallel organization starts to degrade when the processing rate of this DM reaches his maximum value. There are two teams, team 8 and team 17, which have a very slow player playing the supervisor's role in the hierarchical organizations. Table 8.19 shows that these two teams have the most significant discrepancy with the hypothesis.

8.4.2 Testing Hypothesis 2.

If Hypothesis 2 is correct, the rapid reduction in the number of communications and rapid reduction of organizational performance will occur at the same time. Because of the difference in structures and protocols in the hierarchical organization and the parallel organization, the required number of communications is very different. Therefore, it is necessary to take N_{rc} into consideration. The communications ratio is constructed by normalizing the number of communications by the task-required number of communications.

Let t^*_c denote the time ratio at which the number of communications reduces rapidly and t^*_J denote the time ratio when the performance starts to drop significantly. Then Hypothesis 2 can be expressed as

H0:	$t^*_c = t^*_J$;	Hypothesis 2 is accepted;
H1:	$t^*_c \neq t^*_J$;	Hypothesis 2 is rejected.

To test this hypothesis, the relation between the communications ratio and time ratio needs to be analyzed. When the critical available time corresponding to the communications

ratio is determined, it can be compared with the critical time found in Section 8.3.4 to test the hypothesis.

Figure 8.18 is a plot of communication ratio versus the complement of the time ratio for a team. To find the value of t_c*, a piece-wise linear fit is performed. Two asymptotes are found. The intersection point of the two asymptotes can be used to estimate the t_c* value.

To find the asymptotes, the Least Square (LS) fit is used. As an example, Fig. 8.19 shows the original curves and the asymptotes found by using the LS fit for the team in Fig. 8.18 Table 8.20 displays the t_c* values for all teams.

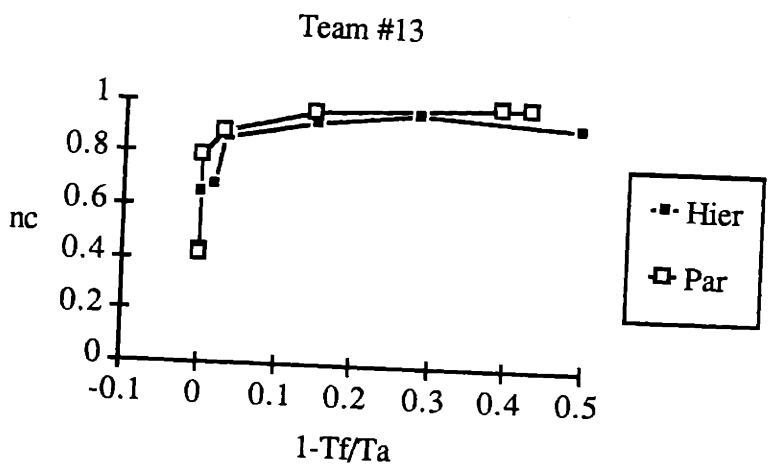


Figure 8.18 Communications Ratio versus Time Ratio

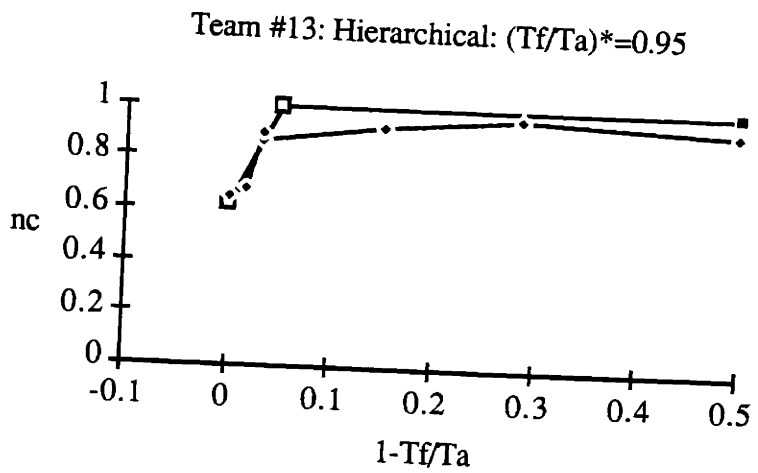


Figure 8.19 Asymptote for n_c and $(1-T_f/T_a)$

Table 8.20 t* Values for J and n_c

	Hierarchical		Parallel	
team	t*(J)	t*(nc)	t*(J)	t*(nc)
3	0.93	0.94	0.69	-
4	0.89	0.94	0.92	0.88
5	0.75	0.80	0.68	0.96
6	0.84	0.87	0.92	0.89
7	0.85	0.87	0.94	0.95
8	0.88	0.91	0.87	0.93
9	0.92	0.93	0.88	0.90
10	0.88	0.74	0.91	0.93
11	0.96	0.98	0.97	0.99
12	0.88	0.88	0.95	0.97
13	0.95	0.95	0.95	0.96
14	0.93	0.92	0.95	0.97
15	0.92	0.90	0.96	0.97
16	0.91	0.91	0.88	0.93
17	0.96	0.96	0.89	0.93
Average	0.89	0.89	0.89	0.94
St. Dev	0.06	0.06	0.10	0.04

In Table 8.20, t*(J) is t_J* while t*(nc) is t_c*. Both of t*'s vary over a narrow range. The standard deviations of t_J* are 0.06 and 0.1 for the hierarchical organization and the parallel organization, respectively, and it is 0.06 and 0.04 for t_c*. One team, #3, never reduced the number of communications when operating as a parallel organization. Therefore, no value for t_c* appears in Table 8.20.

As discussed in Chapter 5, when Hypothesis 2 was formulated, there are two ways to reduce load in the absence of a strategy that permits completion of the task: to reduce the number of communications or to process fewer threats. Clearly, this team chose the second way. Recall that the objective is to process completely all threats. The second way results in

performance degradation at a significantly smaller time ratio, t_J* equal to 0.69. Small t_J* implies that either the team has very fast response time or the degradation of performance occurs at a larger value of the available time. Figures 8.20 and 8.21 show the J-T_a and J-t plots of the parallel organization for this team. From the J-T_a plot, it can be seen that the performance starts to drop significantly at a long available time (100 to 120 seconds). This team continued to coordinate at the cost of responding to threats not in overlap areas. When T_a became very small this organization should showed a second rapid decline in performance.

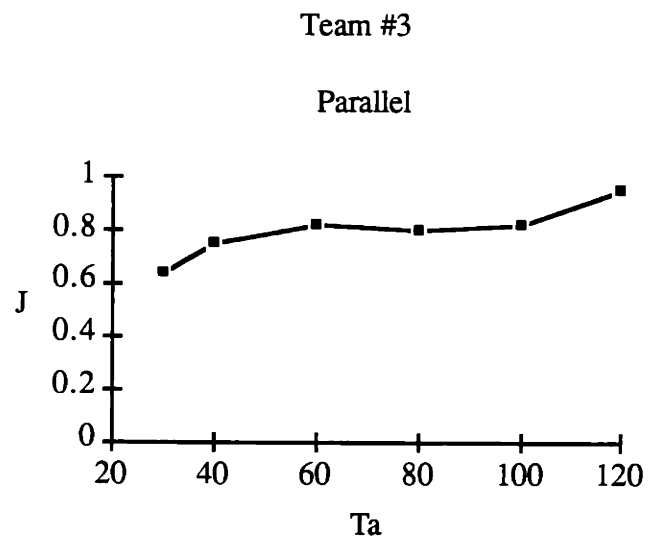


Figure 8.20 Accuracy and Available Time

For the hierarchical organization Table 8.20 shows that

$$\text{mean } t^*(J) = \text{mean } t^*(n_c) = 0.89$$

Therefore, Hypothesis 2 is confirmed. However, for the parallel organization,

$$\text{mean } t^*(J) = 0.89 \text{ and mean } t^*(n_c) = 0.94$$

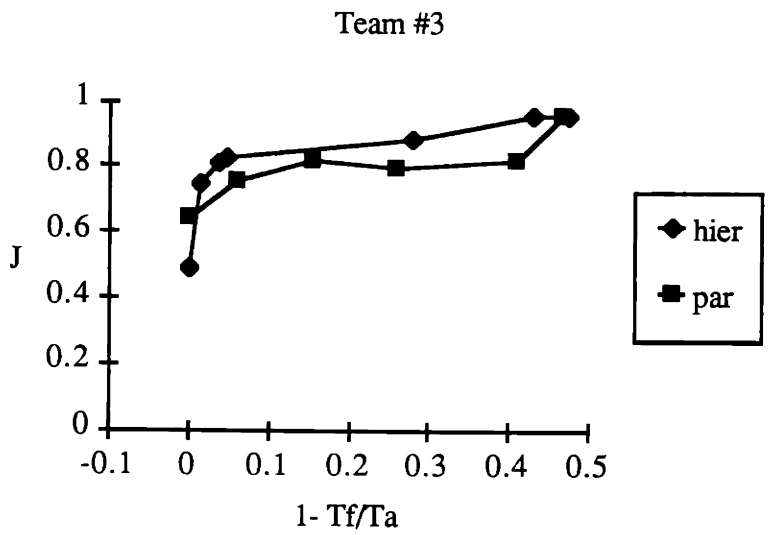


Figure 8.21 Accuracy and Time Ratio

Therefore, Hypothesis 2 is disproved. Since t*(nc) is larger than t*(J), it follows that when the number of communications drops significantly, performance does not yet degrade rapidly. The explanation is the following. When time pressure is very high, the DMs attempt to reduce the number of communications in order to complete the threats in their own sector. However, since the required number of communications in the parallel organization affects only a small portion of the threats that need to be processed, a partial reduction in communications does not affect organizational performance significantly.

8.5 CONCLUSIONS

In this chapter, the experimental results were analyzed and hypotheses were tested. Differences in the behavior of the two organizational structures were established. Although some of the differences are small, they are essential. The main conclusions are as follows:

1) Organizational performance is more predictable than individual performance. The interaction between organizational members compensates for individual differences.

2) The hierarchical organization has less variance in performance than the parallel organization does. Coordination by the supervisor in the hierarchical organization reduces the variance of performance among the teams.

3) For fixed organizational structures, performance requirement cannot be met for all operating conditions. To have robustness in performance over a wide range of conditions, flexible structures are needed.

4) There exists a critical time ratio at which accuracy degrades significantly.

5) The theoretical model can predict organizational performance quite accurately, if the task is well defined and highly structured.

Chapter 9

CONCLUSIONS

This chapter summarizes the results of this thesis. The findings are focussed on the stability of organizational performance for different organization structures. Interactions between organizational members play an important role in performance. The results of the study suggests some future research directions which will be also discussed in this chapter.

9.1 SUMMARY OF RESULTS

The study is focussed on the effect of organizational structure on performance of decision making teams. A design methodology has been introduced to direct the design of model-driven experiments. Since the experiment for investigating performance of a distributed decision making organization involves a large number of parameters and uncertainty in the selection of controlled and the measured variables, controllability and observability are critical to the success of the experimental design. A systematic procedure is necessary for designing experiments. The methodology introduced in this thesis guides the selection of the controlled and the measured variables in the experiment do meet the needs of the problem under investigation.

The methodology was then applied to a multi-person experiment for which hypotheses were generated on the effect of organizational structure on performance. Two different organizational structures were used in the experiment. This application served two purposes: 1) to test and evaluate the methodology for designing model-driven experiments and, 2) to investigate the effect of organizational structure on performance.

The experiment was designed and conducted successfully. The methodology resulted in a feasible, well-controlled experimental design.

There are two aspects which will affect organizational performance. One is the task attributes. Another is the information processing and decision making ability of the organizational members. Task attributes change the operating conditions in which a DDM organization operates. Individual differences of DMs result in a variability of organizational performance. Both aspects were studied. The results are as follows.

The main task attributes that was changes was the available time. When T_a decreases, time pressure is introduced in the organization and DMs have to adjust their processing rate. However, when the processing rate reaches its maximum value, further decrease of the available time causes transition to lower workload strategies until the minimum workload strategy is reached. When no strategy is available to reduce the workload in order to accommodate a shorter available time, rapid degradation in performance occurs. The experimental results confirm a hypothesis which predicts that with decreasing available time, a significant degradation of performance occurs first in the organization which has the highest minimum workload.

When individual performance and team performance are compared, the result shows that organizational performance is more predictable than individual performance. The reason is that the interaction among DMs compensates for differences in individual performance characteristics.

The time ratio introduced by dimensional analysis provides useful information on the determination of the available time organization design. The critical time ratio implies the shortest available time for doing a task. This ratio, together with information from the time calibration, can be used to specify the range of available time for a given task in a new design.

9.2 FUTURE RESEARCH DIRECTIONS

Make Full Use of Dimensional Analysis

In this thesis, dimensional analysis was introduced to include cognitive aspects of distributed decision making. The controlled and the measured variables were determined by applying dimensional analysis. A set of dimensionless groups were formed, which determined organizational performance and characterized organizational behavior. Only two such dimensionless groups were used during this study, namely, the time ratio and the communication ratio. There are others that may be used in the future to study organizational behavior.

For example, in Chapter 6, another dimensionless group derived is

$$\frac{G^i T_a}{H_{T_p^i}}$$

This group of variables can be expressed as the ratio of the actual processing rate and the average input uncertainty rate during a trial. In order to see how this dimensionless parameter

relates to performance, control on G^i is necessary. Such control can be realized by carefully designing the procedures and protocols for the experiment.

Determine Available Time

The time ratio was analyzed while testing a hypothesis. The critical value of the time ratio, t*, provides useful information for the experimental design.
The critical value of the time ratio found in the experiment has been embedded in it the characteristics of bounded rationality. Therefore, it can be used as a reference in new experimental designs. The critical time ratio found in this experiment does not depend on the task. The critical time ratio is

$$t^* = \frac{T_f^*}{T_a^*} \tag{9.1}$$

Therefore, a small scale pilot experiment can be used to determine T^*_f. Then, equation (9.1) can be used to determine the range of the available time. The minimum available time T^*a is obtained by

$$T_a^* = t^* \, T_f^*$$

The relation between T_f and Ta is such that T_f reaches a constant value at some T_a , that is, T_f does not increase with further increase of T_a. The maximum T_a is determined at the time T_f becomes independent of T_a. The minimum and maximum T_a establish a range fit this controllable variables.

Free communication vs. Mandatory Communication

The number of communications among decision makers was studied. It was shown that performance relates to the communication ratio in a linear fashion. The communications in this experiment were mandatory. The results indicate that when the available time is so short that even for the minimum workload DM cannot complete the task, a reduction in required communications will occur. Is it true that reduction of communications may occur earlier if communication is voluntary?

Interaction and Organizational Performance

Interaction between organizational members is an important part of distributed decision making organizations. This study has provided some insights on how interaction affects performance. Since the experiment designed in this study is the first attempt to use the theoretical model and the evaluation procedure for a model-driven experiment, the task and the procedure are highly structured and restricted to a specific class of tasks. Interaction was characterized by the number of communications. Although the study was a much simplified version of reality, the results capture the features of interacting DMs. However, in order to understand better organizational behavior, a more general study on the relation between interactions and organizational performance is desired.

A more precise definition of interaction is required. A more complex task should be used to simulate a decision making environment to obtain more information about organizational behaviors. Some restrictions in this experiment can be relaxed to allow more interaction and coordination. But, it is necessary to emphasize that the experiment must be controllable.

Coordination

When interaction describes the activities between organizational members, coordination measures how these activities fit in the operating sequence. Information has time value in decision making organizations. Therefore, how long a process has to wait for needed information to arrive is an important measure. On the other hand, the information content also needs to be considered. When an item of information arrives at a process, its content has to be checked to determine whether it is the information needed.Consistency of information is necessary.

In the study of this thesis, coordination is not an objective. However, the findings about the relation between interaction and performance point to a need for a fundamental study of coordination. The experimental results suggest that interaction affects performance by reducing variance of accuracy and increasing the robustness of the organization performance. But more detailed study on when the interaction occurs and what information is exchanged during the interaction is essential to establish a causal relation between interaction and performance.

REFERENCES

Andreadakis, S. K. (1988). "Analysis and Synthesis of Decision-Making Organizations." LIDS-TH-1740. PhD Thesis, Laboratory for Information and Decision Systems, MIT, Cambridge, MA..

Boettcher, K. L. (1981). "An Information Theoretic Model of the Decision Maker." LIDS-TH-1096. S.M. Thesis, Laboratory for Information and Decision Systems, MIT, Cambridge, MA,.

Boettcher, K. L. and A. H. Levis (1982). "Modeling the Interacting Decisionmaker with Bounded Rationality," *IEEE Trans. on Systems, Man, and Cybernetics,* Vol. SMC-12, No. 3.

Boettcher, K. L. and A. H. Levis (1983). "Modeling and Analysis of Teams of Interacting Decisionmakers with Bounded Rationality," *Automatica,* Vol. 19, No. 6.

Chyen, G. H.-L. (1984). "Information Theoretic Models of Preprocessors and Decision Aids." LIDS-TH-1377. S.M. Thesis, Laboratory for Information and Decision Systems, MIT, Cambridge, MA,

Charnes, A. and W. W. Cooper (1963). "Deterministic Equivalents for Optomizing and Satisficing Under Chance Constraints." *Operations Research.* Vol. 11.

Cohen, M. D., J. G. March, and J. P. Olsen (1972). "A Garbage Can Model of Organizational Chioce." *Administrative Science Quarterly*, March 1972.

Conant, R. C. (1976). "Laws of Information which Govern Systems," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-6, No. 4.

Eihorn, H. J., and R. M. Hogarth (1981). "Behavioral Decision Theory: Processes of Judgement and Choice." *Annual Review of Psychology*, No.32.

Gerhart, P. M. and R. J. Gross (1985). *Fundamentals of Fluid Mechanics*. Addison-Wesley, Reading, M.A.

Greitzer, F. L. and R. L. Hershman (1984). "Modeling Decision Making Strategies in Simulated AAW Operations," *Proc. 7th MOT/ONR Workshop on C3 Systems*. LIDS-R-1437, MIT, Cambridge, MA.

Hall, S. A. (1982). "Information Theoretic Models of Preprocessors and Decision Aids," LIDS-RTH-1232. S.M. Thesis, Laboratory for Information and Decision Systems, MIT, Cambridge.

Hunsaker, J. D. and B. G. Rightmire (1947). *Engineering Applications of Fluid Mechanics*. McGraw-Hill, New York.

Levis, A. H. (1984), "Information Processing and Decision-making Organizations: A Mathematical Description," *Large Scale Systems*, Vol. 7.

Levis, A. H. (1990). "Distributed Intelligent Systems." Class notes for Course 6.291 in Department of Electrical Engineering and Computer Science, MIT.

Levis, A. H. and K. L. Boettcher (1983). "Decisionmaking Organizations with Acyclical Information Structures," *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. SMC-13, No.3.

Louvet, A. C. (1988). "The Bounded Rationality Constraint: Experimental and Analytical Results," LIDS-TH-1771. S.M. Thesis, Laboratory for Information and Decision Systems, MIT, Cambridge, MA.

Louvet, A. C., J. T. Casey, and A. H. Levis (1988). "Experimental Investigation of Bounded Rationality Constraint," in *Science of Command and Control*, S. E. Johnson and A. H. Levis, Eds. AFCEA International Press, Washington, DC.

March, J. G. (1978). "Bounded Rationality, Ambiguity, and the Engineering of Choice," *Bell J. Economics*, Vol. 9.

March, J. G. and H, A. Simon (1958). *Organizations*.John Wiley and Sons, New York.

McGill, V. J. (1954). "Multivariate Information Transmission," *Psychmetrika*, Vol. 19, No.2

Miller, J.G. (1969). "Adjusting to Overlays of Information." In *Organization: Systems, Control and Adaptation.*, Vol. 2, J.A. Littener, Ed. John Wiley and Sons, New York.

Neter, J., W. Wasserman, and G. A. Whitemore (1978). *Applied Statistics*. Allyn and Bacon, Inc., Boston, London, Sydney, Toronto.

Owen, G. (1968). *Game Theory*. W. B. Saunders, Philadelphia, PA.

Shannon, C. E, and W. Weaver (1949). *The Mathematical Theory of Communication.* University of Illinois Press, Urbana, Illinois.

Sheridan, T. B and W. R. Ferrell (1974). *Man-Machine Systems: Information, Control, and Decision Model of Human Performance*. The MIT Press, Cambridge, MA.

Simon, H.A. (1956). "Rational Choice and the Structure of the Environment", *Psychological Review*, Vol. 63.

Wohl, J. G. (1981). "Force Management Decision Requirements for Air Force Tactical Command and Control," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-11, No.9.

APPENDIX A. INPUT ATTRIBUTES AND ENTROPY

Input attributes are the speed of the threats, V, the number of threats in an observation area, N, and number of aircraft in a threat, m. The values of these attributes are as follows.

For the hierarchical organization:

$V = \{ 300, 360, 450, 600, 720, 900, 120 \}$ in miles per hour
$N = \{ 5, 6, 7 \}$
$m = \{ 2, 3, 4, 5, 6 \}$

For the parallel organization:

$V = \{ 300, 360, 450, 600, 900, 120, 1400 \}$ in miles per hour
$N = \{ 3, 4, 5 \}$
$m = \{ 2, 3, 4, 5, 6 \}$.

The number of threats in an *observation area* is determined by the bearing of each threat. The number of threats in *each sector* is constant: 4 for the hierarchical organization and 3 for the parallel organization. There are two sectors in the hierarchical organization while three sectors in the parallel organization. The total number of threats in the defending area is 8 for the hierarchical and 9 for the parallel organization.

However, depending on the bearing, a threat may be in an overlap area so that it can be also observed by an adjacent DM. For example, in Fig. A.1, the area without gray shading is an observation area for a DM, say DM1. Two solid lines in the observation area are the boundaries of sector 1 in which DM1 has the responsibility for intercepting the threats. There are 3 threats in sector 1 while there are 4 threats in the entire observation area. There is one threat in the overlap area and in other sector. Therefore, the number of threats that DM1 has to process is four.

To avoid threats overlapping each other, a set of angles are selected as the candidates for possible bearings:

For the hierarchical organization: ($0 \leq \theta < 180$ for a sector)

$\theta = \{ 5, 25, 75, 95, 115, 135, 155, 175 \}$ in degrees;

For the parallel organization: ($0 \leq \theta < 120$ for a sector)

$\theta = \{ 10, 25, 40, 55, 70, 85, 100, 115 \}$ in degrees.

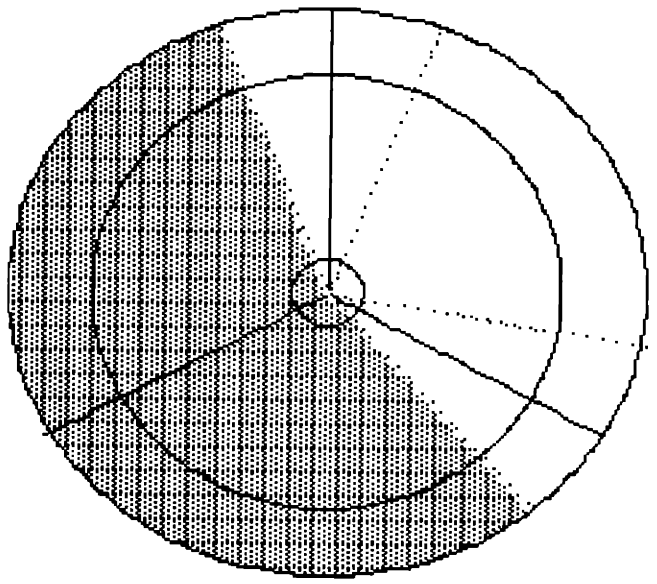


Figure A.1 Defense Area, Observation Area, and Sector

These two sets are for one sector. 180 degrees are added to each value of θ for the second sector in the hierarchical organization, while 120 or 240 degrees are added to each value of θ for the second and the third sectors in the parallel organization, respectively. The size of the overlap area is 30 degrees for the hierarchical and 20 degrees for the parallel organization.

There are some restrictions on the bearings. First, no angle can be selected more than once for a sector. Furthermore, at most one threat can be in an overlap area in the parallel organization. For the hierarchical organization, the maximum number of threats allowed in an overlap area is two. These restrictions eliminate the possibility that a threat overlaps another.

Given these restrictions, all admissible combinations of angles for the two structures are constructed. First, all combinations of possible angles in θ are generated. Then, only those that satisfy the restrictions are stored as candidates for the selection of bearings. There are 120 admissible sets of bearings for the hierarchical organization and 36 for the parallel organization. Table A.1 and A.2 show a part of the admissible sets for each structure. These sets of angles can be divided into three groups:

For the hierarchical organization:

1) there is one bearing falling in an overlap area; => N = 5 (4 + 1);

2) there are two bearings falling in overlap area(s); => N = 6 (4 + 2);
3) there are three bearings falling in overlap areas; => N = 7 (4 + 3).

For the parallel organization:

1) there is no bearing falling in overlap area; => N = 3 (3+ 0);
2) there is one bearing falling in overlap area; => N = 4 (3+ 1);
3) there are two bearings falling in overlap areas; =< N = 5 (3+ 2).

Table A.1 Admissible Bearings for Threats: Hierarchical
(unit: degree)

5	25	55	75
5	25	55	95
5	25	55	115
5	25	55	135
5	25	55	155
5	25	55	175
5	25	75	95
5	25	75	115
5	25	75	135
5	25	75	155
...	...	...	...
95	115	135	175
95	115	155	175
95	135	155	175
115	135	155	175

The required number of communications depends on the threats' bearing. For the input files generated, the required number of communications for each available time is shown in Table A.3. There are two sets of the values for each organizational structure. There are seven teams which have six T_a values in the experiment while the other eight teams had seven T_a values. One T_a value was added to ensure that sufficient data were collected on the behavior of the organization under time pressure. The teams running with six Ta conditions have the same data file; the teams running with seven T_a conditions have the same data file.

Table A.2 Admissible Bearings for Threats: Parallel

(unit: degree)

10	25	40
10	25	55
10	25	70
10	25	85
10	25	100
10	25	115
10	40	55
10	40	70
10	40	85
10	40	100
...	...	...
55	100	115
70	85	115
70	100	115
85	100	115

Table A.3 Required Number of Communications

Hierarchical Organization			Parallel Organization		
Ta	Nrc(6)	Nrc(7)	Ta	Nrc(6)	Nrc(7)
30	-	120	26	-	27
40	123	108	30	26	27
50	102	111	40	27	29
60	117	111	60	27	27
80	117	111	80	27	28
100	126	111	100	26	27
120	99	120	120	30	29

To generate the bearings for the threats, a number of sets are selected independently from all admissible sets.

Hierarchical organization

Since there are two sectors, two sets of bearings are selected independently, one for each sector. Since the number of threats in a sector is constant, denoted by n, the number of threats

in an observation area is equal to n plus the number of threats in the overlap areas with the adjacent sectors. Therefore, the probability of N is equal to the probability of the number of threats in the overlap areas with the adjacent sector. Table A.4 shows a number of sets which result in different values of N.

Table A.4 Number of Sets for Each N

Case	Number of Sets	N
one in overlap area	40	5
two in overlap area	59	6
three in overlap area	21	7
total	120	

The number of threats the supervisor can handle depends on how many threats are in the overlap areas in both sectors. Table A.5 shows the probability of N for the subordinates and the supervisor.

Table A.5 Probability of the Number of Threats: Hierarchical

(a) Subordinate

N	P(N)
5	0.333
6	0.492
7	0.175

(b) Supervisor

N	P(N)
2	0.125
3	0.375
4	0.281
5	0.188
6	0.031

Parallel Organization

There are three sectors in this organization. Three sets of bearings need to be selected. In this organization, the situation is more complicated than in the hierarchical organization. When a threat is in an overlap area, it can be in either one of the adjacent sectors. To show the computation clearly, let us assume a DM is standing at the center of the defending area facing his sector. There are two overlap areas in his observation area, one is on his left, denoted by L, and another is on his right, R. If a selected set has one threat in the overlap area and on the left, it is represented as 1L. 1R indicates that the selected set has one threat in the overlap area on his right. Two threats in the overlap areas, one on the left (1L) and one on the right (1R), are indicated by 2. Therefore, there are three possible cases.

Since there are three sectors, there are 27 possible outcomes when selecting bearings for all sectors. Each of the outcomes will result in 3 or 4 or 5 threats in the observation areas. The selection of the bearing for a threat is independent of the selections of the others. Table A.6 shows the probability of N.

Table A.6 Probability of the Number of Threats: Parallel

N	P(N)
3	0.174
4	0.484
5	0.340

The number of aircraft in a threat is m. The probability of m is uniform, that is,

$$p(m) = \frac{1}{5} = 0.2$$

Therefore, the probability of input x is

$$p(x) = P(N)\, p(m)^N$$

Since there are N threats, N m's need to be generated. The generation of the number of aircraft for each threat is independent of each other. The entropy of input is

$$H(x) = -\sum_{x} p(x) \log(p(x)) = -\sum_{N,\, m} p(N)p(m)^N \log[p(N)p(m)^N]$$

Table A.7 shows the entropy of input for both organizations.

Table A.7 Entropy of Input

		H(x) in bits
Hierarchical	Subordinates	10.297
	Supervisor	9.101
Parallel	All DMs	8.307

APPENDIX B COMPUTATION OF TASK WORKLOAD

Task workload is measured by total activity, G, during the execution of the task. Recall from Chapter 3 that the total activity G is

$$G = \sum_i H(w_i) \tag{B.1}$$

where H(w) is the marginal entropy of a system variable w.

To compute total activity, entropies of all system variables need to be computed. Entropy is computed by

$$H(w) = -\sum_i p(w_i) \log[p(w_i)] \tag{B.2}$$

where $p(w_i)$ is the probability that system variable w takes the value w_i.

Therefore, there are three steps in computing task workload: 1) derive probability for each of the system variables; 2) compute the entropy of each system variable; 3) add up the entropies for all system variables. In Chapter 5, flowcharts for all information processing stages are shown and described. The probability of each system variable is derived according to these flowcharts.

B.1 COMPUTING TOTAL ACTIVITY IN PREPROCESSING STAGE

Figure B.1 shows the flowchart for the preprocessing (PP) stage. The system variables are denoted by w.

Probabilities of these system variables are as follows.

$$p(w1) = p(V)p(\theta)$$

For each given V, p(V) = 1.0 since the available time is deterministic. Then,

$$p(w1) = p(\theta)$$

For the hierarchical organization,

$$p(\theta) = \frac{1}{120}$$

Therefore, since θ is uniformly distributed, the entropy of w1 is

$$H(w1) = -\sum_{\theta} p(\theta)\log[p(\theta)] = -120\left[\frac{1}{120}\log_2\left(\frac{1}{120}\right)\right] = 6.91 \text{ bits}$$

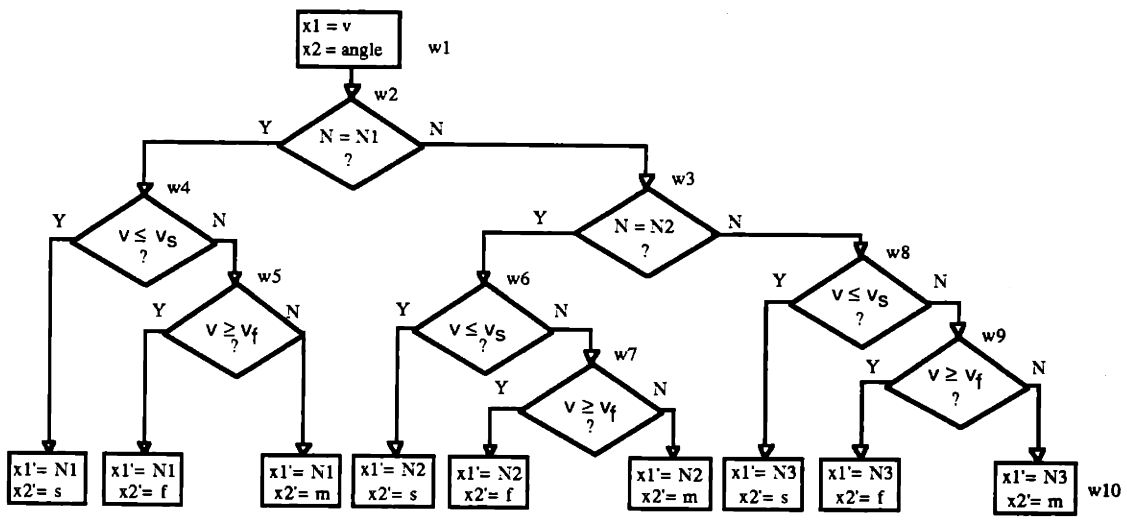


Figure B.1 Flowchart for Preprocessing Stage

For the parallel organization,

$$p(\theta) = \frac{1}{36}$$

Then, the entropy of w1 is

$$H(w1) = -\sum_{\theta} p(\theta)\log[p(\theta)] = -36\left[\frac{1}{36}\log_2\left(\frac{1}{36}\right)\right] = 5.17 \text{ bits}$$

For w2 and w3, p(w2) = p(N=N1) and p(w3) = p(N=N2). N1 and N2 are 5 and 6 for the hierarchical organization and 3 and 4 for the parallel organization. The entropies of w2 and w3 for each organization are computed as follows.

Hierarchical Organization:

From Table A.4, P(N = 5) = 0.33 and P(N = 6) = 0.492, then the entropy is

$$H(w_2) = - \{ p(N = 5) \log_2 [p(N = 5)] + [1 - p(N = 5)] \log_2 [1 - p(N = 5)] \}$$
$$= - [0.33 \log_2 (0.33) + (1 - 0.33) \log_2 (1 - 0.33)] = 0.918 \text{ bits}$$

$$H(w_3) = - \{ p(N = 6) \log_2 [p(N = 6)] + [1 - p(N = 6)] \log_2 [1 - p(N = 6)] \}$$
$$= - [0.492 \log_2 (0.492) + (1 - 0.492) \log_2 (1 - 0.492)] = 1.0 \text{ bits}$$

Parallel organization:

From Table A.5, P(N = 3) = 0.174 and P(N = 4) = 0.484, then the entropies are

$$H(w_2) = - \{ p(N = 3) \log_2 [p(N = 3)] + [1 - p(N = 3)] \log_2 [1 - p(N = 3)] \}$$
$$= - [0.174 \log_2 (0.174) + (1 - 0.174) \log_2 (1 - 0.174)] = 0.665 \text{ bits}$$

$$H(w_3) = - \{ p(N = 4) \log_2 [p(N = 4)] + [1 - p(N = 4)] \log_2 [1 - p(N = 4)] \}$$
$$= - [0.484 \log_2 (0.484) + (1 - 0.484) \log_2 (1 - 0.484)] = 0.999 \text{ bits}$$

The system variables w4 to w9 depend on the speed of threats. Because the speed is deterministic, uncertainty of these variables is zero, that is,

$$H(w_4) = H(w_5) = H(w_6) = H(w_7) = H(w_8) = H(w_9) = 0.$$

System variable w10 takes values from the pair of number of threats, N, and class of the threats: P(w10) = p(N, class). Table B.1 shows all possible combinations of N and class.

Again, the class is deterministic, p(w10) = p(N). The entropy of w10 is

$$H(w_{10}) = - \sum_N p(N) \log_2 [p(N)]$$

For the hierarchical organization:

$$H(w_{10}) = - [0.33 \log_2(0.33) + 0.492 \log_2(0.492) + 0.175 \log_2(0.175)]$$
$$= 1.472 \text{ bits}$$

for the parallel organization:

$$H(w_{10}) = -\ [0.174 \log_2(0.174) + 0.484 \log_2(0.486) + 0.341 \log_2(0.340)]$$
$$= 1.473 \text{ bits}$$

Table B.1 Number of Threats and Class of Threats

Hierarchical		Parallel	
N	**Class**	**N**	**Class**
5	slow	3	slow
6	slow	4	slow
7	slow	5	slow
5	medium	3	medium
6	medium	4	medium
7	medium	5	medium
5	fast	3	fast
6	fast	4	fast
7	fast	5	fast

All system variables in the PP stage are computed. The entropy of the PP stage is obtained by summing all the entropies of the system variables involved in the PP stage.

$$H(PP) = -\sum_{i=1}^{10} H(w_i)$$

For the hierarchical organization:

$$H(PP) = 6.91 + 0.918 + 0.999 + 1.472 = 10.297 \text{ bits}$$

For the parallel organization:

$$H(PP) = 5.17 + 0.665 + 0.999 + 1.473 = 8.307 \text{ bits}$$

The computation of the entropies for all other system variables are the same as described above.

B.2 ALGORITHMS IN THE SA AND RS STAGES

In this section, the algorithms that can be used in the SA and RS stages and the internal variables for these algorithms will be described.

B.2.1 Estimation Algorithm for the SA Stage

This algorithm is used in f2 in the SA stage.

Input: x' = { class of a threat, number of aircraft in the threat m }

output: z = { type of the threat }

Internal variables:

w1 = { class = slow }
w2 = { class = fast }
w3 = { m < 5 }
w4 = { m > 5 }
w5 = { m < 5 }
w6 = { m > 5 }
w7 = { m < 5 }
w8 = { m > 5 }

Let B denote bomber, F denote fighter, and S represent surveillance aircraft. The algorithm works as follows.

```
if class = slow then
    if m < 5 then
        z = S
    else if m > 5 then
        z = B
    else
        z = S or B with probability of one half for each outcome
if class = fast then
    if m < 5 then
        z = F
    else if m > 5 then
```

```
            z = B
        else
            z = F or B with probability of one half for each outcome
    if class ≠ slow and class ≠ fast then { class is medium }
        if  m < 5 then
            z = S or F  with probability of one half for each outcome
        else if m > 5 then
            z = B
        else
            z = B or F or S with probability of one third for each outcome.
```

This algorithm is not deterministic.

B.2.2 Estimation Algorithm in the Resource Allocation Stage

Input: x = { type of a threat z, number of aircraft in the threat m }
Output: y = { resource allocation to the threat }
Internal variables:

$$w1 = \{z = F, m = 2\}$$
$$w2 = \{z = F, m = 3\}$$
$$w3 = \{z = F, m = 4\}$$
$$w4 = \{z = F, m = 5\}$$
$$w5 = \{z = B, m = 5\}$$
$$w6 = \{z = B, m = 6\}$$
$$w7 = \{z = S, m = 2\}$$
$$w8 = \{z = S, m = 3\}$$
$$w9 = \{z = S, m = 4\}$$

The algorithm works as follows.

```
if z = F and m = 2 then
    y1: 2r1 + r2 = 2
if z = F and m = 3 then
    y2: 2r1 + r2 = 3
if z = F and m = 4 then
    y3: 2r1 + r2 = 4
```

if z = F and m = 5 then

$y4: 2r_1 + r_2 = 5$

if z = B and m = 5 then

$y4: 3r_1 + 2r_2 + r_3 = 5$

if z = B and m = 6 then

$y4: 3r_1 + 2r_2 + r_3 = 6$

if z = S and m = 2 then

$y1: 4r_1 + 3r_2 + r_3 = 2$

if z = S and m = 3 then

$y2: 4r_1 + 3r_2 + r_3 = 3$

if z = S and m = 4 then

$y3: 4r_1 + 3r_2 + r_3 = 4$

if z = S and m = 5 then

$y4: 4r_1 + 3r_2 + r_3 = 5$

where the yi 's represent the resource allocation vectors as described in Chapter 5, i.e.,

$$yi = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}$$

B.2.3 Probing Algorithm in the SA and RS Stages

For the probing algorithms in both SA and RS stages, the input and the output are the same as for the estimation algorithms. No internal decision is made. The result is produced by the computer.

The alternative algorithms in the SA and RS stages allow different strategies to be used during the simulation and the experiment.

B.3 TOTAL ACTIVITY IN EACH STAGE

This section summarizes activities for all stages. Since the situation assessment (SA) stage is divided into three functions: f1, f2, and f3, which are explained in Chapter 5, the entropy for each function is shown separately. Tables B.2 and B.3 summarizes the results of the computation.

The total activity, or the task workload G, is computed by summing all activities in all stages. Tables B.4 and B.5 show the task workload for all pure strategies.

Table B.2 Entropies for Different Stages (unit: bits)

Hierarchical			Parallel	
Stage	H(stage) Subordinates	H(stage) Supervisor	H(stage) slow/fast	H(stage) medium
PP	10.297	-	8.307	-
SA f1	11.553	-	9.837	-
SA f3	2.997	-	2.473	-
IF	-	1	3.693	4.381
CI	3.793	-	-	-
POP	-	2	-	-

Table B.3 Entropies for SA Stage (unit: bits)

	Estimation			Probing		
	Slow	Medium	Fast	Slow	Medium	Fast
SA - f2	8.282	8.971	8.282	3.203	3.892	3.203
RS	12.028	13.631	12.497	5.511	6.774	5.502

Table B.4 Workload for Pure Strategies (unit: bits)

	Subordinate			Supervisor		
Workload	Slow	Medium	Fast	Slow	Medium	Fast
G(D1)	200.95	210.75	202.83	90.00	98.31	91.70
G(D2)	174.88	183.32	174.85	-	-	-
G(D3)	175.96	185.76	177.84	71.59	79.90	73.29
G(D4)	149.90	158.34	149.86	-	-	-

Table B.5 Workload for Pure Strategies for Parallel Organization

Workload	Slow	Medium	Fast
G(D1)	138.23	147.42	139.64
G(D2)	118.68	126.85	118.66
G(D3)	121.22	130.40	122.63
G(D4)	101.67	109.83	101.64

B.4 WORKLOAD FOR PURE STRATEGIES

The number of pure strategies of an organization is computed by

$$K = \prod_{i=1}^{n} k^i$$

where k^i is the number of pure strategies of the i-the decision maker and n is number of decision makers in the organization. Therefore, there are 32 pure strategies for the hierarchical organization and 64 pure strategies for the parallel organization.

Tables B.6 and B.7 show the pure strategies and the accuracy and workload associated with them. In these tables, workload g is in bits. The indices under DMs are the indices of pure strategies. There are three groups of performance and workload values, each corresponding to a class of threats.

Table B.6 Pure Strategies For the Hierarchical Organization

DMs						Slow			Medium			Fast	
1	2	3	Js	Jm	Jf	g1	g2	g3	g1	g2	g3	g1	g2	g3
1	1	1	1.00	1.00	1.00	135.6	52.0	135.6	144.1	57.6	144.1	135.6	53.7	135.6
1	1	3	0.87	0.89	0.73	135.6	52.0	155.6	144.1	57.6	165.4	135.6	53.7	157.5
1	1	2	1.00	0.89	0.98	135.6	52.0	160.6	144.1	57.6	169.1	135.6	53.7	160.6
1	1	4	0.90	0.85	0.74	135.6	52.0	180.6	144.1	57.6	190.4	135.6	53.7	182.5
1	2	1	0.99	0.96	0.99	135.6	70.4	135.6	144.1	76.1	144.1	135.6	72.1	135.6
1	2	3	0.93	0.87	0.73	135.6	70.4	155.6	144.1	76.1	165.4	135.6	72.1	157.5
1	2	2	1.00	0.88	0.97	135.6	70.4	160.6	144.1	76.1	169.1	135.6	72.1	160.6
1	2	4	0.88	0.79	0.70	135.6	70.4	180.6	144.1	76.1	190.4	135.6	72.1	182.5
2	1	1	0.89	0.88	0.74	155.6	52.0	135.6	165.4	57.6	144.1	157.5	53.7	135.6
2	1	3	0.77	0.75	0.47	155.6	52.0	155.6	165.4	57.6	165.4	157.5	53.7	157.5
2	1	2	0.87	0.78	0.71	155.6	52.0	160.6	165.4	57.6	169.1	157.5	53.7	160.6
2	1	4	0.77	0.73	0.46	155.6	52.0	180.6	165.4	57.6	190.4	157.5	53.7	182.5
2	2	1	0.90	0.86	0.75	155.6	70.4	135.6	165.4	76.1	144.1	157.5	72.1	135.6
2	2	3	0.81	0.78	0.47	155.6	70.4	155.6	165.4	76.1	165.4	157.5	72.1	157.5
2	2	2	0.87	0.77	0.74	155.6	70.4	160.6	165.4	76.1	169.1	157.5	72.1	160.6
2	2	4	0.77	0.72	0.50	155.6	70.4	180.6	165.4	76.1	190.4	157.5	72.1	182.5
3	1	1	0.99	0.91	0.99	160.6	52.0	135.6	169.1	57.6	144.1	160.6	53.7	135.6
3	1	3	0.91	0.82	0.72	160.6	52.0	155.6	169.1	57.6	165.4	160.6	53.7	157.5
3	1	2	0.99	0.82	0.96	160.6	52.0	160.6	169.1	57.6	169.1	160.6	53.7	160.6
3	1	4	0.89	0.74	0.71	160.6	52.0	180.6	169.1	57.6	190.4	160.6	53.7	182.5
3	2	1	0.99	0.88	0.97	160.6	70.4	135.6	169.1	76.1	144.1	160.6	72.1	135.6
3	2	3	0.88	0.78	0.72	160.6	70.4	155.6	169.1	76.1	165.4	160.6	72.1	157.5
3	2	2	0.98	0.77	0.96	160.6	70.4	160.6	169.1	76.1	169.1	160.6	72.1	160.6
3	2	4	0.89	0.73	0.71	160.6	70.4	180.6	169.1	76.1	190.4	160.6	72.1	182.5
4	1	1	0.88	0.83	0.76	180.6	52.0	135.6	190.4	57.6	144.1	182.5	53.7	135.6
4	1	3	0.76	0.72	0.50	180.6	52.0	155.6	190.4	57.6	165.4	182.5	53.7	157.5
4	1	2	0.89	0.72	0.74	180.6	52.0	160.6	190.4	57.6	169.1	182.5	53.7	160.6
4	1	4	0.80	0.67	0.49	180.6	52.0	180.6	190.4	57.6	190.4	182.5	53.7	182.5
4	2	1	0.91	0.81	0.74	180.6	70.4	135.6	190.4	76.1	144.1	182.5	72.1	135.6
4	2	3	0.83	0.72	0.51	180.6	70.4	155.6	190.4	76.1	165.4	182.5	72.1	157.5
4	2	2	0.90	0.72	0.71	180.6	70.4	160.6	190.4	76.1	169.1	182.5	72.1	160.6
4	2	4	0.82	0.63	0.47	180.6	70.4	180.6	190.4	76.1	190.4	182.5	72.1	182.5

Table B.7 Pure Strategies For the Parallel Organization

DMs						Slow			Medium			Fast		
1	2	3	Js	Jm	Jf	g1	g2	g3	g1	g2	g3	g1	g2	g3
1	1	1	1.00	1.00	1.00	101.7	101.7	101.7	109.9	109.9	109.9	101.7	101.7	101.7
1	1	3	0.95	0.90	0.83	101.7	101.7	116.7	109.9	109.9	125.9	101.7	101.7	118.1
1	1	2	0.99	0.92	0.98	101.7	101.7	118.7	109.9	109.9	126.9	101.7	101.7	118.7
1	1	4	0.95	0.90	0.84	101.7	101.7	133.7	109.9	109.9	142.9	101.7	101.7	135.1
1	2	1	0.96	0.92	0.84	101.7	116.7	101.7	109.9	125.9	109.9	101.7	118.1	101.7
1	2	3	0.92	0.85	0.68	101.7	116.7	116.7	109.9	125.9	125.9	101.7	118.1	118.1
1	2	2	0.94	0.88	0.83	101.7	116.7	118.7	109.9	125.9	126.9	101.7	118.1	118.7
1	2	4	0.91	0.83	0.69	101.7	116.7	133.7	109.9	125.9	142.9	101.7	118.1	135.1
1	3	1	1.00	0.90	0.98	101.7	118.7	101.7	109.9	126.9	109.9	101.7	118.7	101.7
1	3	3	0.91	0.83	0.81	101.7	118.7	116.7	109.9	126.9	125.9	101.7	118.7	118.1
1	3	2	0.99	0.88	0.95	101.7	118.7	118.7	109.9	126.9	126.9	101.7	118.7	118.7
1	3	4	0.93	0.84	0.82	101.7	118.7	133.7	109.9	126.9	142.9	101.7	118.7	135.1
1	4	1	0.91	0.88	0.83	101.7	133.7	101.7	109.9	142.9	109.9	101.7	135.1	101.7
1	4	3	0.87	0.82	0.69	101.7	133.7	116.7	109.9	142.9	125.9	101.7	135.1	118.1
1	4	2	0.93	0.85	0.83	101.7	133.7	118.7	109.9	142.9	126.9	101.7	135.1	118.7
1	4	4	0.86	0.82	0.71	101.7	133.7	133.7	109.9	142.9	142.9	101.7	135.1	135.1
2	1	1	0.94	0.94	0.87	116.7	101.7	101.7	125.9	109.9	109.9	118.1	101.7	101.7
2	1	3	0.89	0.87	0.71	116.7	101.7	116.7	125.9	109.9	125.9	118.1	101.7	118.1
2	1	2	0.94	0.87	0.83	116.7	101.7	118.7	125.9	109.9	126.9	118.1	101.7	118.7
2	1	4	0.90	0.83	0.68	116.7	101.7	133.7	125.9	109.9	142.9	118.1	101.7	135.1
2	2	1	0.89	0.85	0.72	116.7	116.7	101.7	125.9	125.9	109.9	118.1	118.1	101.7
2	2	3	0.83	0.81	0.57	116.7	116.7	116.7	125.9	125.9	125.9	118.1	118.1	118.1
2	2	2	0.89	0.80	0.65	116.7	116.7	118.7	125.9	125.9	126.9	118.1	118.1	118.7
2	2	4	0.84	0.78	0.52	116.7	116.7	133.7	125.9	125.9	142.9	118.1	118.1	135.1
2	3	1	0.91	0.85	0.81	116.7	118.7	101.7	125.9	126.9	109.9	118.1	118.7	101.7
2	3	3	0.86	0.77	0.67	116.7	118.7	116.7	125.9	126.9	125.9	118.1	118.7	118.1
2	3	2	0.94	0.80	0.83	116.7	118.7	118.7	125.9	126.9	126.9	118.1	118.7	118.7
2	3	4	0.87	0.76	0.67	116.7	118.7	133.7	125.9	126.9	142.9	118.1	118.7	135.1
2	4	1	0.84	0.83	0.68	116.7	133.7	101.7	125.9	142.9	109.9	118.1	135.1	101.7
2	4	3	0.79	0.79	0.54	116.7	133.7	116.7	125.9	142.9	125.9	118.1	135.1	118.1
2	4	2	0.88	0.78	0.72	116.7	133.7	118.7	125.9	142.9	126.9	118.1	135.1	118.7
2	4	4	0.83	0.73	0.60	116.7	133.7	133.7	125.9	142.9	142.9	118.1	135.1	135.1

Table B.7 Pure Strategies For the Parallel Organization (Continued)

DMs						Slow			Medium			Fast	
1	2	3	Js	Jm	Jf	g1	g2	g3	g1	g2	g3	g1	g2	g3
3	1	1	0.99	0.93	0.98	118.7	101.7	101.7	126.9	109.9	109.9	118.7	101.7	101.7
3	1	3	0.95	0.87	0.82	118.7	101.7	116.7	126.9	109.9	125.9	118.7	101.7	118.1
3	1	2	0.98	0.86	0.97	118.7	101.7	118.7	126.9	109.9	126.9	118.7	101.7	118.7
3	1	4	0.94	0.82	0.83	118.7	101.7	133.7	126.9	109.9	142.9	118.7	101.7	135.1
3	2	1	0.95	0.86	0.83	118.7	116.7	101.7	126.9	125.9	109.9	118.7	118.1	101.7
3	2	3	0.90	0.79	0.69	118.7	116.7	116.7	126.9	125.9	125.9	118.7	118.1	118.1
3	2	2	0.93	0.80	0.81	118.7	116.7	118.7	126.9	125.9	126.9	118.7	118.1	118.7
3	2	4	0.86	0.76	0.68	118.7	116.7	133.7	126.9	125.9	142.9	118.7	118.1	135.1
3	3	1	0.99	0.88	0.96	118.7	118.7	101.7	126.9	126.9	109.9	118.7	118.7	101.7
3	3	3	0.95	0.86	0.80	118.7	118.7	116.7	126.9	126.9	125.9	118.7	118.7	118.1
3	3	2	0.97	0.73	0.94	118.7	118.7	118.7	126.9	126.9	126.9	118.7	118.7	118.7
3	3	4	0.90	0.70	0.79	118.7	118.7	133.7	126.9	126.9	142.9	118.7	118.7	135.1
3	4	1	0.94	0.86	0.83	118.7	133.7	101.7	126.9	142.9	109.9	118.7	135.1	101.7
3	4	3	0.89	0.80	0.70	118.7	133.7	116.7	126.9	142.9	125.9	118.7	135.1	118.1
3	4	2	0.90	0.69	0.81	118.7	133.7	118.7	126.9	142.9	126.9	118.7	135.1	118.7
3	4	4	0.83	0.67	0.68	118.7	133.7	133.7	126.9	142.9	142.9	118.7	135.1	135.1
4	1	1	0.95	0.91	0.83	133.7	101.7	101.7	142.9	109.9	109.9	135.1	101.7	101.7
4	1	3	0.91	0.82	0.70	133.7	101.7	116.7	142.9	109.9	125.9	135.1	101.7	118.1
4	1	2	0.94	0.84	0.84	133.7	101.7	118.7	142.9	109.9	126.9	135.1	101.7	118.7
4	1	4	0.88	0.80	0.69	133.7	101.7	133.7	142.9	109.9	142.9	135.1	101.7	135.1
4	2	1	0.89	0.84	0.69	133.7	116.7	101.7	142.9	125.9	109.9	135.1	118.1	101.7
4	2	3	0.84	0.77	0.55	133.7	116.7	116.7	142.9	125.9	125.9	135.1	118.1	118.1
4	2	2	0.89	0.80	0.67	133.7	116.7	118.7	142.9	125.9	126.9	135.1	118.1	118.7
4	2	4	0.84	0.77	0.52	133.7	116.7	133.7	142.9	125.9	142.9	135.1	118.1	135.1
4	3	1	0.95	0.86	0.84	133.7	118.7	101.7	142.9	126.9	109.9	135.1	118.7	101.7
4	3	3	0.91	0.80	0.70	133.7	118.7	116.7	142.9	126.9	125.9	135.1	118.7	118.1
4	3	2	0.89	0.70	0.82	133.7	118.7	118.7	142.9	126.9	126.9	135.1	118.7	118.7
4	3	4	0.82	0.68	0.68	133.7	118.7	133.7	142.9	126.9	142.9	135.1	118.7	135.1
4	4	1	0.90	0.81	0.72	133.7	133.7	101.7	142.9	142.9	109.9	135.1	135.1	101.7
4	4	3	0.87	0.76	0.58	133.7	133.7	116.7	142.9	142.9	125.9	135.1	135.1	118.1
4	4	2	0.80	0.66	0.69	133.7	133.7	118.7	142.9	142.9	126.9	135.1	135.1	118.7
4	4	4	0.76	0.63	0.58	133.7	133.7	133.7	142.9	142.9	142.9	135.1	135.1	135.1

B.5 TOTAL ACTIVITY FOR BEHAVIORAL STRATEGY

Equations (B.1) and (B.2) can still be applied except that the probabilities of the system variables have to be computed for the behavioral strategy. In this section the computation of the marginal probability of a system variable will be described.

The probability of a system variable corresponding to a behavioral strategy can be expressed as

$$p(w) = \sum_{k^1k^2k^3} p(w \mid \Delta_{k^1k^2k^3}) \cdot p^1_{k^1} \cdot p^2_{k^2} \cdot p^3_{k^3} \tag{B.3}$$

where k^1, k^2, and k^3 are equal to 1, 2, 3, or 4.

As an example, consider the system variable w19 which is in the estimation algorithm in the SA stage (f2). From Section B.2, w19 is true when the number of aircraft in a threat is less than 5 ($m < 5$). In equation (B.3), when k^1, k^2, and k^3 take values of 1 and 2, the estimation algorithm is active, that is, $p(w19 \mid \Delta_{k^1k^2k^3}) \neq 0$. Then,

$$\begin{aligned} p(w_{19}) &= p(w19 \mid \Delta_{111})\, p^1_1\, p^2_1\, p^3_1 + p(w19 \mid \Delta_{112})\, p^1_1\, p^2_1\, p^3_2 + \\ &\quad p(w19 \mid \Delta_{121})\, p^1_1\, p^2_2\, p^3_1 + \ldots + p(w19 \mid \Delta_{222})\, p^1_2\, p^2_2\, p^3_2 \\ &= \sum_{i,j} p(w19 \mid \Delta_{1ij})\, p^1_1\, p^2_i\, p^3_j + \sum_{i,j} p(w19 \mid \Delta_{2ij})\, p^1_2\, p^2_i\, p^3_j \end{aligned}$$

Because

$$p(w_{19} \mid \Delta_{1ij}) = p(w_{19} \mid \Delta_{2ij}) = 0.6, \qquad \forall\, i,\ j$$

then,

$$\begin{aligned} p(w_{19}) &= 0.6\, p^1_1 \sum_{i,j} p^2_i p^3_j + 0.6\, p^1_2 \sum_{i,j} p^2_i p^3_j \\ &= 0.6\,(p^1_1 + p^1_2) \sum_{i,j} p^2_i p^3_j \end{aligned}$$

The last term in the above equation is equal to one. Therefore, the probability of w19 is

$$p(w_{19}) = 0.6\,(p^1_1 + p^1_2)$$

For each value of P^1_1 and p^1_2, p(w19) can be computed. The entropy of w19 is

$$H(w_{19}) = -[p(w_{19})\log_2[p(w_{19})] + [1 - p(w_{19})]\log_2[1 - p(w_{19})]$$

The same procedure is applied to all system variables; the total activity corresponding to a behavioral strategy then is computed using equations (B.1) and (B.2). When all possible values of p^i_j, for i = 1, 2, 3 and j = 1, 2, 3, 4, are used to compute the entropies for all system variables, all the values in the workload space can be obtained.

To construct the Performance-Workload locus, only a small set of p^i_j's is used. Because of the convexity properties of the J-G relation, not all points need to be computed. The method used to construct the J-G locus is the following.

Since there are two algorithms for the SA and RS stages, the strategy used can be characterized by the probability that each algorithm is selected. Let u denote the algorithm in the SA stage and v denote the algorithm in the RS stage. The probability of an algorithm being selected is

$$p(u = 1), p(u = 2), p(v = 1), \text{ and } p(v = 2).$$

As described in Chapter 5, the pure strategies for individual DMs are

Pure strategy D1:	$p(u=1, v=1) = 1.0$;
Pure strategy D2:	$p(u=1, v=2) = 1.0$;
Pure strategy D3:	$p(u=2, v=1) = 1.0$;
Pure strategy D4:	$p(u=2, v=2) = 1.0$.

Because the choices of algorithms in the different stages are independent, a pure strategy for a DM can be expressed by p(u)p(v). For a mixed strategy, D, the general expression is

$$D(p_k) = \sum_{k=1}^{n} p_k D_k$$

where p_k is the probability that the kth pure strategy is selected. In terms of u and v, p_k can be computed by

$$p_1 = p(u = 1)p(v = 1)$$
$$p_2 = p(u = 1)p(v = 2) = p(u = 1)[1 - p(v = 1)]$$
$$p_3 = p(u = 2)p(v = 1) = [1 - p(u = 1)]p(v = 1)$$
$$p_4 = p(u = 2)p(v = 2) = [1 - p(u = 1)]\,[1 - p(v = 1)]$$

since p(u = 2) is equal to 1 - p(u = 1) and p(v = 2) is equal to 1 - p(v = 1). Therefore, only p(u = 1) and p(v = 1) need to be specified and varied in order to construct the J-G locus. Specific numbers used in the computation for this study are 0, 0.7, 1.0. These values are used for both p(u = 1) and p(v = 1). There are 3^2, or 9, mixed strategies. Table B.7 shows all mixed strategies.

Table B.8 Mixed Strategies for Given Probabilities

p(u = 1)	p(v = 1)	p_1	p_2	p_3	p_4
0	0	0	0	0	1
0	0.7	0	0	0.7	0.3
0	1	0	0	1	0
0.7	0	0	0.7	0	0.3
0.7	0.7	0.49	0.21	0.21	0.09
0.7	1	0.7	0	0.3	0
1	0	0	1	0	0
1	0.7	0.7	0.3	0	0
1	1	1	0	0	0

There are three decision makers in the organizations. The number of behavioral strategies is the combination of the individual strategies. Let K denote number of behavioral strategy for a organization and K_i denote number of strategies implemented for the ith DM. K is computed by

$$K = \prod_{i=1}^{3} K_i$$

For the hierarchical organization, each of the subordinates has 9 strategies and the supervisor has 3 strategies (because there only one stage (SA) in which the supervisor has different algorithms to select). The number of behavioral strategies for the hierarchical organization is 243.

For the parallel organization, every DM has 9 strategies. The number of behavioral strategies for the parallel organization is 729.

The workload corresponding to all these behavioral strategies must be computed for constructing the J-G locus.

APPENDIX C EXPERIMENT DISPLAY AND INSTRUCTIONS

The experiment involves a small computer network. In this Appendix, the displays and command functions for each role will be described in detail and the instruction for doing the experiment will be given.

As discussed in previous chapters, the experiment simulates a naval outer air battle. Each E2C has a radar which covers one sector of the defense area. There are three concentric circles representing a radar screen. The goal is to defend the center of the circles, where the command center is located. There are several "threats" or "targets" (e.g., enemy aircraft) converging simultaneously to the center. The particular display for each role is described in the following sections.

Display for E2C Mission Commander

Figures C.1 and C.2 show the displays for E2C mission commanders in the parallel structure and in the hierarchical structure, respectively. The only difference is that the defense area is divided into three sectors in the parallel structure but only in two sectors in the hierarchical structure. A radar screen and three status windows, as well as a control panel are displayed. The detailed description of each window follows:

Radar screen

The three circles represent different regions of the defense area. The largest circle with radius R1(Fig. C.3) is the region that the radar can cover. When threats enter this region, a DM determines the position and speed of the threats. The second circle with radius R2 in Fig. C.3 is the region in which more information can be obtained, such as the number of aircraft in a threat. The smallest circle defines the enemy's missile release line at which the carrier may be in danger. If a threat reaches this region, the defense is considered as having failed with respect to that threat. For the convenience of the description, let us call the annular area between the largest and the second circles in Fig. C.3 as region 1 and call the area inside the second circle as region 2.

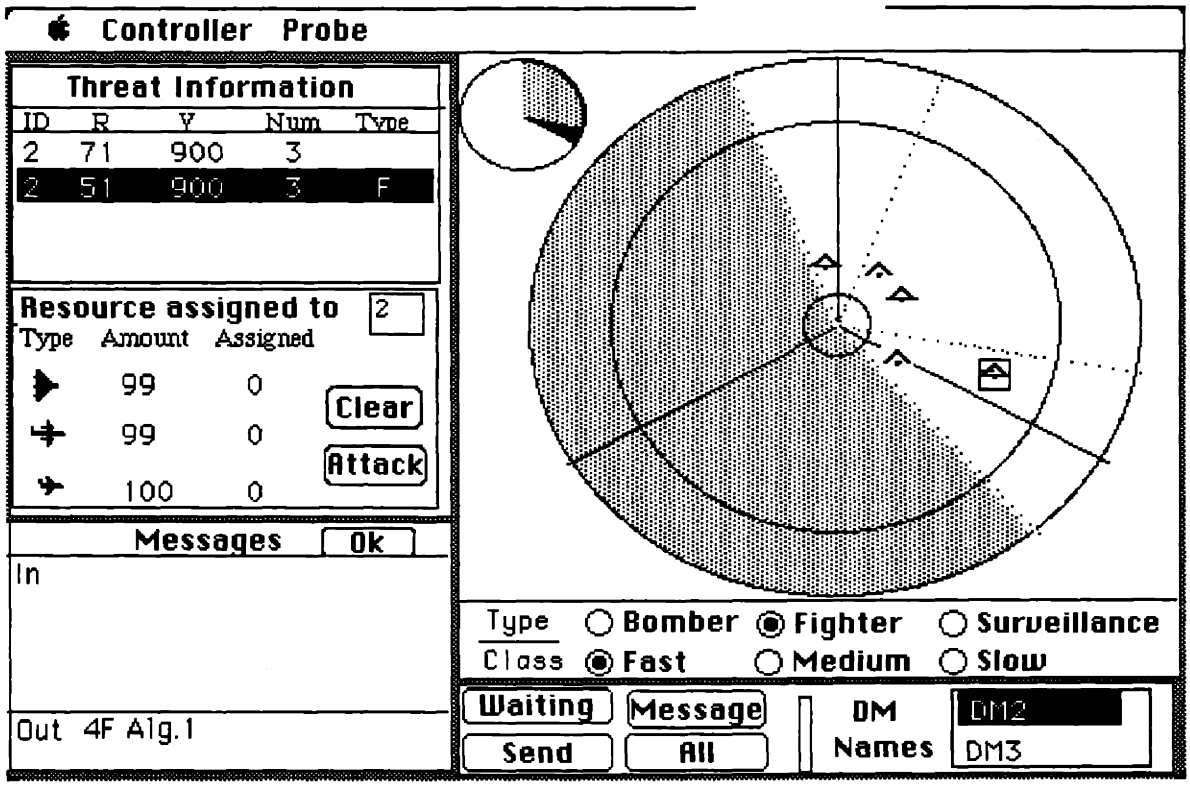


Figure C.1 Display for E2C Mission Commander in the Parallel Structure

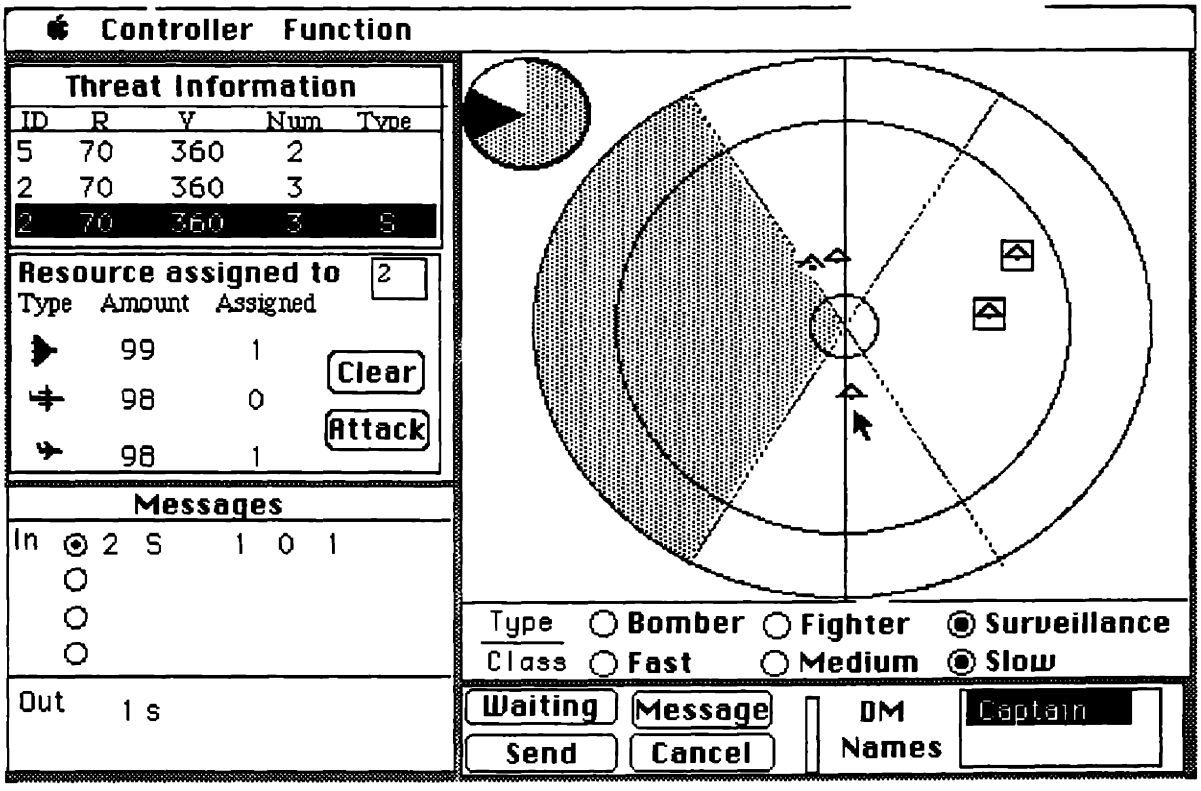


Figure C.2 Display for E2C Mission Commander in the Hierarchical Structure

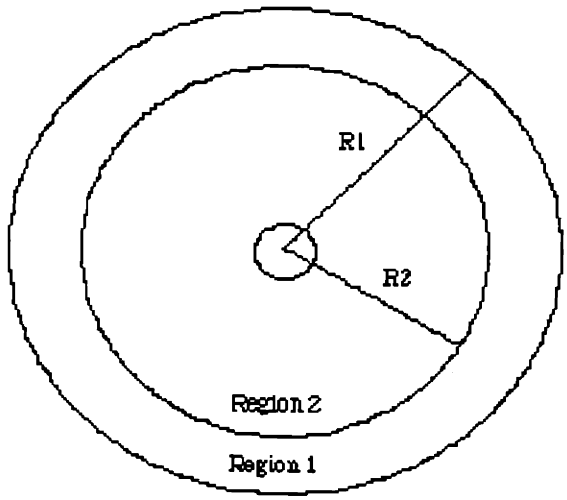


Figure C.3 Defense Region Shown in the Radar Screen

There are two symbols for threats:

means the threat consists of few aircraft, that is, the number of aircraft is less than five.

means the threat consists of many aircraft, that is, the number of aircraft is equal to or larger than five.

When the type of a threat is determined, a horizontal bar will appear under its icon to indicate the readiness for allocating the resources. When a threat is attacked, a square box will outline its icon. For example, in Fig. C.1 the types of two threats have been determined and one threat has been attacked.

In addition, a pie–chart clock representation is shown on the upper left corner of the window (Fig.C.1 and Fig.C.2). The clock depicts the remaining time (black area) and the time elapsed (gray area) during the experiment. The purpose of the clock is to assist DMs in allocating time properly during the execution of the task.

Control panel

The control panel is underneath the radar screen (Figs. C.1 and C.2). There are three parts in the control panel: the name list, a set of functional buttons (circles), and a set of control buttons (rectangles with rounded corners).

The name list displays the names of DMs in the organization and can be used as a reminder in addressing messages.

A set of functional buttons (circles) is for assigning the class and type to threats. From the information available about a threat, DM can determine the class and the type of the threat. The *class* of the threat specifies whether the threat is fast, medium, or slow according to its speed. The *type* of the threat refers to the type of the aircraft that a threat contains, that is, fighter, bomber, or surveillance aircraft. The type can be determined from the class and the number of aircraft in a threat as shown in Table C.1. Knowing the type of a threat is necessary for allocating the resources.

There are two rows of functional buttons. The first row is for selecting class; and the second is for assigning the type. Only *one* button can be chosen in each row. A black dot inside a button indicates that the button is selected. When another one is selected, the one previously chosen is released.

A set of control buttons (rounded rectangles) is used for sending messages. To send a message, a team member has to be selected by choosing his name. Then, the **Message** button is selected to review the message going to be sent. If the message is correct, the **Send** command is issued to transfer the message; otherwise, the **Cancel** button can be selected to modify the message. The **Waiting** button should be activated when a DM has nothing to do but is waiting for information from other DMs in the team.

Threat Information window

The threat information window is on the upper left corner of the screen (Fig. C.1 and Fig. C.2). Threat attributes will be displayed in this window. The attributes of a threat are as follows:

Attributes={threat ID, position, speed, number, type}

The position is the distance from the center. Resources cannot be allocated before knowing all the attributes of a threat. The attributes of a threat can be probed by selecting the threat icon on the screen. When probing the attributes of a threat, the information will be

displayed in this window. Because there is difference in the amount of information available in different regions of the defense area, the attributes which are not available will be left blank. For example, in the threat information window of Fig. C.2, the type of the threat with ID 2 is a surveillance aircraft represented by S in the third line, while the type of the threat with ID 5 is a blank space because it has not been assigned.

Resource window

The resource window is under the information window (Fig. C.1 and C.2).

On the upper right corner of this window, the threat that is currently being processed is indicated. The rectangular box following "Resource assigned to" displays the ID of the threat. There are three types of resources shown by different icons in this window. From Fig. C.1, the first icon represents a Tomcat (or F-14) fighter plane ; the second one represents a Hornet (or F-18) fighter plane, and the third one is an Prowler (or EA-6B). The symbols for these resources are shown in Fig. C.4. The number of available resources and the number assigned to a threat will appear on the row of that resource. For example, in Fig. C.2, the ID of the threat being processed is 2. One F-14 and one EA-6B are assigned to the threat.

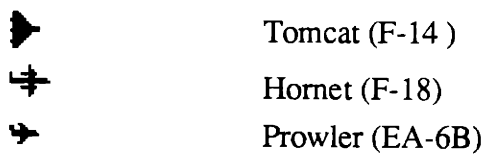


Fig. C.4 Symbols of the Resources

The **Clear** button can be used to cancel the current resource assignment. The **Attack** button is chosen as the final action on a threat.

Message window

This window is under the resource window. The window consists of two message areas:

Incoming message: the message sent by other DMs in the team.
Outgoing message: the message to be sent to others.

The content of the messages is different in parallel and hierarchical structures. In the parallel structure, a message must include the threat ID, the type of the threat, and the algorithm used to determine the type. In Fig. C.1, an outgoing message of the parallel structure is shown: "4 F Alg.1". It means that the message is for the threat whose ID is 4 (4), the type of

the threat is fighter (F), the algorithm used to determine the type of the threat is algorithm 1 (Alg.1). In the parallel structure, incoming messages have the same format and content as outgoing messages.

In the hierarchical structure, outgoing messages of subordinates contain threat ID and the class of threat. In Fig. C.2, the outgoing message is "1 s" which means the ID of the threat is 1 (1) and the class is slow (s). The incoming message comes from the AAW Commander (the supervisor) and has different content. The message includes the ID of the threat, the type of the threat determined by the supervisor who has global information, and the selection of resources for intercepting the threat. In the "in" message area of Fig. C.2, "2 S 1 0 1" means that for threat ID 2 (2), the type is surveillance (S), one F14 and one AE-6B (1 0 1) may be allocated to intercept the threat.

Display for AAW Commander (Commander on the carrier)

Figure C.5 shows the display for the AAW Commander(the supervisor).

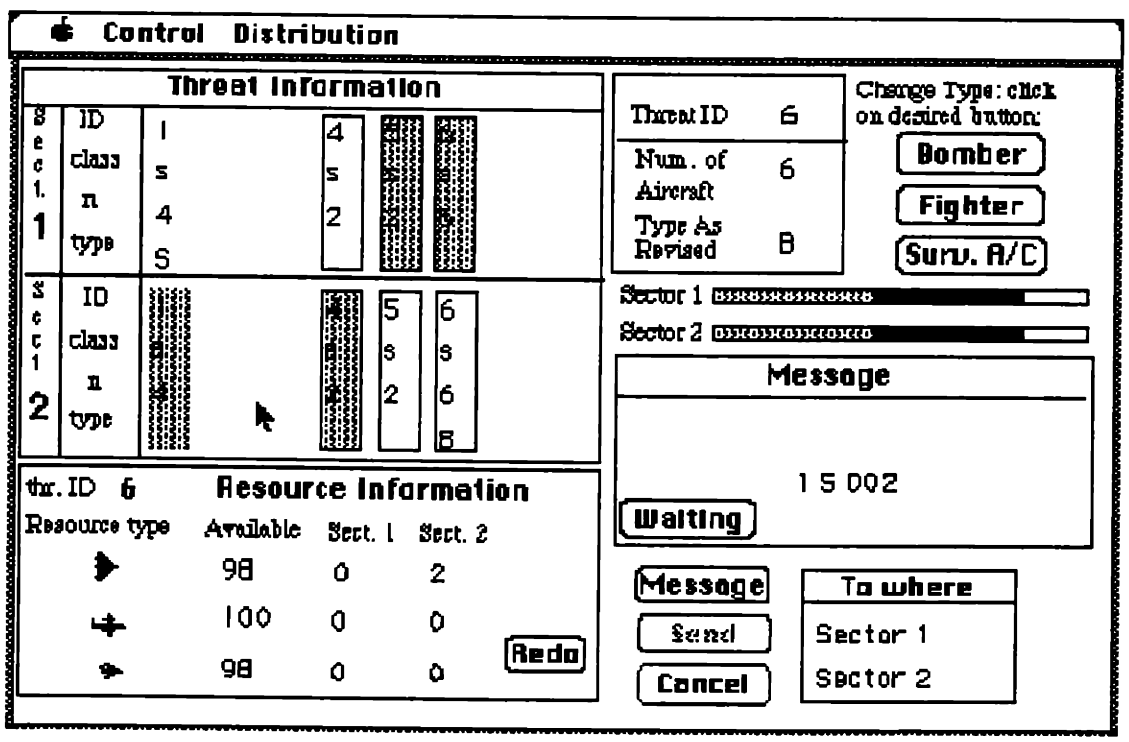


Figure C.5 Display for AAW Commander in the Hierarchical Structure

There are two threat information windows, named Sect.1 and Sect.2. There are also a resource window, a type window, a message window, and a control panel.

Threat Information Windows

Threat information windows are on the upper left of the display. The windows display the information reported by the corresponding subordinates. Sect.1 represents DM1 in the sector 1; Sect.2 is DM2 in the sector 2. The message received is displayed as a column within a threat box in the window corresponding to the sector where the message comes from. If the threat is not in the sector from where the message is sent, the display is in a gray shaded box, as threat 4 is in Sect.2 in Fig. C.5. The information displayed in a threat box is the ID, the class, the number of aircraft, and type of the threat. When information comes in, the type of the threat has not been determined, therefore, is not shown, e.g., threat 4 in Fig. C.5. The threats not being processed have a rectangular box around them. When the process is completed for a threat, the outline box disappears. For example, threat 1 in Fig. C.5 is completely processed while the others are not.

Type window

The type window is on the upper right of the display and is used to assign a type to threats. There are three buttons: Bomber, Fighter, and Surv.A/C (abbreviation for surveillance aircraft). A square box on the left of the buttons shows the threat ID, the number of aircraft , and the type assigned to the threat currently under the process. When a button is pressed, you are assigning a type to the threat whose threat ID is displayed in the box on the left. For example, in Fig. C.5, the threat with ID 6 is the one currently processed; there are six (6) aircraft, and the threat type is bomber (B).

Time bars

There are two bars under the type window. These bars are time bars which indicate time past and time available in the sector 1 and sector 2, respectively.

Resource window

The resource window is under the threat information window (Fig. C.5). The function of this window is basically the same as the one for the subordinate. The only difference is the resource allocated to different sectors is shown in separate columns. The **Redo** button allows modification of the resource assignment.

<u>Message window</u>

The message window is used to check messages that are going to be sent to subordinates. In Fig. C.5, the message "1 S 0 0 2" means that threat ID 1 (1) is a surveillance aircraft (S) and

two EA-6B (0 0 2) are selected for intercepting this threat. The **Waiting** button is pressed when the supervisor is waiting for the report before able to do anything.

Control panel

The control panel is used for sending commands to the sectors. The selection of the sector to which threat is assigned is indicated by highlighting the name of the sector. The **Message** button allows the supervisor to see if the message is correct. He can select the **Send** button to send the message to the selected sector, or modify the message by selecting the **Cancel** button. After sending the message out, the task for that threat is completed.

C.2 INSTRUCTION OF THE EXPERIMENT

Instruction for doing the experiment was given to subjects before the start of training. Subjects were asked to read the previous section to learn the display, then read the following instructions. The text in the instructions is as follows.

This section describes how you play the game. There are four actions you need to take sequentially for processing each threat. These actions are: 1)probe for raw data, 2) assign the class and type of the threat, 3) allocate resources, and 4) attack the threat. The result of each action is required by the following action. Therefore, the sequence of actions has to be kept in a fixed order. No one action can be skipped. For example, you can not determine the type before specifying the class of a threat; and you cannot allocate resources before you know the type of a threat. Fig.C.6 shows the block diagram of the sequence of the actions. In the following paragraphs, each action will be described in detail.

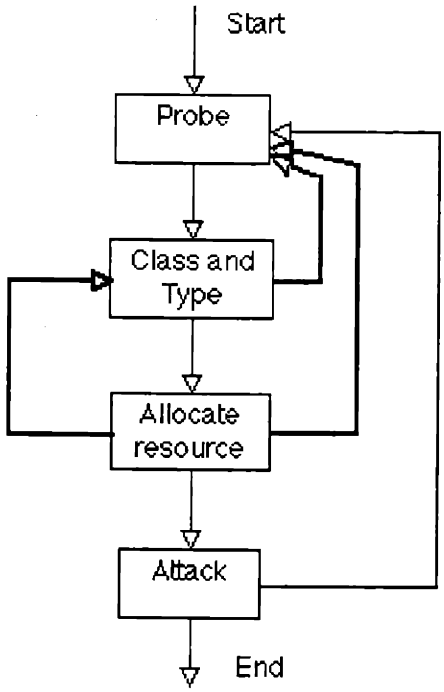


Figure C.6 Four actions in the experiment

1. Probe

Probing is done by clicking on a threat. This is the first action you should take after the game starts. More than one threat will appear on the radar screen. You have to specify which one of them you will process by clicking on the threat's icon. You cannot do anything before you click on at least one threat.

After you click on a threat, raw data pertaining to that threat will be shown in the information window. The raw data includes the identification number, position, speed , and the number of aircraft in the threat. The number of aircraft in the threat will not be shown if a threat is in the region 1, that is, in the outside ring of the radar screen.

This action can be repeated as many times as needed until the threat is attacked. Unless another threat is probed (clicked on), any succeeding actions will refer to the threat clicked on most recently. In Fig. C.6, the bold line indicates the iteration for the same threat. You can go back to previous action to make a modification any time before the threat is attacked.

2. Assign class and type of a threat

The class of a threat is determined by its speed. The class of the threat is assigned by clicking on the desired class button on the Type-Class panel of the radar screen. Table C.1 is the Class-Speed table.

Table C.1 Class-Speed Table

Speed	Class
< 400	Slow
400 - 800	Medium
>800	Fast

After the class is determined, the type can be found according to the class of the threat and the number of aircraft in the threat. However, the number of aircraft will not be available until the threat enters the region 2. Two methods can be used to find the type of a threat. One method is to estimate and the other is to probe.

Estimate

There are two methods for estimating the type. The first one estimates the type by the class and the icon shape of the threat. At this time, the number of aircraft is not available, that is, the threat has not entered region 2 yet. As described in Part 1, there are two threat icons: one indicates that the number of aircraft is less than five; the other indicates that the number of aircraft is equal to or larger than five.

In the second method, the threat enters region 2 so that the class and the number of aircraft are both available. Since more information is given, the estimation is more accurate than according to the first method. However, you can not use the second method for estimation before the threat enters region 2. Therefore, there are time-accuracy trade-offs.

To assign type to a threat, click on the threat, then, click on the corresponding button in the Type-Class panel. Although the radio button will indicate the type you have assigned to a threat, you may click on the threat to see the type appear in the information window. The possible types associated with combinations of class and number of aircraft are shown in Table C.2. In Table C.2, F, B, and S represent fighter, bomber, and surveillance aircraft respectively. In some case, there are more than one type associated to a combination of class and number of aircraft. This creates uncertainty. When there are more than one possible type, the chance for the threat to be any of the admissible types is equally likely.

Probe

If you want a more accurate result, you can activate the probe algorithm by choosing the **Probe** command in the menu bar, then drag down to select the **Type** command in it. There is a time delay of four seconds associated with this method. You have to wait for four seconds.

During this delay period, you *cannot* take any action except to move the mouse to the position for the next click. When the algorithm returns, the attributes of the threat will be shown in the information window. This algorithm can only be used when the threat enters region 2.

Table C.2 Type of Threats

	Fast	Medium	Slow
N < 5	F	F, S	S
N = 5	F, B	F, B, S	B, S
N > 5	B	B	B

3. Allocate resources to a threat

To allocate resources to a threat whose type has been determined, you can also choose from two methods. As in determining the type, one method is to estimate and the other is to probe.

Estimate

You can click on the resource icon in the resource window to allocate resources to the threat being processed. Each click is counted as assigning one unit of that type of resource. When you are assigning resources, the number available and the number assigned will change and be shown in the window.

Table C.3 shows the capability of resources to handle different types of threats. Each number in the table represents the number of threats that one particular resource can deal with. For example, in row 1 and column 1 of Table C.3, 2 indicates that one F14 should be assigned to intercept two fighter threats.

Table C.3 Resource Allocation Matrix

	F14	F18	A-6E
Fighter	2	1	0
Bomber	3	2	1
Surveillance	4	3	1

There are more than one correct solution in resource allocation. If there are m aircraft in a threat, we can write the following three equations according to Table C.4.

If threat is a fighter:	$2y_1 + y_2 = m$	(C.1)
If threat is a bomber:	$3y_1 + 2y_2 + y_3 = m$	(C.2)
If threat is a surveillance aircraft:	$4y_1 + 3y_2 + y_3 = m$	(C.3)

where y_1, y_2, y_3 are the number of F14, F18, and EA-6B needed, respectively. For a given type of threat, any solution to the corresponding equation will be a correct resource allocation.

Probe

To get a more accurate resource allocation, you can choose the **Probe** command in the menu bar, then drag down to select the **Resource** command. There is a time delay of four seconds associated with this algorithm. You have to wait for four seconds for the screen to return to active status. During this delay period, you cannot take any action except to move the mouse to the position for the next click. When the screen returns, the number of assigned resources and the number of available resources will be shown in the resource window.

The ID of the threat you are working on is shown on the upper right corner of the resource window.

4. Attack a threat

After allocating resources to a threat, you have two options for the next action. You can attack the threat by choosing the **Attack** button, or you can modify the resource allocation by pressing the **Clear** button in the resource window. As soon as **Attack** is chosen, your task regarding that threat is completed. No further modification or action can be taken for that threat.

The four actions described above are the basic actions you must take during the experiment except for the threats in an overlap area *but* not in your sector. For each threat, the sequence of the four actions has to be kept in order. However, interleaves between the processes of different threats are allowed. You can stop working on one threat at any action and start to work on another threat at the stage previously worked. This kind of interleaving occurs when you click on another threat. The thin lines in Fig. C.6 indicate possible interleaves. Note that a click on a threat is *required* whenever you are going to do something for a threat which is not the one currently being processed.

For the threats in the overlap area, there are different cases for different organizational structures. If you are going to do the experiment for parallel organizations, please read the following section. For hierarchical organizations, please skip to section B.

Section A. Parallel Organization

Case 1:

The threat is in the overlap area *but* not in your sector. In this case, at the end of action 2, the type of threat is determined. Then, you have to send the results to the DM whose sector includes the threat. To send a message, first specify to whom you want to send the message by clicking on the name of your peer. The selected name is highlighted. Then, click the **Message** button in the control panel. The message ready to be sent will appear on the message window with the form: threat ID, type of the threat, and the method used to determine the type. For example, "3 F Alg.1" means the threat with ID 3 is a fighter determined by using method 1, that is, estimate method. If the message is correct, click on the **Send** button to send; if it is not, click on the **Cancel** button to redo it.

Please keep in mind that in this case you may want to use the first level estimation because you need to send the result to the other DM who is waiting for your message.

Case 2:

The threat is in the overlap area *and* in your sector. In this case, you have to wait for the message from the other DM before you can allocate resources. The other DM will do exactly the same as you do in Case 1. When you receive the message, if the method used is "probe", that is, there is "Alg.2" in the message, then you can accept the type received by clicking the **Ok** button in the message window. If the method used is "estimate", that is, "Alg.1" is in the message, you have the choice of accepting or redoing yourself by using either method of action 2.

Section B. Hierarchical Organization

Subordinate

When a threat is in the overlap area, you have to send the information of the threat to the supervisor who will look at the overall situation and give an estimation on the type. You assign the class to the threat according to its speed and send it to the supervisor. To send a message, first specify to whom you want to send the message by clicking on the name of the supervisor. (In this case, you only can send messages to the supervisor since there is no communication between two subordinates.) Then, click the **Message** button in the control panel to review the message. If the message is correct, click on the **Send** button to send; or click on the **Cancel** button to redo it. You have to wait for the command form the supervisor before you can allocate the resources to this threat.

The command from the supervisor contains the type of threat and the resource allocation for the threat. After receiving the command, you accept the type of the threat determined by supervisor since he has a more accurate algorithm to decide the type. However, you can decide whether to accept the resource allocation by the supervisor or make some modifications according to the local situation. Then the threat can be attacked. Click on the **Attack** button in the resource window to attack, or click on the threat icon, then go to the resource window to re-select resources.

If the threat is not assigned to you, you will see a square box around it when it is attacked by the other subordinate.

Supervisor

At the beginning of a trial, Sect.1 and Sect.2 windows are empty. When a message is received, it will be displayed in the window corresponding to the sector where the message comes from. As a supervisor, you have to wait for the information about a threat from both sectors before you can do anything to the threat. When there are more than one threat, you select a threat to process by clicking on its threat box.

If the estimate method is used, the type can be assigned by clicking on the desired button in the type window(Fig. C.5). If the "probe" algorithm is used, the menu bar command should be used. After determining the type, you select resources which can be used to intercept the threat.

Select the sector you want to assign the threat. The rule to assign a threat to a sector is to assign a threat to the sector which the threat is in. After selecting the sector, click on **Message** button to see if the message is correct. You can click on **Send** button to send the message to selected sector, or modify the message by clicking on **Cancel** button. After sending the message out, you complete the task for that threat. Information about a threat displayed without a threat box, e.g. threat 1 in Fig. C,5, indicates the process for that threat is complete.

Rules of the Game:

1. Coordination. You are working as a member of a team. The only performance that counts is the team performance, not an individual's performance. Therefore, always be aware of the threats in the overlap area but not in your sector; make sure you process them properly and in a timely manner.
2. As time pressure increases, you will sometimes have too little time to process the threats carefully. Unless you are confident that your response will be correct, it is better to risk letting time run out before you finish all of the threats rather than make arbitrary choice.

3. Use the "probe" method whenever time is available because that method gives more accurate results.

APPENDIX D. COMPUTATION OF MEASURES OF PERFORMANCE

Measure of performance computed in this appendix is accuracy index J.

Y is a column vector of dimension three.

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

where y_i in Y is the number of the type-i resources allocated to intercept the threat. The Resource-Threat matrix A is

$$A = \begin{bmatrix} 2 & 1 & 0 \\ 3 & 2 & 1 \\ 4 & 3 & 1 \end{bmatrix}$$

Each element of A represents the number of threat aircraft that one resource can intercept. Each row of A corresponds to a type of threat; for example, the first row is for a fighter; the second for a bomber, and the third for a surveillance aircraft. Each column of A corresponds to a type of resource: the first column is for F-14s, the second for F-18s, and the third for AE-6Bs. For example, the first column of A indicates that each F-14 can intercept 2 fighters, 3 bombers, and 4 surveillance aircraft. The second row of A means that if bombers are the threat, one F-14 can intercept 3 bombers, one F-18 can intercept 2, and one AE-6B can intercept 1.

Let $\underline{a}_i$ denote a row of matrix A; $\underline{a}_i$ indicates that resources can be used to engage a particular type of threat. Let n be the number of aircraft in a threat. Then the error in resource allocation, Δn, can be computed by

$$\Delta n = \underline{a}^i y - n.$$

There are two cases.

Case 1: $\Delta n \leq 0$.

When Δn is equal to zero, there is no error. When Δn is less than zero, not enough resources have been allocated to the threat. The actual error index is computed by

$$e_a = \text{abs}(\Delta n)\, c_i + (1 - a_{3i})\, y_3.$$

The first term on the right side is the penalty for the lack of resources. The second term is non-zero only when the threat is a group of fighters because in the third column of the A matrix the zero corresponds to fighters. This term represents the penalty for the misuse of the resource because AE-6B cannot be used to intercept fighters.

The maximum computed cost index is

$$e_c = n\, c_i$$

Case 2: $\Delta n > 0$. This case indicates the overuse of resources. The actual error index is

$$e_a = \Delta n;$$

and the maximum computed error index is

$$e_c = \max.\ \{\ n\, c_i,\ \text{Cost Matrix}(\ \text{type}, n)\ \}$$

J is the output of $C(Y_d, Y)$ which is defined as the following:

$$J = C(Y_d, Y) = \begin{Bmatrix} 1 - \frac{e_a}{e_m}, & \frac{e_a}{e_m} \le 1; \\ 0, & \frac{e_a}{e_m} > 1. \end{Bmatrix} \qquad \text{(D.1)}$$

where

$$e_m = \begin{Bmatrix} c \cdot m, & \Delta m < 0; \\ O_R, & \Delta m > 0. \end{Bmatrix} \qquad \text{(D.2)}$$

In the above equations, e_a is an actual error index and e_m is the maximum error computed by using equation (D.2) in which c belongs to the cost index vector C:

$$C = [\ 2\ 3\ 1]$$

Each element in C is a cost index for one particular type of threat, e.g., if a fighter is not intercepted, the cost index is 2. The cost index for a bomber is 3, and the cost index for a surveillance aircraft is 1. The reason to associate theses values to each type of threats is that

for more dangerous type of threats, the penalty for unsuccessful interception is higher. A bomber is considered to be the most dangerous threat; fighter is the second; and a surveillance aircraft is the least dangerous.

When e_a is larger than e_m, the corresponding Y is outside of the designated vector space of Y so that accuracy is zero. The designated vector space for Y consists of all y's which satisfy the assumption that the total number of resources used to intercept a threat is not larger than the number of aircraft in the threat.

APPENDIX E. EXPERIMENTAL DATA ANALYSIS

The objective is to determine whether there is any significant difference between the performance of two organizational structures (see Fig. E.1). The following hypotheses will be tested.

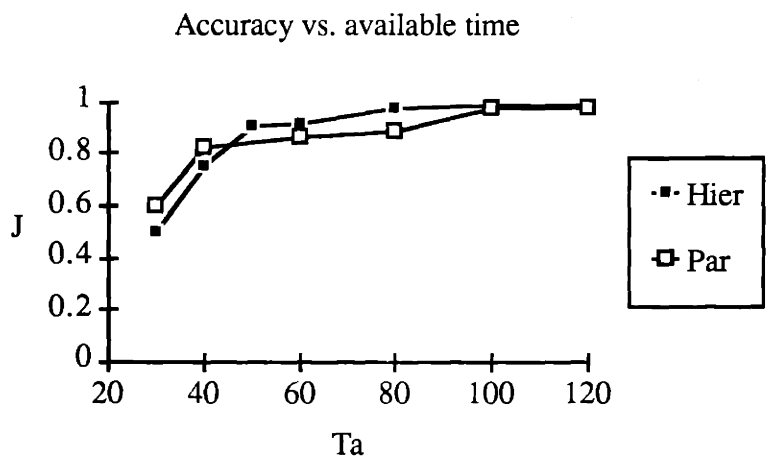


Figure E.1 Accuracy and the Available Time of Team #10.

The first hypothesis has the following alternatives:

H0: there is no difference in the accuracy between the two structures.
H1: there is a difference in the accuracy between the two structure.

Or, expressed in a simpler form,

H0:	D = 0,	no difference
H1:	D <> 0 ,	different

where D = Jx - Jy. The test is run for all time conditions. Since this is a two-tail test, the statistic is $z(1-\alpha/2)$ instead of $z(1-\alpha)$. Therefore, for 95% confidence level,

$$\alpha = 0.05, \qquad z(1-\alpha/2) = z(0.975)$$

Table E.1 shows the result.

Table E.1 Test Result for J: D = 0

Ta	z(0.975)	D(H-P)	A1	A2	Ho
30	1.645	-0.14	-0.04	0.04	Reject
40	1.645	-0.08	-0.04	0.04	Reject
60	1.645	0.01	-0.03	0.03	Accept
80	1.645	0.06	-0.02	0.02	Reject
100	1.645	0.01	-0.01	0.01	Accept
120	1.645	0.00	-0.01	0.01	Accept

Since the null hypothesis is rejected at Ta = 30, 40, and 80, there *is* a difference in accuracy for two structures at these available times. The second hypothesis test needs to be conducted. The alternatives are

H0: $D \geq \delta$, structure x is better than structure y.
H1: $D < \delta$, structure x is not better than structure y.

where D = Jx - Jy and δ is the difference. this is a one-tail test. for 95% confidence level, the test statistics is

$$z(1-\alpha) = z(0.95), \qquad \text{where } \alpha = 0.05.$$

Table E.2 shows the result of the test.

Table E.2 Test Result for J: D <> 0

Ta	z(0.95)	D(P-H)	A	delta	Ho
30	1.96	0.138681	0.135835	0.09	Accept
40	1.96	0.083261	0.081499	0.035	Accept
80	1.96	-0.0627	0.061871	0.04	Accept

From this table, the following conclusion can be drawn.

Ta= 30, the parallel organization is 9% more accurate than the hierarchical organization;

Ta= 40, the parallel organization is 3.5% more accurate than the hierarchical organization;

Ta= 80, the hierarchical organization is 4% more accurate than the parallel organization.

Are these differences significant? The answer to this question depends on a particular application. For example, at $T_a = 30$, the average accuracy (Table E.2) is 0.49 and 0.63 for the hierarchical and the parallel organizations, respectively. However, if the design specification is that the lowest accuracy is 70%, then the divergence between the two structures is outside the range in which the requirement can be satisfied. Therefore, it does not matter whether there is a difference. On the other hand, for another design, the accuracy may be acceptable if it is not lower than 50%. In this case, the parallel organization may be preferred.

Response time

The response time of the organization for two teams is shown in Tables E.3 and E.4. Figures E.4 and E.5 show Tf - Ta plots for these two teams.

Table E.3 Response Time of Team #7

Ta	Tf(Hierarchical)	Tf (Parallel)
30	-	30
40	39.33	38.86
50	43.91	-
60	56.93	57.35
80	63.36	69.56
100	61.03	65.87
120	60.09	69.13

Table E.4 Response Time of Team #10

Hierarchical		Parallel	
Ta	ave T	Ta	ave. T
30	29.86	30	30.00
40	37.76	40	39.70
50	44.31	60	56.51
60	55.51	80	66.74
80	60.34	100	65.48
100	56.97	120	77.43
120	60.74		

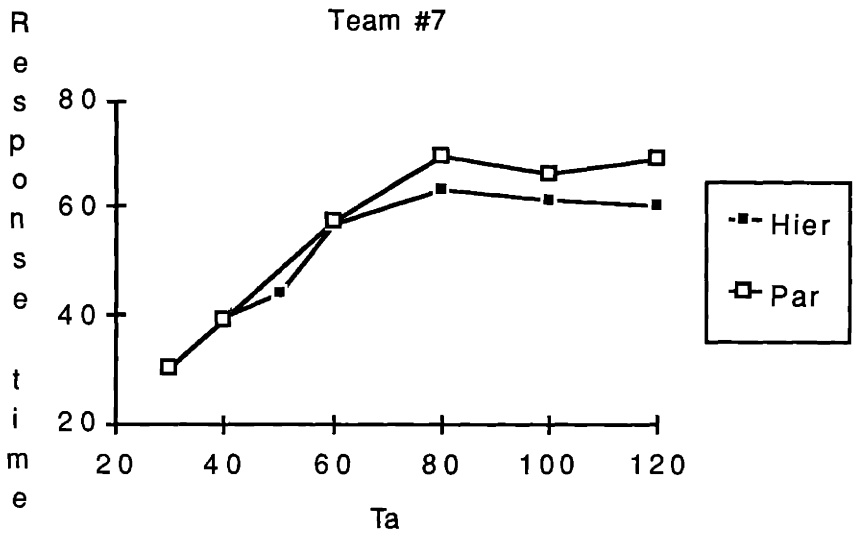


Figure E.4 Response Time versus Available Time for Team #7

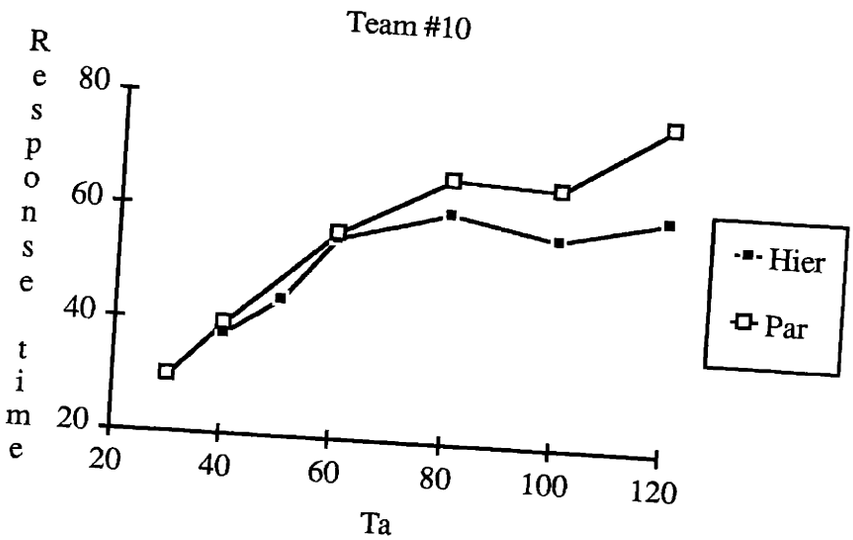


Figure E.5 Response Time versus the Available Time for Team #10

The observations are

1) Response time increases with the available time.
2) At very short available time, the response time is almost equal to the available time. The response time increases slowly when the available time becomes reasonably longer.
3) In the region where the available time is long, the difference on the response time of two structures grows larger.

The alternatives to be tested are

H0:	$D \geq \delta$,	structure x is faster than structure y.
H1:	$D < \delta$	structure x is not faster than structure y.

where $D = T_{fx} - T_{fy}$ and δ is the difference. With 95% confidence level,

two-tail test: $z(1-\alpha/2) = z(0.975)$,

one-tail test: $z(1-\alpha) = z(0.95)$,

where $\alpha = 0.05$.

The test results are shown in Table E.5.

Table E.5a Test Result for T_f: D = 0.

Ta	z(0.975)	s(D)	dD	A1	D(H-P)	H0
30	1.645	0.0393	0	0.0647	0.0487	acc
40	1.645	0.2908	0	0.4784	0.3454	acc
60	1.645	0.5881	0	0.9675	-0.243	acc
80	1.645	1.1367	0	1.8698	-2.344	rej
100	1.645	1.741	0	2.8639	1.8285	acc
120	1.645	1.8401	0	3.0269	-5.176	rej

Table E.5b Test Result for T_f: D <> 0.

Ta	z(0.95)	s(D)	dD	A1	D(P-H)	H0
80	1.96	0.0393	0.2	2.4278	2.3437	acc
120	1.96	0.2908	1	4.6066	5.1761	acc

Table E.5 shows that there are no differences in the response time when the available time is 30, 40, and 60 seconds. When T_a is short, the entire available time is spend by both organizations because the amount of work needed to be done and the bounded rationality constraint require a processing time as long as possible. The difference exists at the longer available time, that is, at T_a = 80 and 120. Just as in the case of accuracy, the difference is small or negligible (T_a = 100 seconds).. However, the small difference implies the natural difference between the two structures. If more time is available to perform the task, the hierarchical organization tends to use longer time because the interaction between the DMs needs longer time than the interaction in the parallel organization.