

**MULTI-AGENT FLIGHT SIMULATION WITH ROBUST
SITUATION GENERATION**

by

Eric N. Johnson

B.S., Aeronautics and Astronautics Engineering
University of Washington, 1991

M.S., Aeronautics
The George Washington University, 1993

Submitted to the Department of
Aeronautics and Astronautics in Partial Fulfillment of the
Requirements for the
Degree of

MASTER OF SCIENCE
in Aeronautics and Astronautics

at the
Massachusetts Institute of Technology
December 1994

© 1994 Massachusetts Institute of Technology
All rights reserved

Signature of Author _____

Department of Aeronautics and Astronautics
December 14, 1994

Certified by _____

Associate Professor R. John Hansman, Jr.
Thesis Supervisor

Accepted by _____

Professor Harold Y. Wachman
Chairman, Department Graduate Committee
Aero

MASSACHUSETTS INSTITUTE
OF TECHNOLOGY

FEB 16 1995

Multi-Agent Flight Simulation with Robust Situation Generation

by Eric N. Johnson

Abstract

A robust situation generation architecture has been developed that generates multi-agent situations for human subjects. An implementation of this architecture was developed to support flight simulation tests of air transport cockpit systems. This system maneuvers pseudo-aircraft relative to the human subject's aircraft, generating specific situations for the subject to respond to. These pseudo-aircraft maneuver within reasonable performance constraints, interact in a realistic manner, and make pre-recorded voice radio communications. Use of this system minimizes the need for human experimenters to control the pseudo-agents and provides consistent interactions between the subject and the pseudo-agents. The achieved robustness of this system to typical variations in the subject's flight path was explored. It was found to successfully generate specific situations within the performance limitations of the subject-aircraft, pseudo-aircraft, and the script used.

Thesis Supervisor:

R. John Hansman, Jr.
Associate Professor Aeronautics and Astronautics

Acknowledgments

Thanks to Amy Pritchett, Koji Asari, Atif Chaudhry, Jim Kuchar, Tom Vaneck, and Prof. John Hansman for the time and effort you gave to this effort. You have been helpful as well as inspirational.

Table of Contents

Abstract	2
Acknowledgments	3
Table of Contents	4
Chapter 1: Introduction	6
Chapter 2: Background	8
2.1 Multi-Agent Simulation	8
2.2 Robust Situation Generation	9
Chapter 3: Robust Situation Generation	11
3.1 Pseudo-Agent Model.....	11
3.2 Robust Situation Generation Approach	12
3.3 Desired Trajectories	13
3.3.1 4D Waypoints	13
3.3.2 Subject Relative Waypoints	14
3.4 Event Plan	16
3.5 Amendments	17
3.5.1 Amendment Cueing	17
3.5.2 Amendment Uses	19
3.5.3 Updating Desired Trajectories and Event Plans	21
3.6 Overall Configuration	22
Chapter 4: Implementation	25
4.1 PLI/TCAS Experiment Specifications	25
4.2 Simulator Setup	26
4.3 Pseudo-Aircraft Modeling	29
4.3.1 Equations of Motion	29
4.3.2 Guidance Model	32
4.3.3 Pseudo-Aircraft Performance Limitations	37
4.3.4 Pseudo-Aircraft Surface Movement	40
4.4 Subject Perception of the Pseudo-Aircraft	40
4.4.1 Voice Communication	41
4.4.2 TCAS Output	43
4.5 Script Writing	47
4.5.1 Experimenter’s Station Tools	47
4.5.2 Pre-Recording Radio Communications	49

Chapter 5: Demonstration	51
5.1 Sample Script	51
5.2 Writing the Demonstration Script	52
Chapter 6: Achieved Robustness	56
6.1 Speed Error	56
6.2 Position Error	58
6.3 Blunder Error	59
6.3.1 Late turn	59
6.3.2 Descending Below Cleared Altitude	60
Chapter 7: Summary and Conclusions	61
References	63
Appendix A: Performance Limit Development	64
A.1 State Limits	64
A.2 Control Limits	67

1. INTRODUCTION

For experiments that involve a human subject guiding a vehicle among other vehicles or agents in a simulated system, there is a need to reliably generate specific situations between the subject and these other agents. A robust situation generation approach to this problem has been developed and is presented here.

A multi-agent simulation mimics an actual system with at least two agents. An agent is defined as a single element in the system, such as an aircraft, a train, or an air traffic controller. If a human is at the controls of an agent in the experiment, then that agent is referred to as a subject-vehicle. All other agents are called pseudo-agents. This system is depicted schematically in Figure 1.

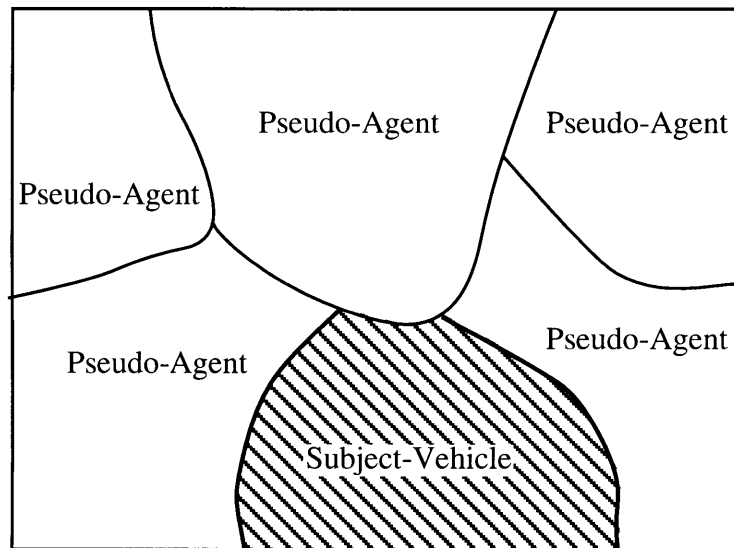


Figure 1 -- A Simulated Multi-Agent System

If the purpose of the experiment is to test subjects' responses to pseudo-agent stimuli, then the experimenter must be able generate specific situations involving the pseudo-agents. This is often difficult because subjects may not act consistently or as expected. Generating these situations in the presence of this uncertainty is referred to as robust situation generation, robust because it must happen within a range of possible actions by the subject.

Chapter 2 provides further motivation and additional background on multi-agent simulation involving human subjects. The scope of this work is also discussed.

The multi-agent simulation architecture is presented in Chapter 3. The architecture is designed to cue individual pseudo-agent actions and generate desired trajectories for a pseudo-agent model to follow. The majority of pseudo-agent actions, including interactions with the subject-vehicle, are scripted in advance.

An implementation of this multi-agent simulation architecture is presented in Chapter 4. The particular multi-agent simulation application is the current Air Traffic Control (ATC) system. The demonstration experiment is a test of the affects of the Traffic alert and Collision Avoidance System (TCAS) and Party-Line Information (PLI) on air-transport pilots' traffic situational awareness. The pseudo-agent models are described in detail. This chapter also includes discussion of how the subject can perceive the pseudo-agents and the impact this has on pseudo-agent model complexity.

An example simulator script is presented in Chapter 5. The experiment is designed to repeatedly generate three situations for human subjects. A discussion of the development of this script further clarifies the robust situation generation architecture.

The achieved robustness of the software architecture in the context of the script presented in Chapter 5 is discussed in Chapter 6. Several parameters of the subject's flight path are varied to determine the limits of this example script.

Chapter 7 is a summary of the multi-agent simulation architecture and the achieved robustness. Discussion about the potential of the architecture for other multi-agent simulation applications is included.

2.0 BACKGROUND

2.1 Multi-Agent Simulation

A multi-agent system consists of at least two agents that are individually engaged in tasks that may require cooperation and/or competition between agents. A multi-agent simulation is a model of such a system.

When an experimenter uses multi-agent simulation to test a human subject, a categorization of the agents is necessary. Humans in the simulated environment directly control subject-agents. The agents that make up the subjects' environment are termed pseudo-agents. Figure 2 illustrates a subject-aircraft as a single subject-agent, and pseudo-aircraft and controllers as the pseudo-agents.

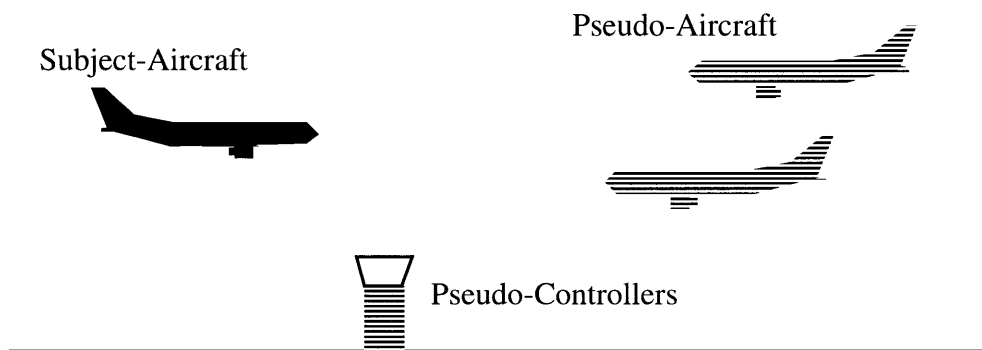


Figure 2 -- A Multi-Agent Simulation Experiment

The condition of any particular agent is defined by its states. A multi-agent simulation must maintain the states of all pseudo-agents. Examples of states are position, velocity, fuel, angular rates, engine thrust, etc. The simulation designer must design a dynamic model for each pseudo-agent, and determine what states are necessary to sufficiently model each one.

Often the purpose of a multi-agent experiment is to test a subject's response to interactions with one or more of the pseudo-agents. These interactions, or situations, consist of the trajectories and/or actions of one or more of the pseudo-agents over some period of time. The experimental situation may not involve all the pseudo-agent in the simulation. Also, the desired situation normally does not fully constrain the trajectories of the pseudo-agents.

In this context, an experimental situation may involve anything from a single pseudo-agent doing a minor action to the complete state make up of every agent in the simulation for a period of time. A script contains a series of situations to be presented to subjects.

Realistic air traffic must be generated when using flight simulation to test aircraft collision avoidance systems such as the Traffic alert and Collision Avoidance System (TCAS). Systems such as TCAS directly affect a subject's relationship to other agents. However, realistic pseudo-agents may also be required when testing systems that indirectly affect this relationship with other agents.

2.2 Robust Situation Generation

If pseudo-agents act autonomously, each with their own goals, knowledge, and processing, then the sequence of interactions between pseudo-agents and the human subject is extremely sensitive to the initial conditions of the simulation and actions by the subject. For example, if a subject-aircraft flies five knots slower than the experimenter had expected over 90 minutes, then the subject-aircraft would be eight nm away from where it was expected to be. Had the desired situation been a collision hazard, a pseudo-aircraft intended to have a near miss might now safely pass as far as 8 nm away. Figure 3 shows both the desired situation (a collision in this case) and the resulting situation if the subject flies slower than expected before the collision is scripted to occur.

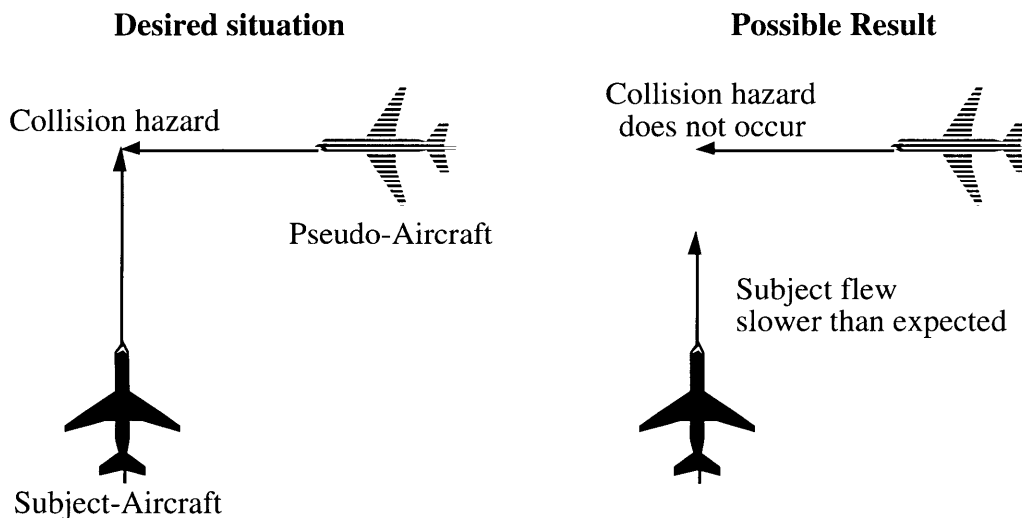


Figure 3 -- Need for Robust Situation Generation

Tolerance to actions of the subject-vehicle, at a fundamental level, implies the need for some kind of feedback from the subject-vehicle. In current air traffic research, this is typically achieved by having the pseudo-aircraft be ‘flown’ by another human, [Bayne]. Another possibility is for the experimenter to change the flight plan of the pseudo-aircraft or subject-aircraft in real time to create specific situations, acting as an air-traffic controller. These options can be quite labor-intensive, and are inherently prone to inconsistent situations; which can confound experiments.

Given the power of computers used for real-time simulation, it has become possible to automate the pseudo-agents and generate specific situations using state feedback from the subject-vehicle. This approach is referred to as *robust* situation generation. The experimenter can, within limits, predetermine the situations that the subject is exposed to. These situations can then be replicated for multiple subjects.

In order for robust situation generation to be effective, pseudo-agents must also appear to maneuver realistically from the subject’s point of view. Behaving realistically has two components. First, the pseudo-agents must individually maneuver within performance constraints. Second, the pseudo-agents must interact with each other properly, in ways that would be normal in the actual system. Since these two problems are matters of appearance, care must be taken only when the subject can perceive the pseudo-agents. The limits of the ability of the pseudo-agents to maneuver result in limits to how robust this type of system can be.

A common part of traditional pseudo-aircraft generation is voice-communications provided by human pseudo-pilots. With robust situation generation, it is possible to digitally record most of these communications ahead of time, and cue these voice calls at the desired points in the experiment. This has the potential to dramatically reduce the labor involved in multi-agent simulation.

The robust situation generation architecture allows specific situations to be created within a range of possible actions of a subject-vehicle in an experiment. These situations are generated within the constraints of realistic trajectories and performance limitations of the agents when necessary. The architecture is of direct application to multi-agent simulations where pseudo-agents need desired trajectories and triggered events in order to create specific situations.

3. ROBUST SITUATION GENERATION

When generating specific situations, the experimenter can control where the pseudo-agents go, and what they do. During simulation, robust situation generation utilizes control over these items to give specific situations to subject-agents. This chapter describes the process.

3.1 Pseudo-Agent Model

A desired trajectory tells pseudo-agents where to go. An event plan tells them what to do and when. The pseudo-agent model maneuvers along its desired trajectory, resulting in a real-time actual trajectory. Also, it executes the event plan. The event plan contains a list of actions at a criterion to cue each action.

This relationship is shown in Figure 4. If the pseudo-agent is a vehicle, this model contains guidance and equations of motion for the vehicle. Otherwise, only the even plan is required.

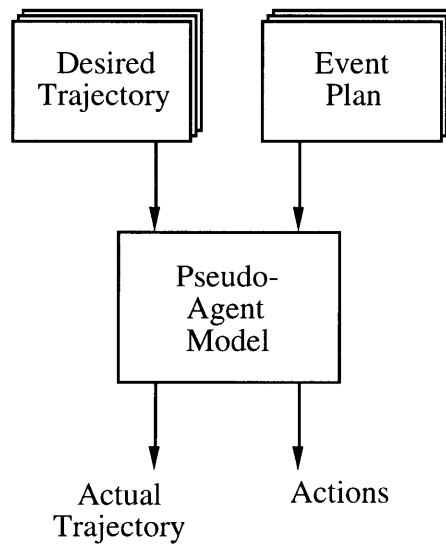


Figure 4 -- Pseudo-Agent Model

Pseudo-agent actions need to be realistic when the subject can perceive them. Maintaining a separate model for each pseudo agent allows the experimenter to ensure these agents individually behave in a realistic manner. For example, a tank will not be able to move at 400 mph, or fly, because they are limited by their models. By using

processing that follows the desired trajectory as closely as possible within performance limits, realistic maneuvering of individual pseudo-agents is assured.

The performance limits of the pseudo-agents are a fundamental limitation to robust situation generation. There will always be some limit to how much subject variation can be handled before one or more of the pseudo-agents can no longer maneuver to where they are needed.

3.2 Robust Situation Generation Approach

To generate specific situations, the pseudo-agents must be given sufficient guidance and system state knowledge to accomplish their roles in the situation. Based on current state information and situations that need to occur, a situation generation controller adjusts the desired trajectories and event plans for the pseudo-agents to follow. For the situation generation to be robust, state information from the subject-vehicle must be utilized in real time.

Figure 5 depicts such a system. Complete state information about the multi-agent simulation, including the subject-vehicle, is used. The pseudo-agents' desired trajectories and event plans are used to generate specific situations for the subject. The pseudo-agents then follow their desired trajectories and event plans.

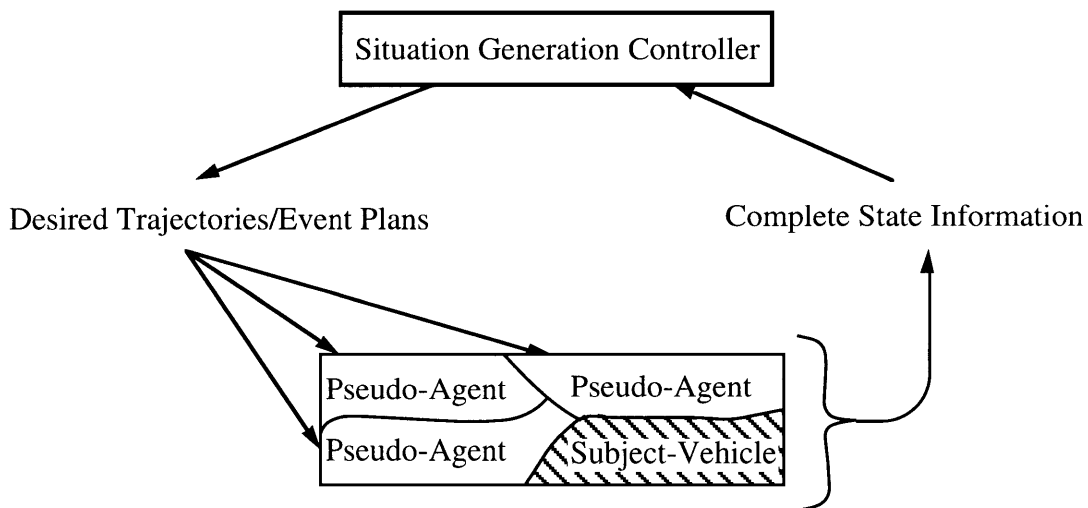


Figure 5 -- Robust Situation Generation System

3.3 Desired Trajectories

Each pseudo-vehicle has a desired trajectory. This desired trajectory can be defined by a list of waypoints. There are many possible desired trajectories, or waypoint types, applicable to robust situation generation. Examples include: traditional position/time waypoints to ‘maintain heading, use pitch and speed to collide with subject’ or ‘stay one mile behind subject’.

The types of waypoints chosen by the experiment designer are highly dependent on the types of pseudo-agents simulated and the situations desired. Both conventional waypoints and waypoints that change position or time based on the state of the subject-vehicle are discussed here.

3.3.1 4D Waypoints

The fundamental type of waypoint used is the 4-dimensional waypoint, referred to here as a 4D waypoint. A 4D waypoint is a position in space plus a desired time to be at that point, as illustrated in Figure 6. A series of 4D waypoints define a desired trajectory in space. The desired trajectory is assumed to be a series of lines at a constant velocity between waypoints. The pseudo-vehicle model provides the necessary transitions (turns, accelerations, etc.) to yield the actual, physically realizable, trajectory.

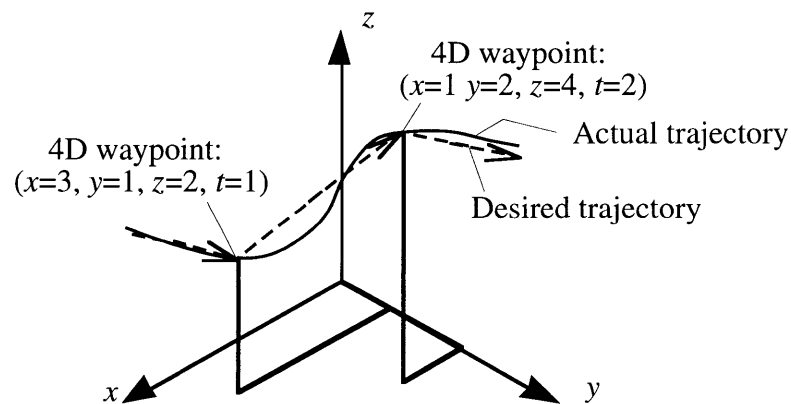


Figure 6 -- 4D Waypoints Defining a Desired Trajectory

3.3.2 Subject Relative Waypoints

One way to use real time subject state feedback is to utilize waypoints that are defined to be relative to the subject-vehicle. Examples of a spatially subject relative waypoint and a time subject relative waypoint are discussed here.

A spatially relative waypoint specifies that the pseudo-aircraft is to be in a position relative to the subject-vehicle at a prescribed time. The use of two such waypoints is illustrated in Figure 7, where the pseudo-agent maneuvers to arrive on a parallel course with the subject-vehicle. In this case, the waypoints specify to be 5 nm South of the subject-vehicle at two moments in time.

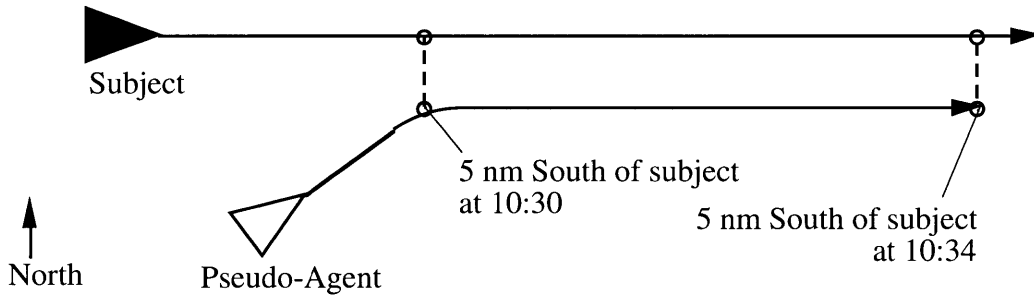


Figure 7 -- Spatially Subject Relative 4D Waypoints

To use subject relative waypoints, a predictor algorithm is needed to predict where the subject-vehicle will be at the 4D waypoint time. A simple technique is to use the current velocity of the subject-vehicle, and then determine the position of the subject-vehicle at the time associated with the 4D waypoint, assuming that the subject-vehicle maintains this velocity,

$$\bar{x}_{predicted} = \bar{x}_{subject} + \bar{\dot{x}}_{subject}(t_{4Dwaypoint} - t). \quad (1)$$

Because these points move spatially as the subject-vehicle changes velocity, more advanced predictor algorithms may be necessary in some applications. For example, if the subject-vehicle initiates a turn, then the predicted point the subject-vehicle will be at will move if Equation 1 is used. Two computationally simple predictor methods that use acceleration information are

$$\bar{x}_{predicted} = \bar{x}_{subject} + \bar{\dot{x}}_{subject}(t_{4Dwaypoint} - t) + \bar{\ddot{x}}_{subject} \frac{(t_{4Dwaypoint} - t)^2}{2},$$

which assumes constant acceleration, and

$$\bar{x}_{predicted} = \bar{x}_{subject} + \left(\dot{\bar{x}}_{subject} + t_{characteristic} \ddot{\bar{x}}_{subject} \right) (t_{4Dwaypoint} - t),$$

where $t_{characteristic}$ is a characteristic length of time assumed for subject-vehicle maneuvers.

Depending on the type of multi-agent simulation experiment, other types of subject-relative waypoints may be useful. For example, rather than having the spatial elements relative to the subject-vehicle, it may be useful for the time element to be relative to the subject-vehicle. A waypoint could use normal spatial points, but use a time that is adjusted so the pseudo-agent maintains a prescribed distance from the subject-vehicle.

Figure 8 illustrates such a waypoint. The first waypoint is a conventional 4D waypoint with a normal desired time. The second waypoint is a prescribed position with a desired time that is adjusted continuously so the pseudo-agent maintains a prescribed distance from the subject-vehicle. This waypoint uses time, and therefore speed, to accomplish the specified range between the two vehicles.

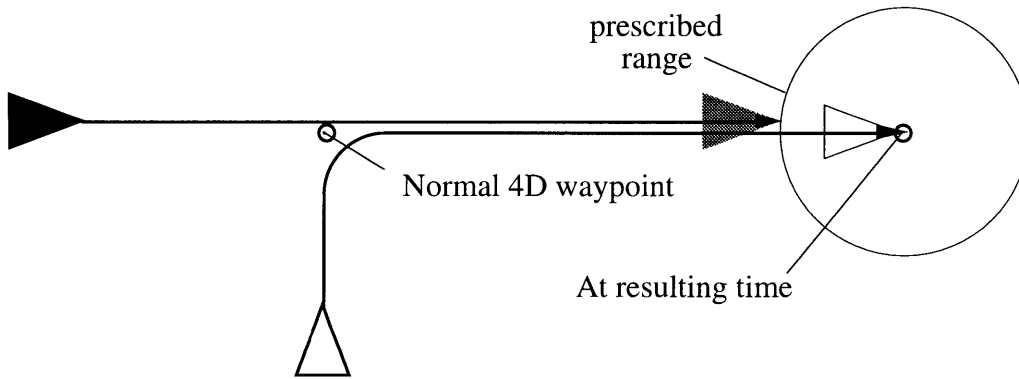


Figure 8 -- Time Subject Relative 4D Waypoint

The exact value used for the time associated with the 4D waypoint is dependent on the type of pseudo-vehicle modeled. If the pseudo-vehicle is capable of generating pure acceleration along its flight path, with first-order lag response with a time constant of τ_V to errors in waypoint arrival time, then

$$t_{4Dwaypoint} = t + \frac{d}{V + \tau_V (\omega_n^2 R_{err} + 2\zeta \omega_n V_{err})}$$

determines the waypoint time, where d is the distance to the next waypoint, V is the current pseudo-vehicle speed, R_{err} and V_{err} are the velocity and range errors, and ω_n and ζ are the frequency and damping ratio of the final response to range errors. Range and velocity errors are found by

$$R_{err} = (R_{desired} - R)\cos(\psi - \psi_{relative})$$

$$V_{err} = V_{subject} \cos(\psi_{subject} - \psi_{relative}) - V \cos(\psi - \psi_{relative})$$

respectively, where $R_{desired}$ is the desired range to the subject, R is the range to the subject, $V_{subject}$ is the subject velocity, $\psi_{subject}$ is the subject heading, ψ is the pseudo-vehicle heading, and $\psi_{relative}$ is the heading to the subject-vehicle.

Clearly, many other possibilities for subject relative 4D waypoints exist. Any particular simulation may require one or more type of 4D subject relative waypoint to accomplish its goals. As illustrated above, subject relative waypoint types range from being completely general to highly vehicle specific.

For some experiments, a problem may arise if the subject-vehicle is able to perceive that a pseudo-agent is continuously tracking the subject-vehicle's movements. For example, if with every speed change the subject-vehicle makes there is a corresponding change in the speed of a particular pseudo-agent, then that pseudo-agent is not behaving realistically. A technique to prevent this problem is to place a sufficiently large time lag on the state information used in predicting the point location. If a time delay is placed on the subject velocity used to predict the future location of the subject, then this lag disappears as the time associated with the 4D waypoint approaches.

3.4 Event Plan

The experimenter often requires the pseudo-agents to do more than just navigate relative to the subject-vehicle. Pseudo-agents that do radio communications, configuration changes, or any of a number of things may need to be a part of the experiment. Anything a pseudo-agent does besides navigate is referred to in this work as an event.

Events are cued by some criteria; such as time, subject-vehicle ETA to a key point, or subject-vehicle location. An event plan is assigned and maintained for each pseudo-agent, Figure 9. From this plan, events are cued and result in discrete actions by

that pseudo-agent. By allowing events to be cued on criteria other than time, increased robustness to varied subject actions is obtained.

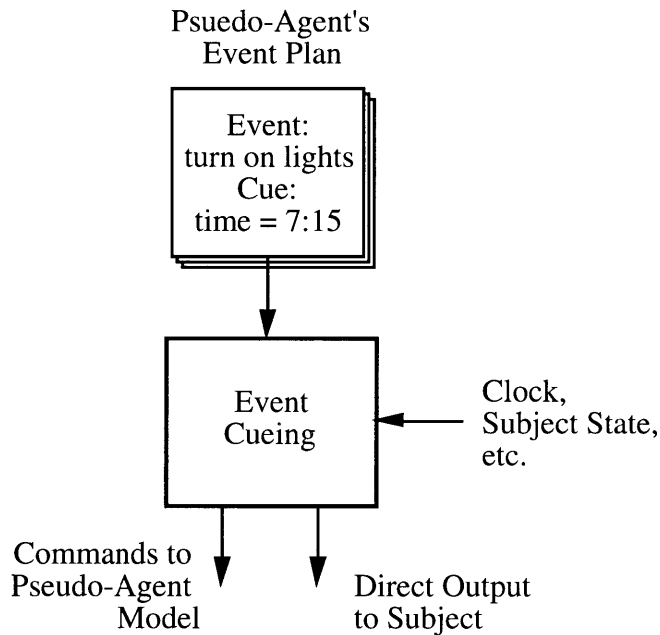


Figure 9 -- Event Cueing, Single Pseudo-Agent

As shown in Figure 9, events can result in either direct output to the subject or commands to the pseudo-agent model or perhaps both. An example of the former would be a radio transmission on a frequency the subject is listening to. Examples of the latter are firing a cannon, lowering landing gear, or turning off lights.

3.5 Amendments

From time to time, the situation generation controller or human experimenter may want to make discrete changes to the desired trajectories and event plans of one or more of the pseudo-agents. In terms of the architecture presented, an amendment contains waypoints and events for one or more of the pseudo-agents in the simulation. The amendment also has a cueing criterion associated with it.

3.5.1 Amendment Cueing

In general, an amendment cueing criterion is a logical expression. An amendment could be scripted to occur when time is greater than 45 seconds, subject speed is less than 200 knots, or some distance is less than 5 nautical miles. The simplest example is a cue

time. That is, a particular amendment occurs when the clock reaches a certain time. The use of other amendment cues will allow for increased robustness.

A very flexible amendment cue is a manual one. This can simply be the experimenter hitting a button to trigger an amendment. A rigid amendment cue is time, because variations in subject actions are not be accounted for. In between these extremes are other aspects of the subject or pseudo-agent state, such as ETA, range, velocity, or others. In addition, amendment cues can also be made conditional on logical operations of multiple cues, such as within a range AND after a certain time.

A particularly useful amendment cue is to wait until the Estimated Time to Arrival (ETA) of the subject-vehicle to a prescribed point in space becomes less than a threshold value. Such an amendment cue allows pseudo-agents to have the proper desired trajectory *before* the subject-vehicle gets to a specific point.

For example, a pseudo-agent that is to arrive and maneuver in front of a subject-vehicle wants to arrive at a point two minutes *before* the subject-vehicle does. This example is depicted in more detail in Figure 10. Here, the amendment cue is ETA at a point in space less than four minutes. In this way, the proper sequencing occurs (subject-vehicle and pseudo-agent two minutes apart) with an accuracy that relates to how much the subject-vehicle changes speed in the two minutes after the amendment cues, instead of speed variations over the entire simulation.

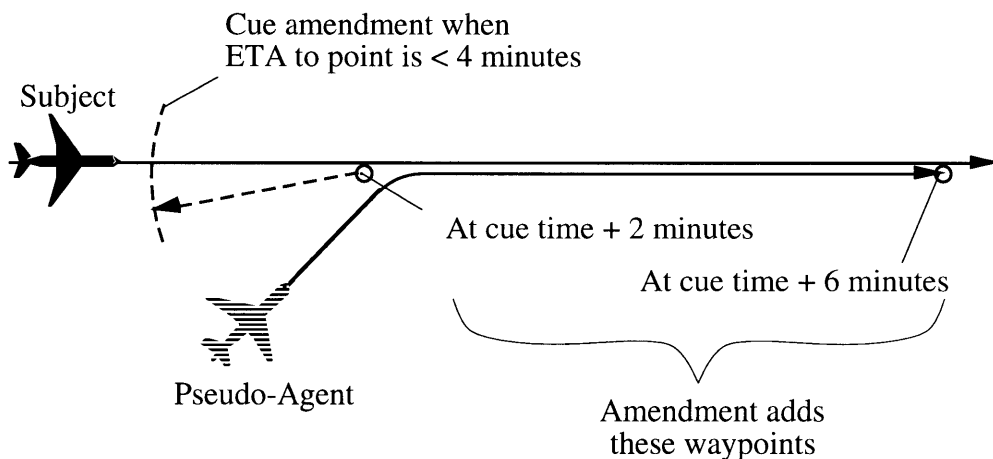


Figure 10 -- ETA Amendment Cue Example Application

One possible way to implement the ETA amendment cue is to divide the distance between the scripted point and the subject-vehicle by the subject-vehicle's speed. The result is a time. If this time is less than the ETA time, the amendment is cued. This is depicted in Figure 11. This amendment will cue even when the subject-vehicle does not maneuver directly through the scripted point, which is desirable in most cases.

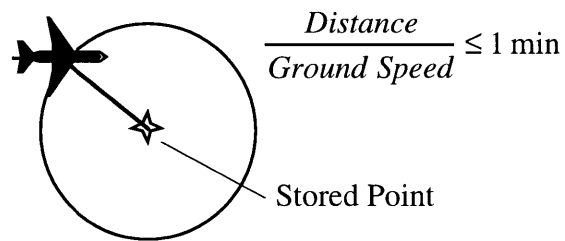


Figure 11 -- ETA Amendment Cue

3.5.2 Amendment Uses

There is a large number of different ways to use amendments and amendment cues to facilitate robust situation generation. Some key examples are illustrated in Figure 12. As the subject-vehicle travels through space, its trajectory will be different than the expected subject trajectory. Amendment can be used to adjust the desired trajectories and event plans of all pseudo-agents based on the arrival time of the subject to an area of space, as shown for the first amendment in Figure 12.

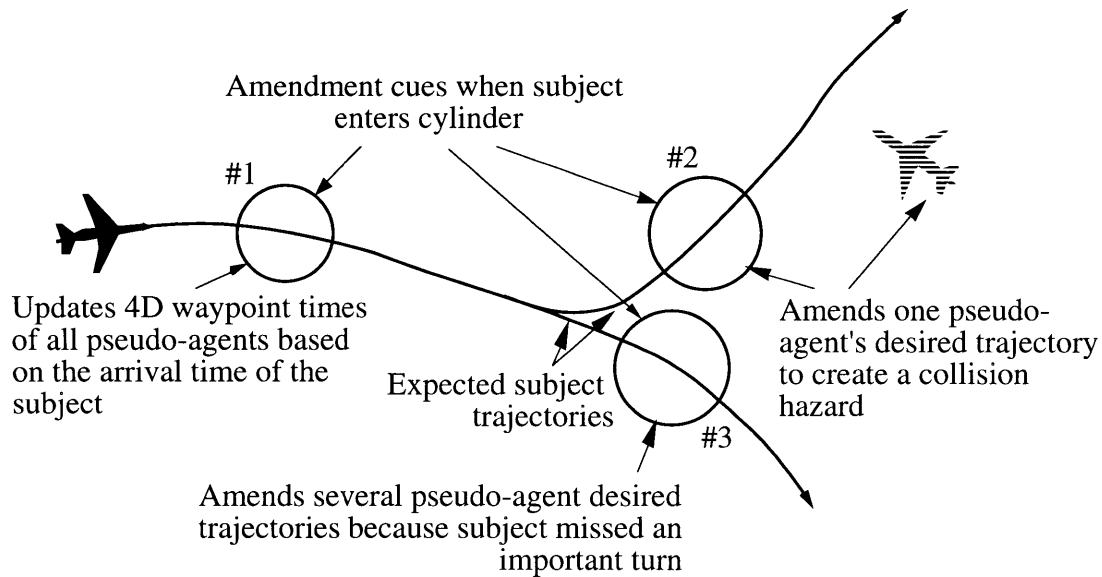


Figure 12 - Amendment Cueing Illustration

A second example of an amendment is the collision hazard situation also shown in Figure 12. This amendment changes the desired trajectory of one of the pseudo-agents so it maneuvers relative to the subject-vehicle. By use of the amendment, this situation will only begin once the subject is in the proper position. If the subject is late or early, this is accounted for. If the subject does not arrive in the proper position at all, then unreasonable attempts by the pseudo-agent to create a collision hazard are avoided.

A third example of an amendment in Figure 12 is a blunder made by the subject. If the experimenter wants to allow the subject to take different discrete paths, or wants to design for different blunders the subject could make, then amendments could be made to account for these discrete variations in the subject path.

In another example, a subject could fly an airplane in an experiment to more than one destination airport, Figure 13. Separate amendments are scripted for each of the two airports. The amendment cues are designed so that the proper response from the pseudo-agents results, whichever on the subject's choice. Perhaps distance from the airport A would trigger an amendment associated with flight to airport A, and similarly for airport B.

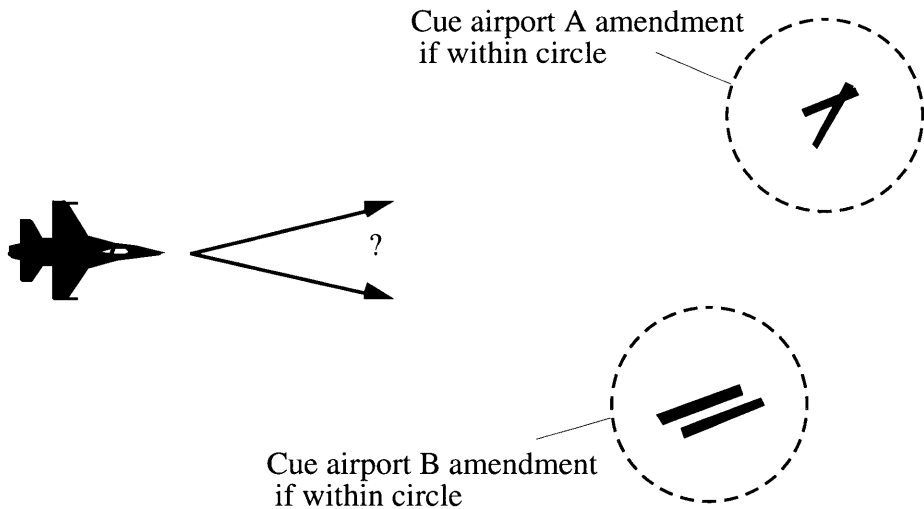


Figure 13 -- Subject-Vehicle Variation Handled with Amendment Cueing

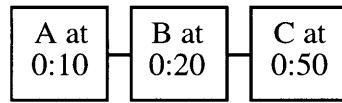
Any pseudo-agent interactions that the subject can perceive must appear realistic. An effective way to avoid unreasonable interactions is to script interacting pseudo-agents simultaneously. It is also necessary to allow amendments to contain desired trajectory updates for these interacting pseudo-agents to preserve the scripted interactions.

3.5.3 Updating Desired Trajectories and Event Plans

The newly cued waypoints must override any conflicting waypoints when a pseudo-agent receives an amendment. When determining where to place the new waypoints in the waypoint list that defines the pseudo-agent's desired trajectory, it is possible to take advantage of the time element of 4D waypoints.

This can be accomplished by deleting all points from the current waypoint list that occur after the first waypoint in the new set, and then adding the new waypoints to the end of the list. This approach is appropriate when the cued actions permanently alter the remainder of the pseudo-agent's trajectory such that it will not return to its former course. This is referred to as waypoint adding. An example is shown in Figure 14. The current set consists of points A, B, and C. The inserted set, D and E, overwrites C because D is to occur before C.

Initial desired trajectory:



Two waypoints added:

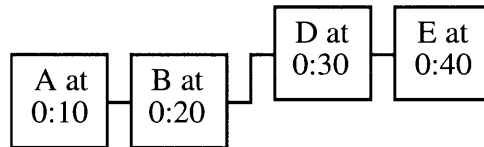


Figure 14 -- Waypoint Adding

When adding or inserting events into a particular pseudo-agent's list, it may be necessary to delete existing events from the event plan. This can be accomplished in the same way as 4D waypoints if the event cue is time. If the event cue is not time, then some other method to delete events may be necessary. In practice, the experimenter can simply make the event cue conditional on the new situation *not* being cued. If the new situation that replaces the event gets cued, the undesired event can no longer be cued.

3.6 Overall Configuration

Two fundamentally different uses are made of subject state feedback in this work. First, amendments and events can be cued based on subject state. Event cues allow pseudo-agent actions to occur at the proper time even when the proper time depends on what the subject does. This increases the robustness of the system.

A second, fundamentally different, use of subject state feedback is real time adjustment of a pseudo-agent's desired trajectory by use of subject relative waypoints. A simple example is maneuvering a pseudo-agent that needs to collide with a constantly maneuvering subject-vehicle. However, real-time adjustment of the desired trajectory is also useful for generating many other specific interactions with the subject-vehicle.

A block diagram of a system that allows subject-relative trajectories and event cueing is shown in Figure 15. This system maintains a desired trajectory and event plan that are interpreted by a pseudo-agent model. This happens for all pseudo-agents in the simulation. The output of these models is depicted to the human subject through some display or displays. The subject responds, and affects the trajectory of the subject-

vehicle. The state of the subject-vehicle is then used to allow subject-relative trajectories and event cueing.

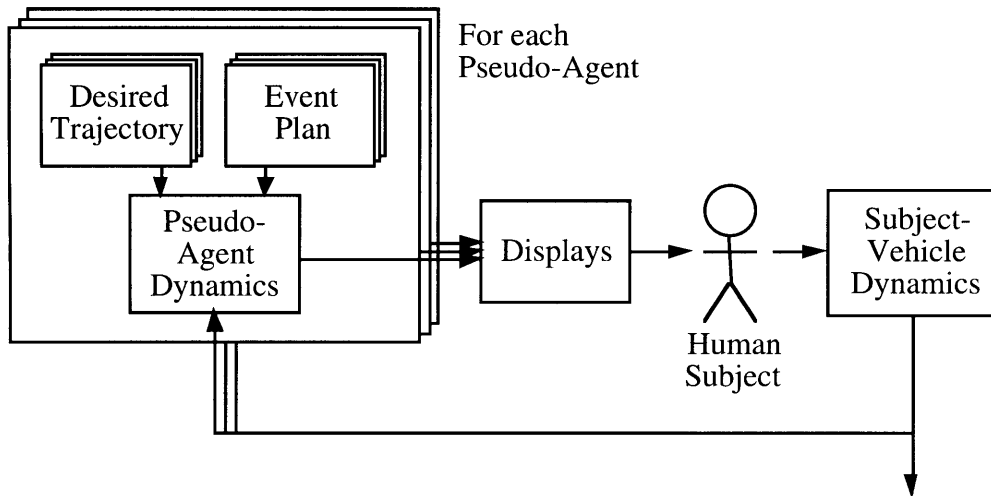


Figure 15 -- Subject Relative Trajectories and Event Cueing Structure

The final robust situation generation architecture must include amendments and amendment cueing, and is depicted in Figure 16. Each amendment contains a desired trajectory update and/or event plan updates for one or more of the pseudo-agents. It also contains a cueing criterion. This cueing criterion can be based on subject-vehicle state or can be triggered by an experimenter. Cueing amendments by subject state adds robustness to the simulation by allowing the experimenter to design in advance for different discrete “paths” the subject may take. Also, cued amendments can adjust several interacting pseudo-agents simultaneously. In addition to subject state, amendments can also be cued manually by an experimenter.

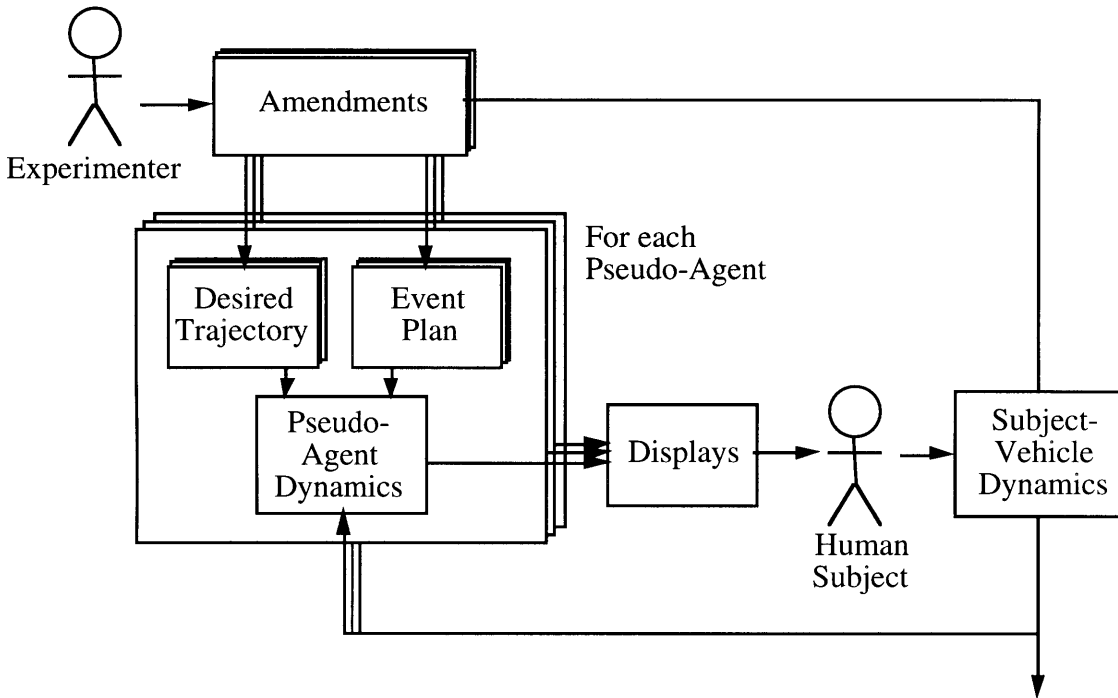


Figure 16 -- Overall Configuration

In summary, an amendment is cued when its cueing criterion is met. When an amendment is cued, waypoints are added or inserted to one or more of the pseudo-vehicles' waypoint lists (desired trajectories), and events are added to one or more of the pseudo-agents' event plans.

The pseudo-vehicle model maneuvers through its waypoint list, but only within the performance limitations of the vehicle model. For robustness, the waypoint list can contain subject relative waypoints.

Each of the pseudo-agents also has an event plan containing actions to be taken and the cue to trigger the action. These actions may provide output to the subject, such as honking a horn; or effecting the pseudo-agent model, such as shutting down an engine. Like amendments, events can be cued by subject actions, time, or by experimenter actuation.

4. IMPLEMENTATION

The design of a multi-agent simulation that relies on robust situation generation serves to validate and clarify the software architecture presented. Such an implementation is described in this chapter. A variety of important operational details that arose when creating this type of simulation are also discussed.

4.1 PLI/TCAS Experiment Specifications

Today, voice communications between aircraft and Air Traffic Control (ATC) are on frequencies shared with one or more aircraft. Because pilots overhear communications between their controller and other aircraft using the same frequency, they receive valuable 'party line' information (PLI). This PLI contains information about weather deviations, sequencing, aids traffic avoidance, and others. Proposed implementations of datalink communication between air traffic controllers and aircraft, and between aircraft, will likely result in PLI loss. Unfortunately, in some cases the PLI is considered of critical importance by pilots [Midkiff].

The Terminal alert and Collision Avoidance System (TCAS) includes a display of the relative positions of nearby aircraft. By presenting this information, TCAS provides considerable information about nearby traffic previously only available from the PLI.

The ability of TCAS displays to compensate for PLI loss and/or enhance PLI is to be studied utilizing an implementation of the robust situation generation architecture. Specifically, responses to specific situations are compared while varying the sources of information available: PLI only, TCAS only, and PLI with TCAS. Pseudo-aircraft traffic situations are generated to test pilot responses.

Several different types of pseudo-aircraft traffic situations are required during each flight. These include collision and near collision situations, traffic advisory situations, and other traffic related situations. Traffic advisory situations are those that cause the TCAS system to generate a cautionary Traffic Advisory (TA) when the encounter is not yet severe enough to generate evasive maneuvers. Other traffic related situations include expected sequencing around weather, expected holding, and others.

The pseudo-aircraft also make realistic radio transmissions, some of which are critical to the experiment. There are also radio transmissions from controllers to the

pseudo-aircraft. Realistic state information is maintained for all pseudo-aircraft that show up on the TCAS display, which has a nominal range of 40 nm (the display capabilities of the TCAS system simulated are discussed in detail in Section 4.4.2).

In addition, the multi-agent simulation was designed to handle other, more advanced traffic displays beyond the current TCAS displays, which for the most part only gives estimated position information. These enhanced traffic displays might show pseudo-aircraft heading, airspeed, turn rate, and vertical speed, or even ATC clearances, in addition to the position information. Enhanced traffic displays are expected to come with the appearance of new generations of TCAS or other datalink systems, and will need to be designed and tested with multi-agent simulation.

4.2 Simulator Setup

The simulation experiments were conducted on the Aeronautical Systems Laboratory (ASL) Advanced Cockpit Simulator (ACS). The simulator is centered around an SGI INDIGO, used to integrate the subject aircraft's dynamics and provide the desired displays. The simulator also provides a Control Display Unit (CDU), Mode Control Panel (MCP), sidestick, and throttle quadrant, shown in Figure 17.

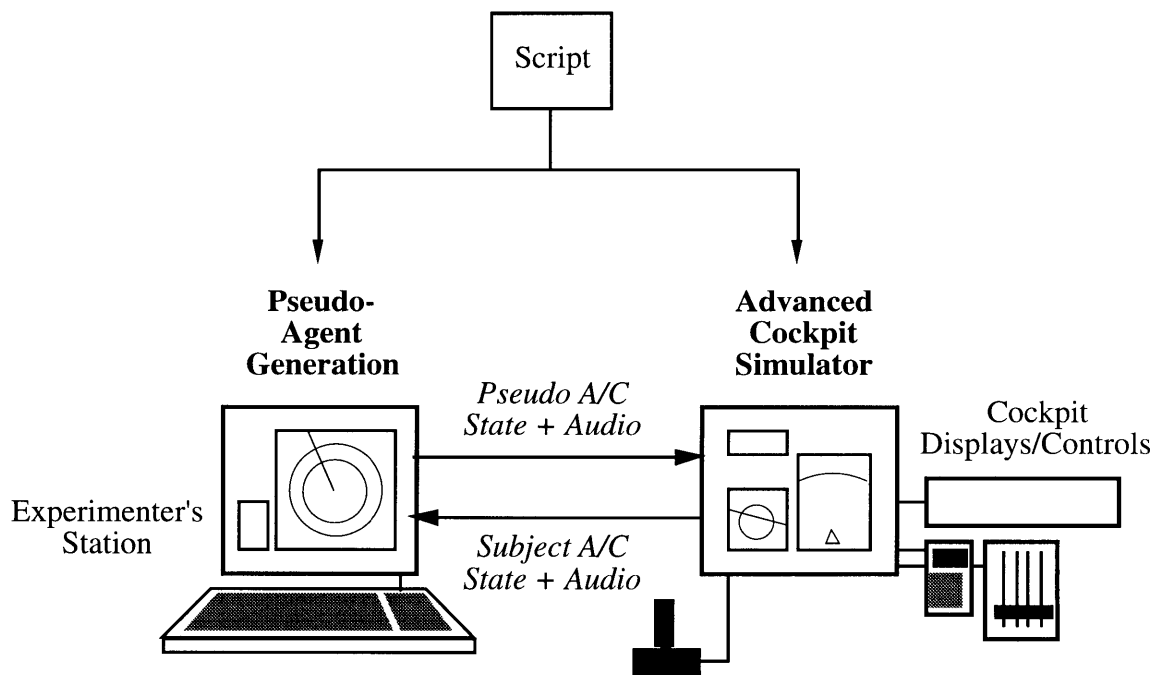


Figure 17 -- Simulator Setup



Figure 19 -- Experimenter's Display Close-up

The experimenter's display is an electronic map of an area. The scale and location of the center can be adjusted. All aircraft, including the subject-aircraft, are shown as symbols at their proper locations. Pseudo-aircraft are shown as a white symbol, the subject-aircraft orange. The name and current altitude of each aircraft is shown next to its symbol. A trend line, often referred to as a 'noodle', is drawn out the front of each aircraft symbol. Its length is the distance that the aircraft will travel in the next 30 seconds if the pseudo-aircraft's ground speed remains constant. If the aircraft is turning, then the line becomes an arc with the radius that the aircraft is turning around.

Airports, navigation fixes, intersections, and radio navigation aids are shown in white on this map. The display can be de-cluttered by turning off any or all of the text on the display, as well as the navigation symbols.

In a separate window on the screen, a variety of information about the subject-aircraft is displayed, including: TAS, vertical speed, heading, altitude, commanded speed, commanded vertical speed, commanded heading, and commanded altitude. This

information is received from the ACS and is shown as a reference for the experimenter. Displaying this information is also important for script writing, discussed in Section 4.5.

TAS, vertical speed, heading, and altitude can be displayed for a pseudo-aircraft in another window by clicking the mouse button with the cursor on the aircraft symbol. In addition, the previous, next, and the following 4D waypoint that the pseudo-aircraft is flying through can be shown as purple symbols with connecting lines on the experimenter's display, along with the waypoint altitudes as text, if desired. This feature is also specifically important for script writing.

4.3 Pseudo-Aircraft Modeling

Multi-agent simulation relies on realistic modeling of the pseudo-aircraft. This includes equations of motion, performance limitations, and a guidance model. The specifications of the PLI/TCAS experiment and possible enhanced traffic displays directly determine the fidelity and accuracy required of this model.

4.3.1 Equations of Motion

The requirements for the TCAS display imply that pseudo-aircraft states must result in realistic update of latitude, longitude, and altitude. Many proposed enhanced traffic displays, as well as useful TCAS alerts, imply that airspeed, vertical speed or Flight Path Angle (FPA), and heading should also be states.

Because the experiment does not require any information about pseudo-aircraft conventional control locations (aileron, elevator, rudder) or real aircraft dynamics, a rather simple aircraft model can be used. It is possible to use only the above six states plus roll angle, and use airspeed rate-of-change, roll rate-of-change, and FPA rate-of-change as the controls. These choices for states and controls are summarized in Table 1. In essence, this model assumes the modeled pilot does what is necessary with aileron, elevator, flaps, etc. to achieve the desired state rates. Roll angle was added to provide realistic heading rate changes if pseudo-aircraft turns are depicted on an enhanced traffic display. Performance limitations are placed on the controls and states as a function of pseudo-aircraft state (e.g. altitude, airspeed), discussed further in Section 4.3.3.

state	symbol	unit	control	symbol	unit
Ground Speed	V	knots	Ground Speed Rate	\dot{V}	knots/sec
Roll Angle	ϕ	rad	Roll Rate	$\dot{\phi}$	rad/sec
Flight path angle relative to air	γ	rad	Flight path angle Rate	$\dot{\gamma}$	rad/sec
True Heading	ψ	rad			
Latitude	μ	rad			
Longitude	l	rad			
Altitude	h	feet			

Table 1 --Pseudo-Aircraft States and Controls

It is desired to have a pseudo-aircraft that could fly through a complex wind pattern, such as a microburst, and behave appropriately. This is accomplished by the unusual combination of states, namely ground speed, heading, and an airmass relative flight path angle. By using these as the state definitions, a wind velocity vector can be made spatially and time dependent. The pseudo-aircraft behave appropriately by changing crab angles and adjusting true airspeed to changes in wind.

The first step in updating the states is determining true airspeed (V_{TAS}) and the actual course and flight path angle, ψ_{track} , and γ_{track} ,

$$V_{TAS} = V - u_{wind} \cos \psi - v_{wind} \sin \psi,$$

$$\psi_{track} = \psi + \frac{u_{wind} \sin \psi + v_{wind} \cos \psi}{V},$$

$$\gamma_{track} = \gamma - \frac{w_{wind}}{V}.$$

These calculations assume small wind correction angle, where u_{wind} , v_{wind} , and w_{wind} are the North, East, and down wind velocity components respectively. In general, these wind velocity components can be time and or spatially dependent.

The chosen combination of states and controls yield simple equations of motion. Airspeed, roll angle, and FPA change is directly controlled. Heading is governed by the equation

$$\dot{\psi} = \frac{g}{V_{TAS}} \tan \phi \quad (2)$$

where g is the local acceleration of gravity. Position can be governed by

$$\dot{\mu} = \frac{V}{R_{Earth}} \cos \psi_{track}, \quad (3)$$

$$\dot{l} = \frac{V}{R_{Earth}} \sin \psi_{track} \cos \mu, \text{ and} \quad (4)$$

$$\dot{h} = V \tan \gamma_{track} \quad (5)$$

where R_{Earth} is the distance from the pseudo-aircraft to the center of the Earth. For this implementation, a constant value (e.g. $2.0888 \times 10^7 \text{ ft}$) achieves the desired accuracy. These equations assume a spherical Earth and are unacceptable close to the Earth's poles, but are appropriate for this application due to their simplicity and achieved accuracy.

Because the pseudo-aircraft fly almost exclusively at a constant velocity, a first order numerical integration scheme is adequate for updating the states based on the controls and Equations 2 through 5. The state vector,

$$\bar{x}_s \equiv \{V \ \phi \ \gamma \ \psi \ \mu \ l \ h\}^T,$$

with knowledge of

$$\bar{\dot{x}}_s \equiv \{\dot{V} \ \dot{\phi} \ \dot{\gamma} \ \dot{\psi} \ \dot{\mu} \ \dot{l} \ \dot{h}\}^T$$

is updated using

$$\bar{x}_s^* = \bar{x}_s + dt \cdot \bar{\dot{x}}_s.$$

where dt is the length of time being integrated. First order integration achieves desired accuracy so long as velocities are largely constant and the time step is small. If greater accuracy is required, then a smaller minimum time step or a higher order integration method should be used [Burden].

4.3.2 Guidance Model

The guidance model, shown in Figure 20, contains the specific control functions that allow the pseudo-aircraft to fly to a specific waypoint and to arrive at that point at a prescribed time. This must be accomplished within the performance limitations of the vehicle model and rely on maneuvers that would realistically be seen in the air traffic environment. This guidance model consists of four elements: waypoint placement, 2D (latitude and longitude) waypoint capture, altitude capture, and time capture. Together, these elements transform a pseudo-aircraft's state, desired trajectory, and the subject-aircraft state into state rate commands.

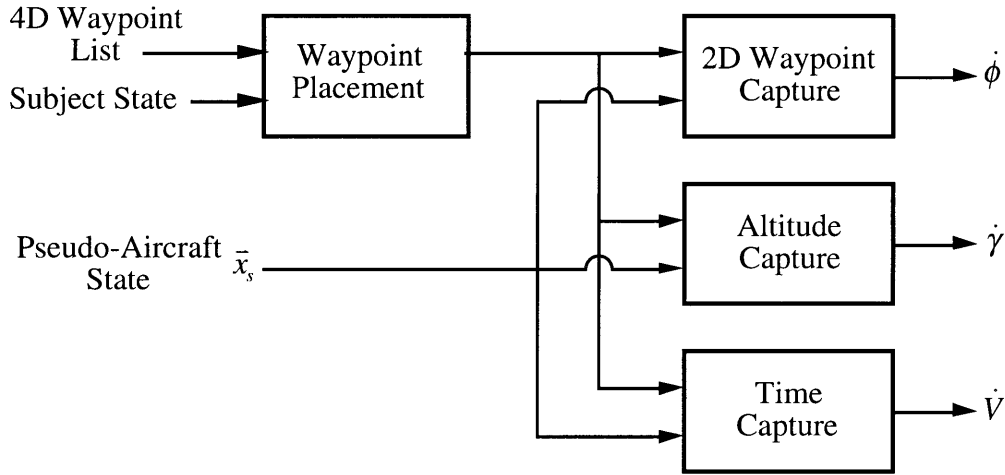


Figure 20 -- Pseudo Aircraft Guidance Model

The first essential element in the system is referred to as waypoint placement. Waypoint placement converts subject-aircraft relative waypoints into conventional waypoints. This is fundamentally a problem of predicting the location of the waypoint at the time associated with the 4D waypoint, as was discussed in Section 3.3.2. For this work, the spatial location of a subject-aircraft relative waypoint was determined using

$$\text{latitude: } \mu_{4Dwp} = \mu_{subject} + \frac{\left[x_{relative} + (t_{4Dwp} - t) V_{subject-TAS} \cos \psi_{subject-track} \right]}{r_{Earth}},$$

longitude:

$$l_{4Dwp} = l_{subject} + \frac{\left[y_{relative} + (t_{4Dwp} - t) V_{subject-TAS} \sin \psi_{subject-track} \cos \mu_{subject} \right]}{r_{Earth}},$$

altitude:
$$h_{4Dwp} = h_{relative} + h_{subject} + (t_{4Dwp} - t)V_{subject-TAS} \sin \gamma_{subject-track}$$

where $x_{relative}$, $y_{relative}$, and $z_{relative}$ is the desired subject-aircraft relative position vector. At any particular time, this conversion is made on the previous 4D waypoint, the next waypoint, and the following waypoint, for a total of three, illustrated in Figure 21.

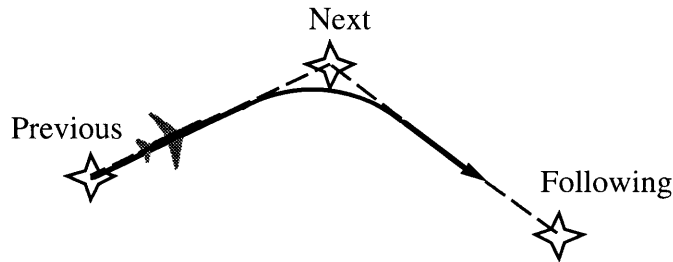


Figure 21 -- Definition of Previous, Next, and Following Waypoints

Capturing the 4D waypoint is de-coupled to form three elements: 2D waypoint capture, altitude capture, and time capture. The 2D waypoint capture refers to other lateral-directional dynamics of the vehicle and places the pseudo-aircraft above a point on the Earth's surface. Altitude and time captures refer to arriving at this point at the proper altitude and time respectively, and utilize the longitudinal dynamics of the pseudo-aircraft.

The 2D waypoint capture is accomplished by pointing the pseudo-aircraft velocity vector at proper point on the map. This must be accomplished with the single lateral-direction control, bank angle rate or $\dot{\phi}$. This is accomplished in three steps: the determination of a desired heading rate, finding the bank angle that would achieve this rate, and then commanding a bank angle rate to achieve desired bank angle. The symbols used are illustrated in Figure 22.

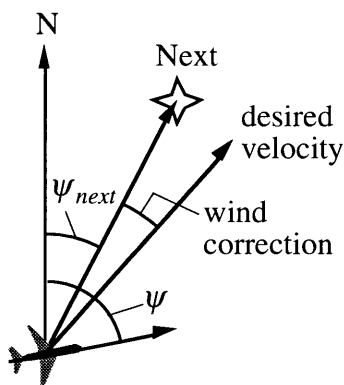


Figure 22 -- Symbols Used for 2D Waypoint Capture

Standard rate turns are used. When below 250 kts TAS, two minutes for a 360° turn, or 3° per second, is used. Above 250 kts, four minutes is used for a turn, or 1.5° per second. Determining the angle between the pseudo-aircraft's ground track and the next waypoint location is the first step in 2D waypoint capture. A desired heading rate is then determined based on this error,

$$\dot{\psi}_{desired} = G_{\psi} \left[\psi_{next} - \frac{u_{wind} \sin \psi_{next} + v_{wind} \cos \psi_{next}}{V} - \psi \right]$$

where G_{ψ} is a gain, relating a heading error to a desired heading rate. The values of all guidance model gains are given in Table 2. If the result is of greater magnitude than the standard rate, the standard rate is used. The performance limitations ensure that excess control power or bank angle is not used when correcting large errors.

Gain	Symbol	Value	Time Constant
Heading error to heading rate	G_{ψ}	0.71/sec	1.4 sec
Bank angle error to bank angle rate	G_{ϕ}	1.4/sec	0.71 sec
Flight path angle error to flight path angle rate	G_{γ}	1.0/sec	1.0 sec
Speed error to speed rate	G_V	1.0/sec	1.0 sec

Table 2 -- Guidance Model Gains

The desired bank angle is calculated directly based on this desired heading rate,

$$\phi_{desired} = \frac{V_{TAS}}{g} \dot{\psi}_{desired}$$

This is based on Equation 2 with the approximations that $\tan \phi \cong \phi$ and that $\dot{\gamma}$ is small. Finally, $\dot{\phi}$, the lateral-directional control, is determined by

$$\dot{\phi} = G_{\phi}(\phi_{desired} - \phi)$$

where G_{ϕ} is a gain with value listed in Table 2. The gains for heading and bank error were chosen to result in a second order system of heading command to heading with frequency of 1 radian per second and a damping ratio of 0.7.

Altitude capture is accomplished with the use of $\dot{\gamma}$, or flight path angle changes. A desired flight path angle is found that would put the pseudo-aircraft at the correct altitude when it passed the waypoint 2D position assuming it is traveling directly at the 2D position,

$$\gamma_{desired} = \arctan\left(\frac{h_{4Dwp} - h}{d_{next}}\right) + \frac{w_{wind}}{V}.$$

where d_{next} is the distance to the waypoint to be captured. A $\dot{\gamma}$ is then found to capture the desired flight path angle

$$\dot{\gamma} = G_{\gamma}(\gamma_{desired} - \gamma).$$

where G_{γ} is a gain whose value is listed in Table 2. The acceleration necessary to change flight path angle is equal to $V\dot{\gamma}$. If the magnitude of $\dot{\gamma}$ is such that it imparts an acceleration of more than 0.3 times the acceleration of gravity, then $\dot{\gamma}$ is clipped such to limit acceleration to this value.

The final element of the guidance model determines acceleration necessary to arrive at the next waypoint at the scripted time. First, the desired velocity for flight between the current position and the 4D waypoint is determined,

$$V_{desired} = \frac{d}{t_{4Dwp} - t},$$

assuming that the current velocity is used and our ground track is directly toward the waypoint. Finally, the acceleration is found,

$$\dot{V} = G_V(V_{desired} - V),$$

where G_V is a gain with value listed in Table 2.

The transition from flight toward the next 4D waypoint to the following 4D waypoint is a critical period for the pseudo-aircraft guidance model. Real aircraft do not fly directly through a waypoint, and then afterwards correct course to meet the one after. Instead, they smoothly transition from one linear trajectory to another, Figure 23. This is accomplished by anticipating the course changes necessary to capture the next linear trajectory, and beginning the course change before reaching the waypoint.

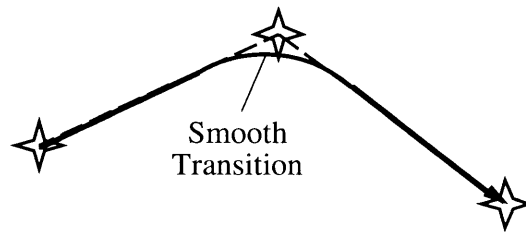


Figure 23 -- Smooth Transition Between Linear Trajectories

Specifically, anticipating course changes was accomplished by triggering the capture of the following waypoint when a transition is determined to be required. This determination is made assuming the standard rate turn and a standard rate pull/push over. The following waypoint becomes the next waypoint when a turn is needed, a push/pull over is needed, or the 4D waypoint time has expired, whichever comes first.

The turn anticipation point is determined by first computing the heading difference, ζ , between the two linear trajectories

$$\zeta = \psi_{previous/next} - \psi_{next/following}$$

with a check to ensure this angle is less than or equal to 180° . If not, a turn in the opposite direction is appropriate so $360^\circ - \zeta$ should be used. Next, the distance prior to reaching the next waypoint that a standard rate turn should begin is found by

$$d_{turn} = \frac{V}{\dot{\psi}_{standard}} \tan \frac{\zeta}{2},$$

assuming a constant ground speed is nearly maintained. A standard rate, $\dot{\psi}_{standard}$, of 3° per second is used when true airspeed is less than 250 kts, otherwise, a value of 1.5° per second is used. Once the aircraft reaches this distance from the waypoint, the turn is initiated and the waypoint is considered past.

Pull/push over anticipation is determined in a similar manner. The push/pull over is assumed to be 0.3 times the local acceleration of gravity. This anticipation distance is found by

$$d_{pull/push} = \frac{V^2}{0.3g} \tan \left| \frac{\gamma_{previous/next} - \gamma_{next/following}}{2} \right|.$$

This distance is compared to the current distance to the next waypoint. If the current distance is the lesser, the waypoint is considered past.

4.3.3 Pseudo-Aircraft Performance Limitations

As previously discussed, performance limitations were applied to both the controls and to the states. These limitations, in addition to the equations of motion used, ensure the pseudo-aircraft individually behave in a realistic manner.

All pseudo-aircraft modeled are subsonic, transport category airplanes with high-bypass turbofan engines. Different aircraft types are simulated through database of aircraft performance parameters, Table 3. The parameters are then used in a generic performance limits structure for all pseudo-aircraft. The exact type of any particular pseudo-aircraft is set in the script. Performance limitations are placed on the states and controls as depicted in Figure 24.

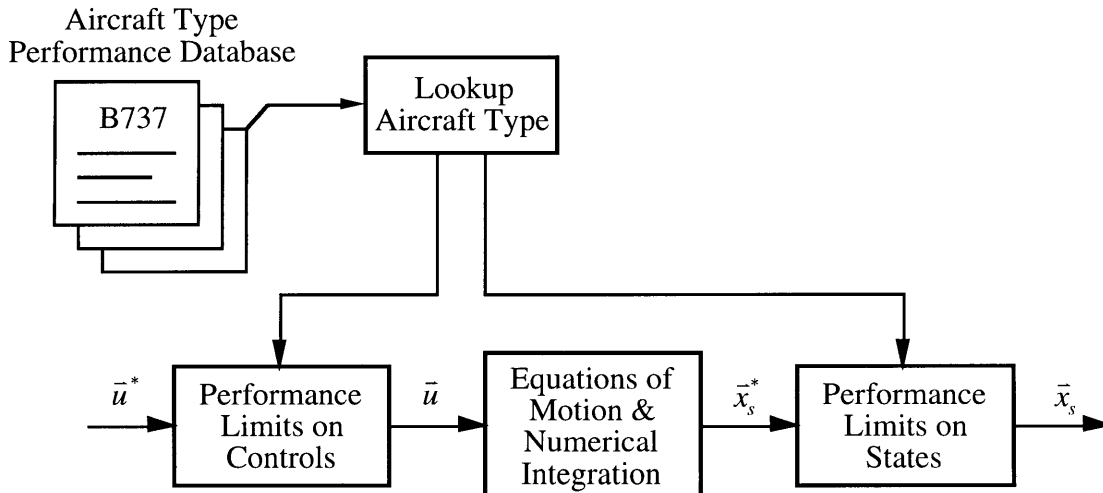


Figure 24 -- Performance Limitations

parameter	description
W	total weight
S	reference wing area
$T_{\max}(0)$	maximum sea level thrust
T_{\min}	minimum thrust
M_{cruise}	maximum cruise Mach number
$c_{L\max}$	maximum lift coefficient
$c_{L\min}$	minimum lift coefficient
c_{Do}	minimum drag coefficient
$c_{D\delta}$	incremental configuration drag coefficient
k	induced drag parameter
n_{\max}	maximum structural load factor
n_{\min}	minimum structural load factor
$ \dot{\phi} _{\max} / V$	maximum absolute roll rate divided by reference airspeed
$ \phi _{\max}$	maximum absolute roll angle

Table 3 -- Performance Model Parameters

The performance limitations, in general, need only be active for a particular pseudo-aircraft when that vehicle is on the TCAS display. Otherwise, the experimenter can allow the performance limitations to be relaxed or ignored. This has the potential to improve situation robustness. In practice the benefit is small due to the range of the TCAS display.

Because ATC normally expects pilots to adhere to altitude clearances with priority over speed clearances, the performance limitations used for the pseudo-aircraft have an altitude priority. That is, the pseudo-aircraft try to capture the desired flight path angle first, and capture speed only if the desired flight path angle is captured and sufficient margin remains. A summary of the performance limitations is listed in Table 4.

		lower bound	upper bound
states	V	stall speed & bottom of power curve based on current altitude	regulations & cruise mach number based on current altitude
	γ	best attainable in steady state at current altitude within speed limits	best attainable in steady state at current altitude within speed limits
	ϕ	arbitrary	arbitrary
controls	\dot{V}	best available given altitude, airspeed, and flight path angle	best available given altitude, airspeed, and flight path angle
	$\dot{\gamma}$	lift coefficient & structural load factor	lift coefficient & structural load factor
	ϕ	best available at current airspeed	best available at current airspeed

Table 4 -- Performance Limitations Summary

The equations used to determine these performance limitations are summarized in the following tables, and developed in Appendix A. Table 5 shows the equations for the lower bound on each state and control. Table 6 shows the upper bound on each state and control. No limits were placed on states or controls not indicated in these tables.

		lower bound
states		$V_{\min}(h) = \sqrt{\frac{2W}{\rho(h)S} \max\left(\frac{1}{c_{L\max}}, \sqrt{\frac{k}{c_{D0}}}\right)} - V_{TAS} + V$
		$\gamma_{\min}(h) = \frac{T_{\min} - (c_{D0} + c_{D\delta})\frac{1}{2}\rho(h)V_{TAS\max}^2S}{W} - \frac{kW}{\frac{1}{2}\rho(h)V_{TAS\max}^2S}$
		$\phi_{\min} = -30^\circ$
controls		$\dot{V}_{\min}(h, V_{TAS}, \gamma) = \left[\frac{T_{\min} - (c_{D0} + c_{D\delta})\frac{1}{2}\rho(h)V_{TAS}^2S}{W} - \frac{kW}{\frac{1}{2}\rho(h)V_{TAS}^2S} - \gamma \right] g$
		$\dot{\gamma}_{\min}(h, V_{TAS}) = \frac{g}{V} \max\left\{ n_{\min}, \left[\frac{c_{L\min} \frac{1}{2}\rho(h)V_{TAS}^2S}{W} - 1 \right] \right\}$
		$\dot{\phi}_{\min}(V_{TAS}) = -V_{TAS} \cdot \left(\frac{c_{l\delta_A} \delta_A}{\frac{b}{2} c_{l_p}} \right)_{\max}$

Table 5 -- Lower Performance Limitations Equations

upper bound	
states	$V_{\max}(h h \leq 10,000 \text{ feet}) = \sqrt{\sigma(h)} \cdot 250 \text{ knots} - V_{TAS} + V, \text{ and}$ $V_{\max}(h h > 10,000 \text{ feet}) = M_{\text{cruise}} a(h) - V_{TAS} + V$
	$\gamma_{\max}(h) = \frac{T_{\max}(h) - c_{Do} \frac{1}{2} \rho(h) V_{TAS \min}^2 S}{W} - \frac{kW}{\frac{1}{2} \rho(h) V_{TAS \min}^2 S}$
	$\phi_{\max} = 30^\circ$
controls	$\dot{V}_{\max}(h, V_{TAS}, \gamma) = \left[\frac{T_{\max}(h) - c_{Do} \frac{1}{2} \rho(h) V_{TAS}^2 S}{W} - \frac{kW}{\frac{1}{2} \rho(h) V_{TAS}^2 S} - \gamma \right] g$
	$\dot{\gamma}_{\max}(h, V_{TAS}) = \frac{g}{V} \min \left\{ n_{\max}, \left[\frac{c_{L \max} \frac{1}{2} \rho(h) V_{TAS}^2 S}{W} - 1 \right] \right\}$
	$\dot{\phi}_{\max}(V_{TAS}) = V_{TAS} \cdot \left(\frac{c_{l_{\delta_A}} \delta_A}{\frac{b}{2} c_{l_p}} \right)_{\max}$

Table 6 -- Upper Performance Limitations Equations

4.3.4 Pseudo-Aircraft Surface Movement

Pseudo-aircraft surface movements are made possible by slightly different performance limitations when on the ground. Minimum flight path angle and speed are zero. The effect of wind on heading and flight path angle is ignored. The effect of wheel brakes and thrust reversers is included in the \dot{V}_{\min} calculation. The performance limitation on $\dot{\gamma}$ prevents the aircraft from leaving the ground until it has sufficient airspeed to do so. The bank angle state is still used, although its physical interpretation differs. It is considered to be proportional to the steering angle of the nose wheel.

4.4 Subject Perception of the Pseudo-Aircraft

The methods that the human subject has of perceiving the state of the pseudo-aircraft are critical to the success of this implementation. The subject has two primary ways of perceiving pseudo aircraft in the PLI/TCAS experiment: voice radio communications containing PLI and the TCAS system, including a traffic display and aural alerts. The specifications and setup for both are discussed here.

4.4.1 Voice Communication

Pseudo-aircraft PLI is accomplished by organizing individual pseudo-aircraft voice radio transmissions as events, discussed in Section 3.4. This is done by digitally recording them ahead of time, and then using the multi-agent software architecture to play them back at the proper time.

Once a radio transmission event is cued, it goes into a voice queue, Figure 25. The voice queue is important for several reasons. It is used in combination with the voice player, Figure 26, to prevent more than one call from occurring at a given time, to suspend voice calls when a human is transmitting, and to ensure that the subject hears only those transmissions on the frequency selected. Included in the definition of each voice call is a frequency that it is transmitted on, a priority value, a maximum wait time, and the file name of the digital sound file to play back.

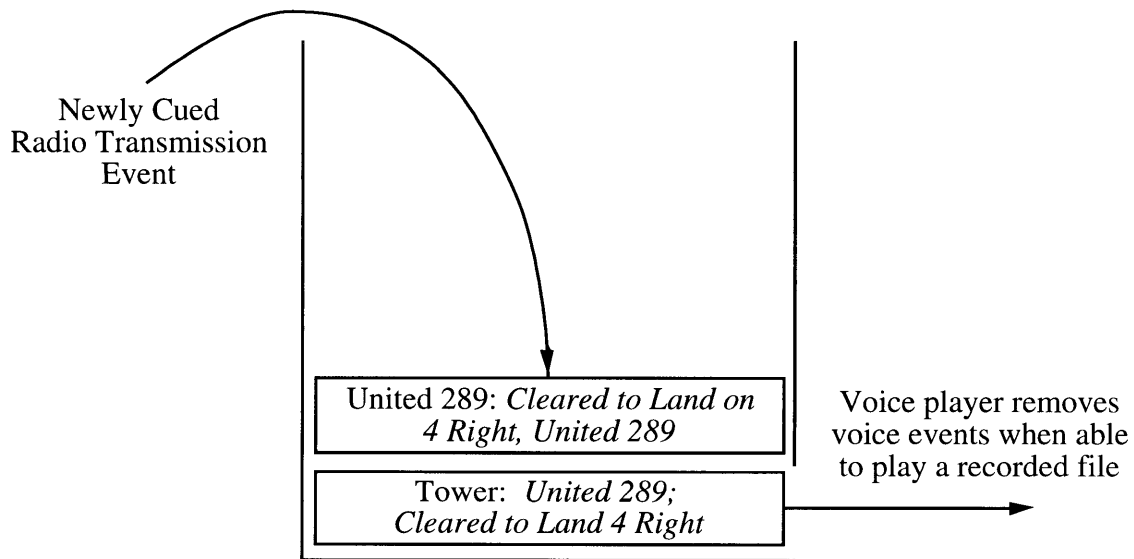


Figure 25 -- Voice Queue

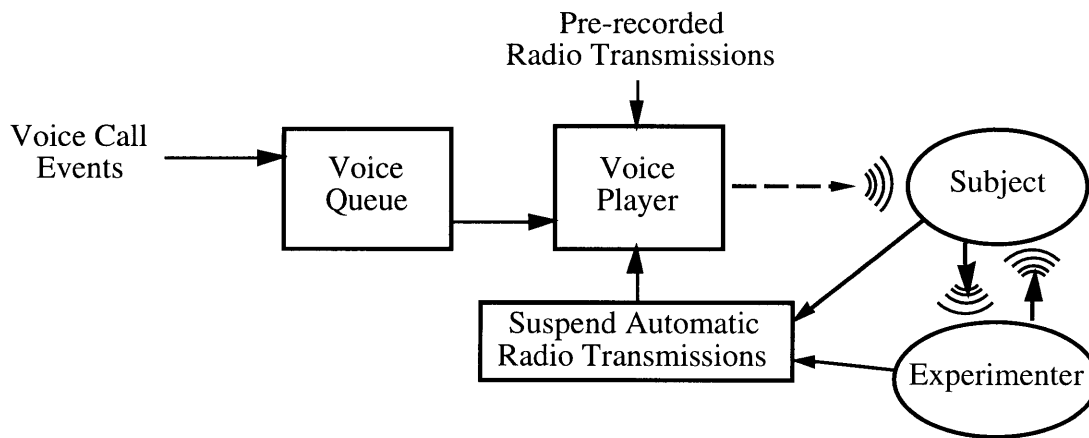


Figure 26 -- Voice Generation

The priority value prevents important transmissions from having to wait too long. Transmissions with higher priority simply skip those of lower priority when entering the queue. A maximum wait time is used to delete a radio communication event when it is no longer relevant, to prevent clearances for unimportant aircraft from occurring after the time period during which they make sense.

An experimenter acts as the controller that the subject is currently communicating with. Although most of the controller’s voice calls can be scripted and pre-recorded, all of the possible requests of the subject cannot be realistically prepared for or interpreted with available resources.

However, transmissions by the controllers to other aircraft are all pre-recorded and scripted as events. These voice calls are normally tied to pseudo-aircraft transmissions such that one is played immediately after the other. For example, the sound file:

KBOS Tower: *United 111, you are cleared to land, runway 4 left.*

would be immediately followed by:

United 111: *United 111, cleared to land 4 left.*

without interruption. This is accomplished simply by having both calls cued by the same criteria (example, same time) and with the same priority. With this method, the only way they can be split up is if a higher priority message enters the cue while the first one is playing.

The subject has a communications radio control console where the transmitting/receiving communications frequency and volume can be changed. The voice queue is suspended when the subject or experimenter-controller transmits, and restarted manually by the experimenter. At this point, radio transmission events would resume playback.

4.4.2 TCAS Output

The ACS was configured with two visual displays that contain TCAS information. Both place TCAS related information on existing displays. The first is traffic position and alert status on the Horizontal Situation Indicator (HSI). The second is RA maneuver command bars on the Attitude Indicator (AI). In addition to these visual displays, audible alarms are heard.

The HSI traffic display consists of symbols representing other aircraft depicted on the moving map display, Figure 27. The pseudo-aircraft symbol shape and color indicate the threat status of the pseudo-aircraft, Figure 28. The position of the symbol on the display shows the relative position of that aircraft. Relative altitude in hundreds of feet and an arrow indicated the vertical speed trend are shown next to the symbol. No heading or speed information is measured or displayed by TCAS.

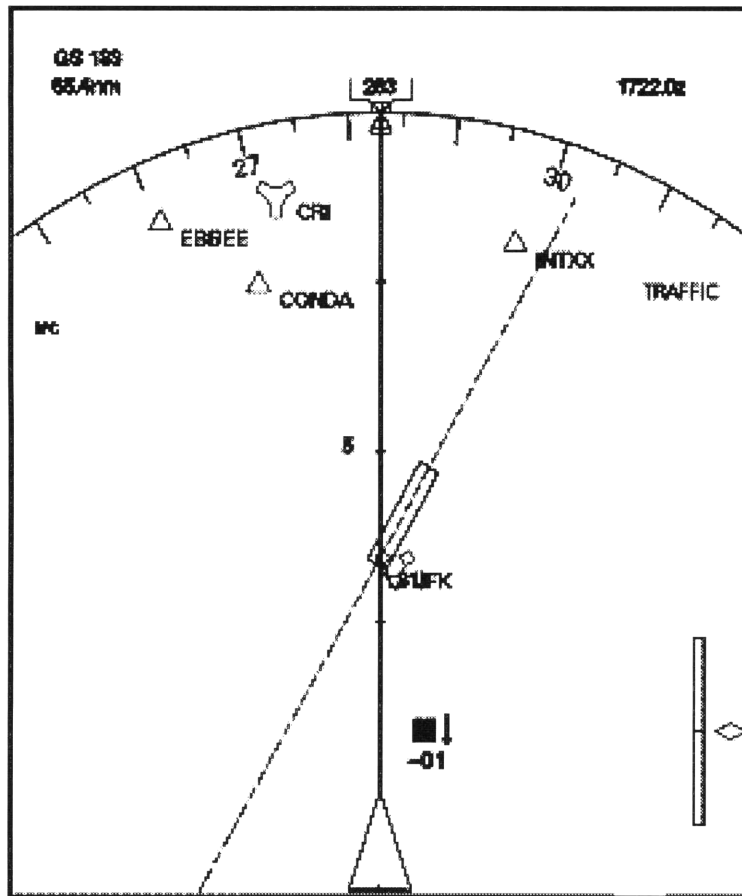


Figure 27 -- HSI Traffic Display

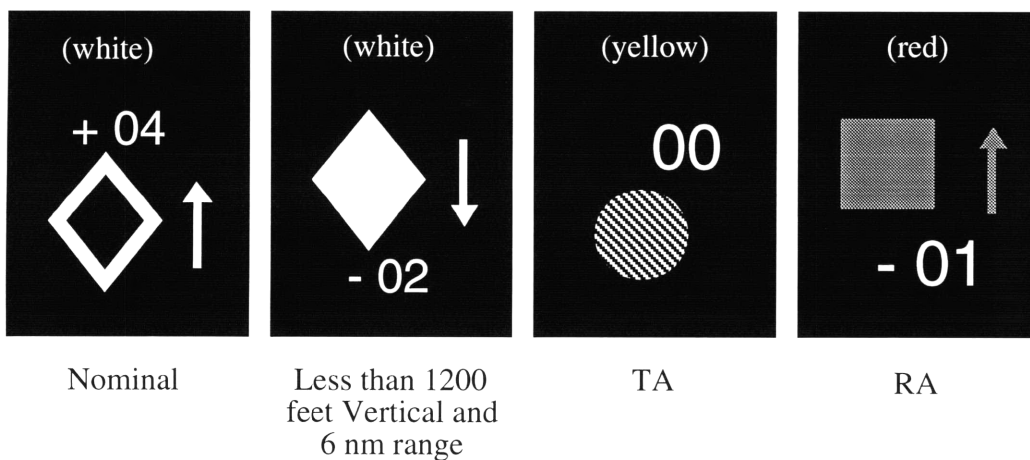


Figure 28 -- Traffic Symbols

The volume of space around the subject-aircraft containing the pseudo-aircraft that are displayed on the HSI depends upon the range selected by the subject. Nominally,

the HSI displays pseudo-aircraft that are with 40 nm range and within a relative altitude of 2700 feet above and below. If the subject-aircraft is climbing at greater than 300 feet per minute then the upper limit of the volume is raised to 9900 feet above the subject-aircraft. If descending at greater than 300 feet per minute, the lower limit is lowered to 9900 feet below the aircraft. It is possible for an aircraft to generate a TCAS alert and not show up on the HSI due to selected scale. In this case, the message “OFFSCALE TRAFFIC” appears on the HSI.

RA maneuvers are visually depicted by the RA symbol on the HSI as well as command bars, a red trapezoid, on the attitude indicator. The area created by these lines indicates the area to keep the aircraft attitude bars out of to comply with the RA vertical speed bound. These maneuvers are inhibited below 1000 feet AGL.

The TCAS alarms include audible alerts. There are 14 distinct audible alerts, summarized in Table 7. These 14 alerts were digitally recorded, and played based on the criteria indicated in the table. All TCAS aural alarms are inhibited below 1000 feet AGL.

Audio Alert	Meaning
<i>"traffic traffic"</i>	TA
<i>"climb climb climb"</i>	New RA; Aircraft level or descending asked to climb
<i>"climb climb now, climb climb now"</i>	Updated RA; Aircraft previously given a descend command switched to a climb command
<i>"climb crossing climb, climb crossing climb"</i>	RA; Aircraft asked to climb to avoid an aircraft that is presently at a higher altitude
<i>"increase climb, increase climb"</i>	RA; Climbing aircraft asked to increase its rate of climb
<i>"reduce climb, reduce climb"</i>	RA; Climbing aircraft asked to reduce climb to stay below a commanded rate
<i>"descend descend descend"</i>	New RA; Level or climbing aircraft asked to descend
<i>"descend descend now, descend descend now"</i>	Updated RA; Aircraft previously given a climb command switched to a descend command
<i>"descend crossing descend, descend crossing descend"</i>	RA; Aircraft asked to descend to avoid an aircraft that is presently at a lower altitude
<i>"increase descent, increase descent"</i>	RA; Descending aircraft asked to increase its rate of descent
<i>"reduce descent, reduce descent"</i>	RA; Descending aircraft asked to reduce climb to stay below a commanded rate
<i>"monitor vertical speed, monitor vertical speed"</i>	New RA; Aircraft is already within RA vertical speed command limits
<i>"monitor vertical speed"</i>	Updated RA; New RA that puts aircraft within command limits
<i>"clear of conflict"</i>	RA demoted to a TA or lower

Table 7 -- TCAS Aural Alarms

The PLI/TCAS experiment calls for pseudo-aircraft to have encounters with the subject-aircraft where TA and RA alerts result. When the subject-aircraft has an RA, the associated pseudo-aircraft has its own RA generated.

Each pseudo-aircraft has a prescribed parameter that determines whether it flies generated RA maneuvers or not. If the parameter is true, the aircraft flies the RA with a higher priority than its normal 4D waypoint guidance. If not, the RA is ignored and the pseudo-aircraft continues along its current desired trajectory. If a pseudo-aircraft is to fly its RA maneuver, it will wait a prescribed number of seconds before commencing the generated maneuver, thus simulating a normal pilot's reaction time.

4.5 Script Development

To script the flight of numerous aircraft over a significant length of time is not a trivial task. Add to this the creation of specific situations for a subject with varied actions, and it is clear that a critical aspect of this approach to multi-agent simulation is writing the script for the experiment.

4.5.1 Experimenter's Station Tools

For this implementation, an effective way to write and edit scripts was to include tools specifically for this purpose in the experimenter's station, Figure 18. This section contains a description of these tools.

A list of all aircraft is shown at the left of the screen, allowing the user to select specific aircraft. A circle appears around the selected aircraft on the display. Four other menus can be brought up: aircraft, amendment, event, and waypoint.

New pseudo-aircraft can be added and the initial conditions of any of the pseudo-aircraft changed by selecting the aircraft menu. The states that can be set are: name, latitude, longitude, altitude, speed, and heading. A flag is set if the aircraft is TCAS equipped. The pilot delay to TCAS alerts, transponder status (on/off), and aircraft type are also set here.

The amendment menu allows amendments to be inserted, deleted, and named. For the purposes of showing waypoints, an estimated amendment cue time can be entered. If a waypoint time is relative to the amendment cue time, then this estimated

amendment cue time is added to the waypoint time when it is displayed. This allows the experimenter to look at the results of different possible amendment cue times.

Events can be inserted and deleted in the event menu. The event type and cue can be changed. Parameters for the event cue and the event itself (such as voice communication frequency) can be modified.

The waypoint menu allows waypoints for a specific aircraft and situation to be inserted, deleted, and modified. Their positions, times, and type (subject relative, etc.) can be changed. An estimated speed to and from a selected waypoint is shown for reference. Waypoint sets can be stored, loaded, and time shifted. This allows standard trajectories, such as an approach to a specific runway, to be reused for multiple pseudo-aircraft. This considerably reduces the labor required to script a large number of pseudo-aircraft trajectories.

Waypoints are displayed in text as well as graphically on the display. A series of three waypoints for a particular aircraft is shown in Figure 29. Waypoint latitudes and longitudes are modified by clicking the mouse button with the mouse cursor at the desired location.

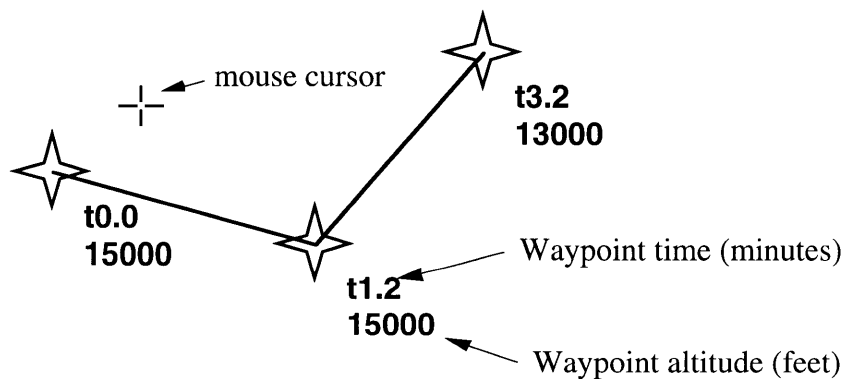


Figure 29 -- Script Editor Waypoint Display

Two other significant features of the experiment station tools are the fast time and pause modes. These functions accelerate the speed at which the simulation runs or stop it. These functions were found to be essential for effectively testing and improving scripts. By using a fast time mode, checking a situation at the end of a 30 minute flight may take only 3 minutes.

4.5.2 Pre-Recording Radio Communications

An organized way to record the numerous digital audio recordings to be played back is a necessary component of this type of system. Due to the large number of calls from any individual, it was effective to make an interactive program for the specific purpose of recording.

The program uses its own type of script that contains the dialog the experimenter wants recorded. The user reads the text to be recorded, hits a key, and then speaks into a microphone. The recorded file can be played back, and the user prompted as to whether to re-record or to keep the file. The process is depicted in Figure 30. An example of a short script that might be used to record the voice of Delta 018 is shown in Table 8.

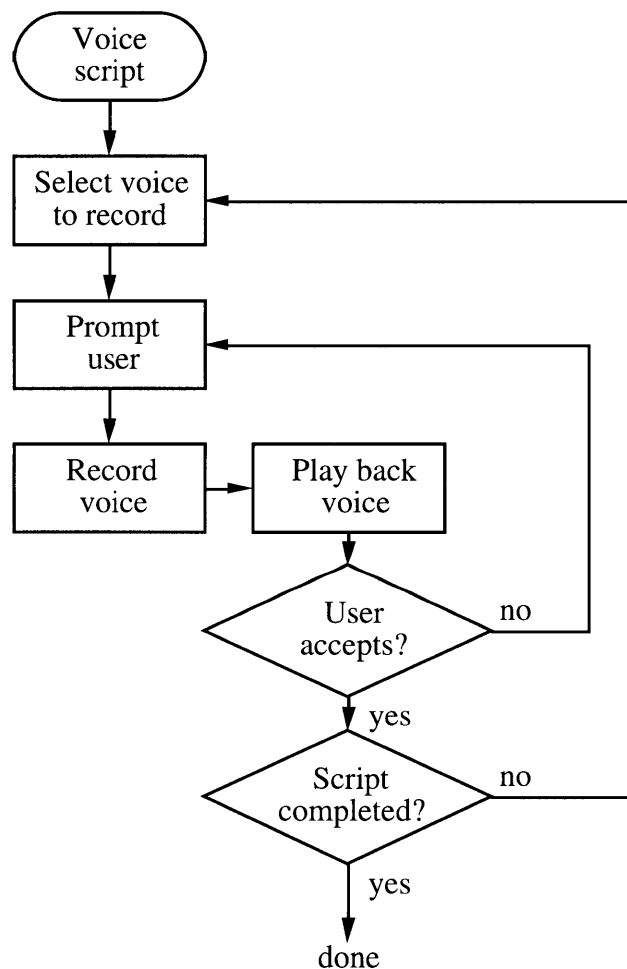


Figure 30 -- Digital Recording Process

	Prompt for User:	File Name:
Script for Delta 018:	Tower, Delta Zero One Eight Ready on Three One Right	del018.ready
	Position and hold Delta Zero One Eight	del018.position
	Delta Zero One Eight Rolling	del018.rolling
Script for Tower: (as relating to Delta 018)	Delta Zero One Eight, Hold Short Landing Traffic	del018.twrhold
	Delta Zero One Eight, Taxi to Position and Hold Three One Right	del018.twrposit
	Delta Zero One Eight, Cleared for Take-off	del018.twrelrtakeoff

Table 8 -- Sample Sound File Recorder Script

Due to the voice queue, a fair amount of latitude is available in scripting the radio transmission events because two pseudo-aircraft cannot transmit simultaneously. Voice transmissions wait in the queue until all other cued digitally recorded sound files have been played in their entirety.

5.0 DEMONSTRATION

This chapter describes a sample script, or flight, used to demonstrate the robust situation generation architecture and the implementation developed. In addition, the process of developing amendments for the pseudo-agents is discussed.

5.1 Sample Script

The expected flight path of the subject-aircraft is depicted in Figure 31. The subject-aircraft starts at 23,000 feet above LVZ (Wilkes Barre) VOR and proceeds to a landing at New York's JFK airport runway 31 Left. The expected flight path is defined by a series of 4D waypoints. Each point has a latitude, longitude, altitude, and time. The subject receives clearances as necessary to match the expected flight path as closely as possible. The robust situation generation architecture allows specific situations to happen even when the subject does from the expected path or speed. The general locations of the three situations are labeled on the figure as A, B, and C.

There are three situations included in this demonstration script. First, the subject is to see and hear (using the TCAS and/or PLI) aircraft ahead request lower altitudes due to turbulence at 19,000 feet. This requires several pseudo-aircraft to fly similar flight paths as the subject while maintaining a scripted separation from the subject and each other. It also requires radio communications from the pseudo-aircraft. It is scripted to occur in area A shown on Figure 31.

The second situation is a TA caused by an aircraft passing below. This is to appear normal to the subject. A pseudo-aircraft is to pass 2000 feet below the subject on a perpendicular course while the subject is in area B in Figure 31.

The final situation is a collision hazard while on final approach to runway 31L at JFK, shown as area C in Figure 31. The intruder pseudo-aircraft flies to a scripted location relative to the subject-aircraft on its parallel approach to runway 31R. It deviates from its flight path and creates a collision hazard at five nm from touchdown. The intruder and subject each get a TCAS RA.

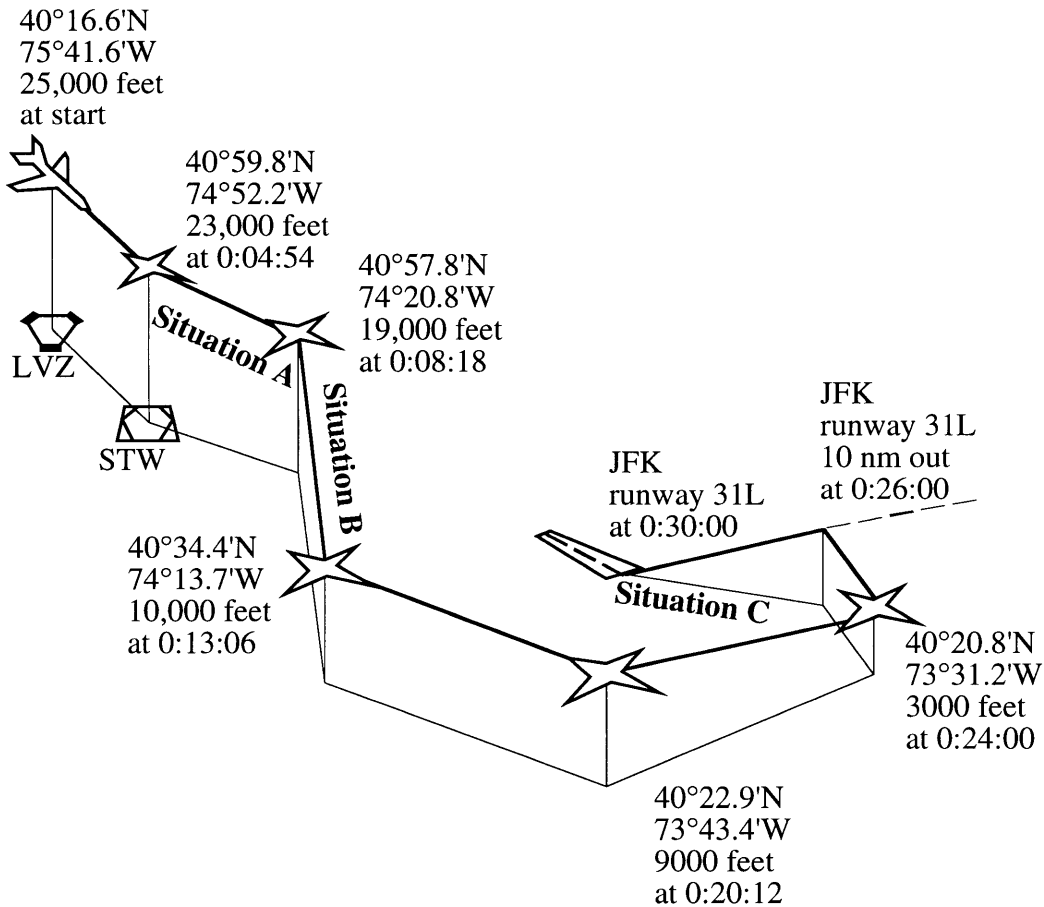


Figure 31 -- Nominal Subject Flight Path

Reasonable background traffic, in the form of pseudo-aircraft not directly involved in any of the above situations, is also used. All aircraft, including this background traffic, must perform within reasonable performance limits and interact properly for an individual flight to be a success.

5.2 Writing the Demonstration Script

Normally, the first step in writing a script is determining a nominal flight path for the subject-aircraft, shown in Figure 31 above. If branches of significantly different flight paths are to be allowed, these other branches should also be determined. In the case of the demonstration script, the only branch allowed occurs during situation A. If the subject requests a lower altitude, it is given. If not, the new altitude will be given at the end of situation A.

The 4D waypoints for the pseudo-aircraft are then created based on the subject-aircraft expected flight path. This approach ensures that the highest fidelity pseudo-aircraft generation occurs where it is needed, in view of the subject. In this case, standard arrival and departure waypoint sets are stored for JFK, Newark, and La Guardia runways. These sets were used repeatedly to eventually define the flight paths of 34 pseudo-aircraft.

The next step taken in this demonstration flight was to split the flight into 15 amendments. The first 14 amendments are designed to cue approximately every two minutes. They provide waypoint updates for all active pseudo-aircraft. The amendment cue for each will be subject-aircraft ETA of less than one minute to a point on the map. By using this approach; if the subject travels slower or faster than expected, amendments will be cued later or earlier respectively. This effectively adjusts the pseudo-aircraft waypoint times every two minutes to variation in subject-aircraft speed.

The ETA amendment cue was implemented in the following manner. The horizontal distance between a point on the ground and the subject-aircraft is calculated. This distance is divided by the subject-aircraft's ground speed to get an ETA. If this value is less than the scripted value, the amendment is cued. This amendment cue is depicted in Figure 11. This implementation was chosen because it is independent of subject-aircraft altitude and cues even when the subject does not fly directly over the point on the ground.

The demonstration flight amendment list is shown in Table 9. The flight contains three situations that are critical to the experiment. The turbulence reports situation corresponds to the third amendment in the list. The TA corresponds to the fifth. The parallel approach RA begins when the 14th amendment is cued, and an additional amendment is cued so that the RA occurs at 5 nm from the runway threshold. The RA amendment updates only the pseudo-aircraft that will cause the RA. The other amendments are necessary to maneuver the pseudo-aircraft realistically and place them for the three experiment critical situations in a robust manner.

Amendment	Expected cue time	note:
1	0:00:00	Active at start-up (initial set of waypoints and events)
2	0:02:00	
3	0:04:00	Turbulence reported at 19,000 feet from pseudo-aircraft situated ahead of the subject-aircraft
4	0:06:00	
5	0:08:00	TA situation; TA aircraft gets a subject relative waypoint 2000 feet directly below the subject-aircraft
6	0:10:00	
7	0:12:00	
8	0:14:00	
9	0:16:00	
10	0:18:00	
11	0:20:00	
12	0:22:00	
13	0:24:00	
14	0:26:00	Intruder aircraft gets subject relative waypoints on the parallel approach in order to get in the proper position
RA	0:26:45	Collision hazard (RA); only the intruder receives new waypoints; intruder flies directly into the subject-aircraft until RA occurs

Table 9 -- Demonstration Script Amendments

A time of two minutes between amendments was chosen as a trade-off. This time is long enough for the pseudo-agents to change their arrival times. Also, two minutes is short enough that the subject cannot dramatically change speed and then fly for a long time at this new speed between amendment cues. This ensures that the arrival time at the following amendment cue cylinder is never dramatically different than expected.

The possibility of the subject receiving an altitude clearance lower than 19,000 feet did not affect the script writing, because the subject-aircraft is expected to return to the same flight path regardless of the altitude used during the situation. Had this not been the case, then more than one set of amendments might have been required downstream of this point.

In summary, an expected subject flight path was chosen for the subject aircraft. Pseudo-aircraft flight paths and events were defined to generate the desired situations. The flight was split into 15 amendments, three of which are considered critical to an experiment. Subject-relative waypoints were utilized in two of the experiment critical situations.

6. ACHIEVED ROBUSTNESS

In this chapter, achieved robustness was evaluated by varying the subject's flight path to extremes of speed and lateral position error, as well as subject blunder errors. They will be varied to the point that the script presented in Chapter 5 and the robust situation generation architecture can no longer adequately control the pseudo-agents, generate desired situations, or when the extreme of the subject-aircraft's performance envelope has been reached. Speed and lateral position errors are differences between the expected subject-aircraft flight path and the actual flight path. Blunder errors are wrong turns and incorrect altitude commands made by the subject.

6.1 Speed Error

One way the subject-aircraft's trajectory can vary from the expected trajectory is if the actual speeds used during the flight are different than the expected speeds. In addition, pilots will slow to landing speed at different times, perform slightly different descent profiles, etc.

The achieved robustness to speed variation was explored by varying the speed of the subject by a multiplicative factor. The system was tested by flying the subject-aircraft at 120, 110, 90, and 80% of the expected speed profile, which contained speed near both the upper and lower bounds of the subject-aircraft's speed envelope.

The resulting amendment cue times are shown in Figure 32. Early in the 110 and 120% cases the subject-aircraft was performance limited. This is shown by the curves lying roughly on top of each other in the first 5 amendments. Though not apparent from the figure, the subject-aircraft flew close to stall speed for much of the later part of the slowest test, 80% speed.

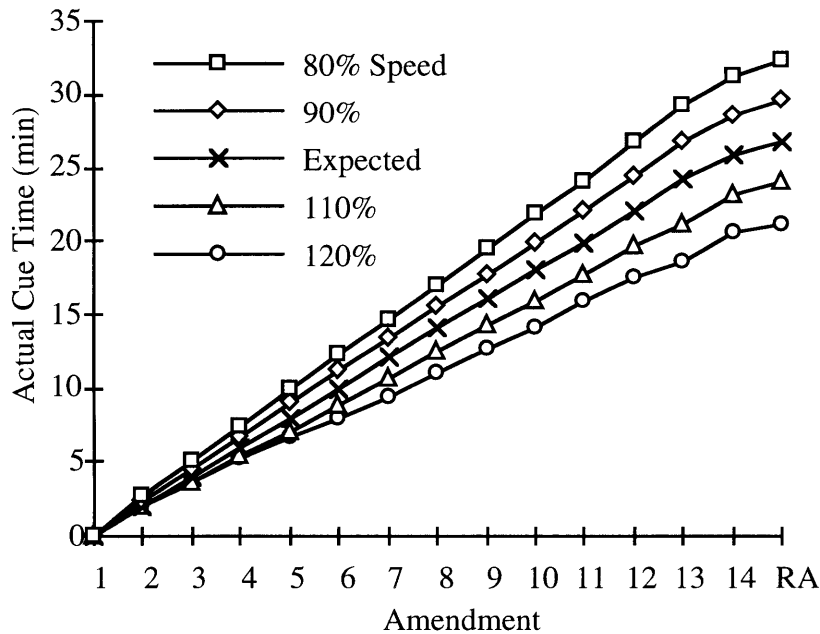


Figure 32 -- Actual Cue Times for Different Subject Speeds

For these four tests, the three experiment critical situations and the background traffic were observed. For all but the fastest test, 120% speed, experiment critical situations occurred as scripted and background traffic appeared to maneuver properly.

For 120% case, it was observed that some background traffic could not keep up with the experiment due to the 250 kts speed limit below 10,000 feet. As a result, some turns were cut short and trailing distance in some landing sequences became unreasonably small. For the same reason, the pseudo-aircraft involved in the parallel approach RA could not arrive in time to cause a collision hazard at 5 nm from the runway as scripted. A summary of these results is shown in Table 10.

Speed	Turbulence			
	Reports	TA	RA	Background
80%	yes	yes	yes	yes
90	yes	yes	yes	yes
110	yes	yes	yes	yes
120	yes	yes	no	no

Table 10 -- Results of Subject Speed Variation

Although the subject is highly unlikely to fly the 120% trajectory, it is possible that the waypoints given to offending pseudo-aircraft could be modified to allow the system to work under these conditions. The approach would be to have these aircraft fly slower when the subject is on the expected flight path. Clearly, there is a balance between tolerance allowed at the bottom and top of the subject's speed range. The experimenter can shift the range of speeds up and down by adjusting pseudo-aircraft waypoints. As shown, this speed range available to the experimenter in this implementation is approximately the same as the subject-aircraft's performance limits.

6.2 Position Error

Another way the subject can vary flight path from expected, beyond speed, is to fly slightly off the expected course. This can manifest itself as being slightly left, right, above, or below the expected flight path. This is clearly of significance for those situations requiring a collision hazard. Amendment and event cues must also be tolerant of such position errors.

Achieved robustness to position error was tested by flying the subject-aircraft one, two, and four nm right of the expected flight path. These errors are extreme for a transport category aircraft following an ATC clearance, but were chosen in order to demonstrate the limitations of the system. No position error was included once the subject was established on the localizer. Position errors that result from missed turns, confused clearances, etc. are discussed in the next section, and are referred to as blunder errors.

Position errors of one and two nm right of the expected course had no effect on the system. This was not the case with a position error of four nm. In this case, using a the ETA of one minute amendment cue discussed in the previous section meant that amendments will not cue if the subject is expected to fly slower than four nm per minute. This corresponds to a ground speed of 240 kts. Once the subject-aircraft speed dropped below this value, which happened shortly after the TA situation, new amendments were not properly cued. This caused the simulation to longer be tolerant to subject actions, so the parallel approach situation and the background traffic were no longer assured. A summary of these results is shown in Table 11.

Position Error	Turbulence Reports	TA	RA	Background
1 nm	yes	yes	yes	yes
2	yes	yes	yes	yes
4	yes	yes	no	no

Table 11 -- Results of Subject Position Error

If an experiment requires tolerance to position errors of this magnitude, four nm, a different amendment cue should be used. Perhaps a larger cylinder could be used to cue the amendments, corresponding to a larger ETA value. Other options include defining a vertical plane that, when crossed by the subject, cues the amendment.

6.3 Blunder Error

The final type of subject variation explored in this chapter is the blunder error. Tolerance to blunder errors were tested by having the subject-aircraft make a key turn at two levels of delay. In all cases the experimenter was assumed to intervene, acting as ATC, and clear the subject to a new heading that will put the subject back on the desired flight path. Achieved robustness was also tested by having the subject-aircraft descend too far when capturing a cleared altitude, and then remain at this lower altitude until it returned to the expected flight path.

6.3.1 Late Turn

The first test was to have the subject aircraft fly 2.5 nm beyond the point at which the base turn was to be initiated, the 9000 feet 0:20:20 waypoint in Figure 31. This could be caused because the subject did not hear the new clearance or was otherwise confused by it. This blunder had no effect on the system. When the blunder distance was increased to five nm amendment #12 in Table 9 did not cue. This *might* cause a problem if the subject were to fly at a very different speed than expected during this period of time, which would be unusual for this particular phase of flight, because the pseudo-aircraft had to wait four minutes between trajectory updates rather than the normal two minutes. A summary of these results is shown in Table 12.

Blunder Error	Turbulence			Background
	Reports	TA	RA	
• 2.5 nm late turn to base	yes	yes	yes	yes
• 5 nm late turn to base	yes	yes	yes	see text
• descent to 17,000 rather than 19,000 feet	yes	yes	yes	yes
• descent to 6000 rather than 10,000 feet	yes	yes	yes	yes

Table 12 -- Results of Subject Blunder Error

6.3.2 Descending Below Cleared Altitude

Altitude blunder errors were also tested. The descent to 19,000 feet shown in Figure 31 was lowered to 17,000 feet. In a separate test, the descent to 10,000 feet was increased to 6000 feet. These blunder errors represent extremes of possible mistakes to be made by subjects. Also, these two cases are among the few possible scenarios where the subject-aircraft can descend significantly too far using a reasonable descent rate. In both cases, the system performed as expected, also shown in Table 12. In the descent to 6000 feet it was noted the subject's flight path came closer to other, background, aircraft, as would be the case in the real ATC system.

7. SUMMARY AND CONCLUSIONS

A robust situation generation approach has been developed. This approach utilizes subject feedback in two fundamental ways. First, the trajectory of a pseudo-agent can be adjusted continuously in response to the motion of the subject. Second, discrete qualitative amendments to the agents' trajectories can be cued by some aspect of the subject's current state, such as being within a certain distance of a given point. In addition, discrete actions can be similarly cued for the pseudo-agents, such as turning on lights or lowering landing gear, to allow for pseudo-agents that can do more than just maneuver in space.

The robust situation generation approach was implemented for an air transportation system experiment. This experiment required voice communication to be heard by a human subject from other aircraft. It also required specific types of collision hazards between the subject and other aircraft. A pseudo-aircraft model and other related software was developed to implement the situation generation architecture for use in this type of research.

A demonstration situation generation script, designed to be a part of an air transportation research experiment, was developed. This script contained three situations that were critical to the experiment. The script developed included fifteen amendments, where these amendments organized background aircraft and led aircraft involved in the three experiment critical situations to their proper positions.

The achieved robustness of this demonstration flight to variations in subject actions was explored. This analysis included varying the subject-aircraft speed and position accuracy, as well as testing blunders by the subject, such as missing a turn.

Achieved robustness indicated that the system allows specific situations to be generated for subjects who perform within a reasonably large envelope of possible action. In cases where the system failed, the subject was performing at an extreme of possible action or a limitation was found in the script that could be rectified if needed.

The achieved robustness depended, in part, on the scripting process. The architecture allows the experimenter to 'design in' possible contingencies of subject actions. As a result, the experimenter can develop scripts that are arbitrarily flexible. The limitation is the ability of the pseudo-agents to maneuver.

The development of a script is an important element of the system presented. Scripting the flight of multiple aircraft in a crowded sky is not a trivial task even before attempting to generate specific situations. This effort can be reduced using software specifically developed for the purpose of script writing. It was found that allowing the experimenter to re-use stored trajectories for more than one pseudo-aircraft, and a simulator that could run faster than real time to test scripts were affective tools.

Triggering voice communications using the robust situation generation architecture, with the implementation of a voice queue, was found to be a powerful technique. It allows pseudo-aircraft radio transmissions to be pre-recorded. This eliminates the need for a large number of 'pseudo-pilots' that would normally provide these transmissions from manned remote stations.

The software architecture developed has proven to be effective for air transportation research. It could also be applied other applications, where only the details of the implementation would differ. Any system that must coordinate one or more pseudo-agents to give specific situations to a maneuvering subject-vehicle is a candidate for a robust situation generation scheme.

References

1. Bayne, S., Schwartz K., Smithwick M., and Weske R. A., *Pseudo Aircraft Systems (PAS) Documentation Package*, NASA Ames Research Center, Release 1.1, November 1990
2. Burden, R. L. & Faires, J. D., *Numerical Analysis*, Fourth Edition, PWS-KENT, Boston 1989
3. Midkiff, A. H. & Hansman, R. J. *Identification of Important "Party Line" Information Elements and the Implications of Situational Awareness in the Datalink Environment*, MIT Aeronautical Systems Laboratory Report ASL-92-2, May 1992
4. *Minimum Operational Performance Standards for Traffic Alert and Collision Avoidance System (TCAS) Airborne Equipment*, Volumes 1 and 2, RTCA/DO-185, September 1983
5. Nolan, M. S. *Fundamentals of Air Traffic Control*, Wadsworth, Belmont California 1994
6. Pritchett, A. R. & Hansman, R. J. *Variations in Party Line Information Requirements for Flight Crew Situation Awareness in the Datalink Environment*, MIT Aeronautical Systems Laboratory Report ASL-94-5, May 1994
7. Simpson, R. W., *Engineering of Air Traffic Control Systems*, Flight Transportation Laboratory, August 1993 Draft copy
8. Stevens, B. L. & Lewis, F. L., *Aircraft Control and Simulation*, Wiley, New York 1992
9. *TCAS II for Glass Cockpit Airplanes*, Video Tape, United Airlines Flight Center, February 1991
10. Ward, D. T., *Introduction to Flight Test Engineering*, Elsevier, Amsterdam 1993

APPENDIX A:

Performance Limit Development

An important part of the pseudo-aircraft model presented is the performance limitations on the states and controls. Derivations for the performance limitations used in this work are described in this appendix.

A.1 State Limits

The first state limit is an upper bound of 250 knots indicated airspeed below 10,000 feet [FAR 91.117]. Speed limits are found for true airspeed first, then converted to ground speed. Assuming a standard atmosphere and a limit of 250 knots on calibrated airspeed, this becomes

$$V_{TAS\max}(h|h \leq 10,000 \text{ feet}) = \sqrt{\sigma(h)} \cdot 250 \text{ knots.}$$

where $\sigma(h)$ is the density ratio calculated from the atmosphere model as a function of altitude. A further regulatory speed conditions, 200 knots indicated, exists within 4 nm and below 2500 feet above the surface of certain airports. However, in practice this would not become an active constraint for a well designed experiment, so this limitation was not implemented. For flight above 10,000 feet, the pseudo-aircraft is limited by its maximum cruise Mach number,

$$V_{TAS\max}(h|h > 10,000 \text{ feet}) = M_{cruise}a(h),$$

where $a(h)$ is the local speed of sound calculated from the atmosphere model as a function of altitude.

For the purposes of this work, the lower limit on true airspeed is the bottom of the power curve or stall speed, whichever is faster. Stall speed is a function of the maximum lift coefficient, $c_{L\max}$,

$$V_{TAS\min c_{L\max}}(h) = \sqrt{\frac{2W}{\rho(h)S c_{L\max}}}.$$

The power curve refers to the regime of flight where decreasing airspeed decreases drag. This needs to be avoided or special attention would be required for the guidance model. This is avoided when the limit

$$V_{TAS\min}(h) = \sqrt{\frac{2W}{\rho(h)S} \max\left(\frac{1}{c_{L\max}}, \sqrt{\frac{k}{c_{D0}}}\right)}$$

is used. To convert these limits to ground speed

$$V_{\max} = V_{TAS\max} - V_{TAS} + V$$

$$V_{\min} = V_{TAS\min} - V_{TAS} + V$$

are used.

The next state which requires performance limitations is flight path angle, or γ . Because flight path angle is given priority over speed, the flight path limits are the best that can be achieved at a constant speed that is within the upper and lower speed limits for the current altitude.

At a given altitude, the maximum available flight path angle is

$$\gamma_{\max}(h) = \frac{T(h) - D(h, V_{TAS})}{W} \Big|_{\max} . \quad (6)$$

This can be found using the drag equation

$$D(h, V_{TAS}) = c_D \frac{1}{2} \rho(h) V_{TAS}^2 S \quad (7)$$

where c_D is the pseudo-aircraft's drag coefficient, found by

$$c_D = c_{D0} + c_{D\delta} \delta + k c_L^2 \quad (8)$$

where δ takes on a value of zero when the aircraft is in its minimum drag configuration, and one when it is in its maximum drag configuration (e.g. spoilers deployed). This is a valid approximation for drag so long as the aircraft remains below the compressibility drag rise Mach number, ensured by the upper speed limit. The steady state lift coefficient, c_L , can be written as a function of TAS, altitude, and weight:

$$c_L = \frac{W}{\frac{1}{2}\rho(h)V_{TAS}^2}. \quad (9)$$

Combining Equations 6 through 9,

$$\gamma_{\max}(h, V_{TAS}) = \frac{T_{\max}(h) - c_{D0} \frac{1}{2}\rho(h)V_{TAS}^2 S}{W} - \frac{kW}{\frac{1}{2}\rho(h)V_{TAS}^2 S}.$$

Finding the airspeed that maximizes, one finds the final form of the maximum flight path angle as a function of altitude,

$$\gamma_{\max}(h) = \frac{T_{\max}(h)}{W} - 2\sqrt{kc_{D0}},$$

with an airspeed that corresponds to the bottom of the power curve. Unfortunately, stall speed may be violated, so the equation

$$\gamma_{\max}(h) = \frac{T_{\max}(h) - c_{D0} \frac{1}{2}\rho(h)V_{TAS\min}^2 S}{W} - \frac{kW}{\frac{1}{2}\rho(h)V_{TAS\min}^2 S}$$

should be substituted to avoid this problem. The maximum available thrust as a function of altitude is approximated by

$$T_{\max}(h) = T_{\max}(0)[\sigma(h)]^{0.6}.$$

Minimum thrust is assumed to be constant.

The lower limit on flight path angle is found by

$$\gamma_{\min}(h) = \left. \frac{T(h) - D(h, V_{TAS})}{W} \right|_{\min}.$$

This normally occurs at maximum speed, maximum drag configuration, and minimum thrust, described by

$$\gamma_{\min}(h) = \frac{T_{\min}(h) - (c_{D0} + c_{D\delta}) \frac{1}{2}\rho(h)V_{TAS\max}^2 S}{W} - \frac{kW}{\frac{1}{2}\rho(h)V_{TAS\max}^2 S}.$$

As a practical matter the flight path angle limits needs to be reduced slightly. This permits a small variation of speed for hitting 4D waypoints during maximum climbs or descents. To this end, flight path limits were set to 90% their calculated values.

The final state with a limit is bank angle. The purpose is to stay within the normal operating range of these types of aircraft. The guidance model determines what bank angle is flown, less than the maximum, by the pseudo-aircraft. A small bank angle limit, such as 30°, also means bank terms in the longitudinal performance limits can be neglected.

A.2 Control Limits

The upper limit, maximum available acceleration, is a function of current altitude, airspeed, and flight path angle,

$$\dot{V}_{\max}(h, V_{TAS}, \gamma) = \left[\frac{T_{\max}(h) - D_{\min}(h, V_{TAS})}{W} - \gamma \right] g.$$

Using Equations 7, 8, and 9 to find the minimum drag, the upper limit is then

$$\dot{V}_{\max}(h, V_{TAS}, \gamma) = \left[\frac{T_{\max}(h) - c_{D0} \frac{1}{2} \rho(h) V_{TAS}^2 S}{W} - \frac{kW}{\frac{1}{2} \rho(h) V_{TAS}^2 S} - \gamma \right] g.$$

The lower limit is determined similarly,

$$\dot{V}_{\min}(h, V_{TAS}, \gamma) = \left[\frac{T_{\min}(h) - D_{\max}(h, V_{TAS})}{W} - \gamma \right] g, \text{ or}$$

$$\dot{V}_{\min}(h, V_{TAS}, \gamma) = \left[\frac{T_{\min}(h) - (c_{D0} + c_{D\delta}) \frac{1}{2} \rho(h) V_{TAS}^2 S}{W} - \frac{kW}{\frac{1}{2} \rho(h) V_{TAS}^2 S} - \gamma \right] g.$$

Limits on $\dot{\gamma}$ are either limits on c_L or structural limits of the airframe, whichever is more conservative. In general, c_L limits occur at low speeds and structural limits occur at high speeds. The lift coefficient upper and lower limits are

$$\dot{\gamma}_{\max c_L}(h, V_{TAS}) = \frac{g}{V} \left[\frac{c_{L\max} \frac{1}{2} \rho(h) V_{TAS}^2 S}{W} - 1 \right] \text{ and}$$

$$\dot{\gamma}_{\min c_L}(h, V_{TAS}) = \frac{g}{V} \left[\frac{c_{L\min} \frac{1}{2} \rho(h) V_{TAS}^2 S}{W} - 1 \right]$$

respectively. The structural limits on $\dot{\gamma}$ are limits of acceleration, normally given as a multiple of the acceleration of gravity, n_{\max} and n_{\min} in this case. The performance limits on $\dot{\gamma}$ are finally

$$\dot{\gamma}_{\max}(h, V_{TAS}) = \frac{g}{V} \min \left\{ n_{\max}, \left[\frac{c_{L\max} \frac{1}{2} \rho(h) V_{TAS}^2 S}{W} - 1 \right] \right\} \text{ and}$$

$$\dot{\gamma}_{\min}(h, V_{TAS}) = \frac{g}{V} \max \left\{ n_{\min}, \left[\frac{c_{L\min} \frac{1}{2} \rho(h) V_{TAS}^2 S}{W} - 1 \right] \right\}.$$

The roll rate limit is found given the traditional lateral stability and control non-dimensional derivatives for rolling moment, c_{l_p} and $c_{l_{\delta_A}}$, as parameters,

$$I_x \ddot{\phi} = \left[c_{l_p} \left(\frac{b \dot{\phi}}{2V} \right) + c_{l_{\delta_A}} \delta_A \right] \frac{1}{2} \rho(h) V_{TAS}^2 S b$$

where I_x is the moment of inertia about the roll axis b is the wing span, and δ_A is the effective aileron deflection. The maximum steady state roll rate occurs when the left hand side is zero, or

$$c_{l_p} \left(\frac{b \dot{\phi}}{2V_{TAS}} \right) = -c_{l_{\delta_A}} \delta_A.$$

This yields a maximum roll rate magnitude, the final performance limitation, as a function of airspeed in the form

$$\left| \dot{\phi} \right|_{\max}(V_{TAS}) = V_{TAS} \cdot \left(\frac{c_{l_{\delta_A}} \delta_A}{\frac{b}{2} c_{l_p}} \right)_{\max}.$$