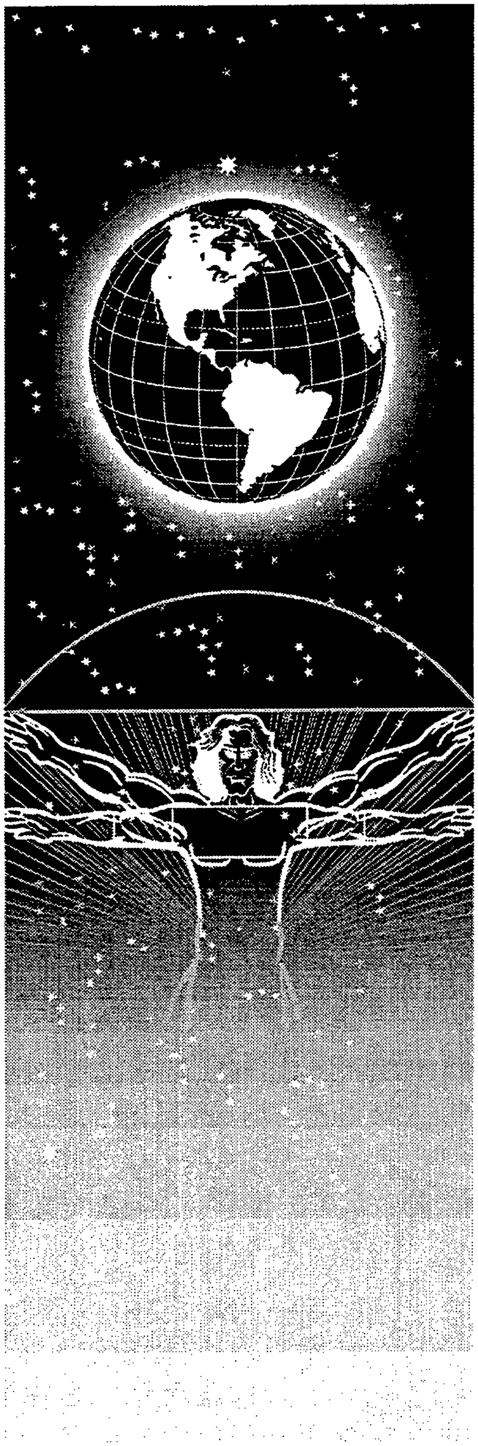
AFRL-HE-WP-TR-1998-0028



UNITED STATES AIR FORCE RESEARCH LABORATORY

A DEMONSTRATION OF DELAY AND CONSTRUCTIVE MODELING EFFECTS IN DISTRIBUTED INTERACTIVE SIMULATION

Jeffrey A. Doyal

SCIENCE APPLICATIONS INTERNATIONAL CORPORATION (SAIC)
4031 COLONEL GLENN HIGHWAY
BEAVERCREEK OH 45431

Robert E. Penry

SCIENCE APPLICATIONS INTERNATIONAL CORPORATION (SAIC)
7400 VISCOUNT BLVD., SUITE 220
EL PASO TX 79925

Edward A. Martin
David G. Hoagland

HUMAN EFFECTIVENESS DIRECTORATE
CREW SYSTEM INTERFACE DIVISION
WRIGHT-PATTERSON AFB OH 45433-7022

FEBRUARY 1998

FINAL REPORT FOR THE PERIOD FEBRUARY 1997 TO DECEMBER 1997

Human Effectiveness Directorate
Crew System Interface Division
Wright-Patterson AFB OH 45433-7022

NOTICES

TECHNICAL REVIEW AND APPROVAL

AFRL-HE-WP-TR-1998-0028

This report has been reviewed by the Office of Public Affairs (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

The voluntary informed consent of the subjects used in this research was obtained as required by Air Force Instruction 40-402.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

HENDRICK W. RUCK, PhD
Chief, Crew System Interface Division
Air Force Research Laboratory

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
February 1998

3. REPORT TYPE AND DATES COVERED
Final, February 1997 - December 1997

4. TITLE AND SUBTITLE

A Demonstration of Delay and Constructive Modeling Effects in Distributed Interactive Simulation

5. FUNDING NUMBERS

C: F33615-92-D-2293
PR: 2829
TA: 01
WU: 10

6. AUTHOR(S)

*Jeffrey A. Doyal
**Robert E. Penry
Edward A. Martin
David G. Hoagland

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

*Science Applications International Corporation (SAIC)
4031 Colonel Glenn Highway
Beavercreek OH 45431

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Air Force Research Laboratory, Human Effectiveness Directorate
Crew System Interface Division
Air Force Materiel Command
Wright-Patterson AFB OH 45433-7022

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

AFRL-HE-WP-TR-1998-0028

11. SUPPLEMENTARY NOTES

**Science Applications International Corporation (SAIC)
7400 Viscount Blvd., Suite 220
El Paso TX 79225

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

A laboratory investigation examined the effect of Protocol Data Unit (PDU) transport delays and the use of constructive modeling of a human operator in the loop to determine how these variables might impact the outcome of tightly coupled Distributed Interactive Simulation (DIS) engagements between high-velocity entities. A series of simulated engagements between an air defense simulation (medium range surface-to-air missile) and an F-16 were conducted. Engagement outcome was evaluated for both a 0 ms transport delay (baseline) and a 100 ms delay condition (the maximum allowable within the IEEE standard). In addition, trials were performed under two different threat control conditions. In half of the trials, a live operator controlled the missile launch sequence, with a constructive model controlling the remaining trials. Analysis of the outcomes showed a dramatic effect of transport delay, with missile effectiveness dropping from 0.73 P_k (Probability of Kill) in the 0 ms delay condition to 0.00 P_k in the 100 ms delay condition. Differences between live operator and constructive threat model behavior were observed but did not significantly impact engagement outcome. Results suggest that the delays and architecture used in today's DID exercises create artifacts (e.g., position errors) that can affect test results and, potentially, lead analysts to draw erroneous conclusions. Approaches for improving DIS performance are discussed.

14. SUBJECT TERMS

DIS, Distributed Interactive Simulation, Transport Delay, Latency, Constructive Model, Human Performance, Clock Error, Protocol Data Unit, PDU, Network Time Protocol, NTP

15. NUMBER OF PAGES
35

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UNLIMITED

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

This page intentionally left blank.

PREFACE

This effort was accomplished in cooperation with the Armstrong Laboratory Design Technology Branch, Veda Incorporated, and Science Applications International Corporation (SAIC). SAIC was working in support of the Design Technology Branch, Human Engineering Division, Crew Systems Directorate of the Armstrong Laboratory (since reorganized as the Human Effectiveness Directorate, Crew System Interface Division of Air Force Research Laboratory (AFRL)) under contract number F33615-92-D-2293.

The authors express special thanks to Mr. Dave O'Quinn of Veda Incorporated who provided quality simulation engineering support, and to Mr. David Malek of SAIC who helped us to navigate through vast amounts of data. Thanks are also due to Lt George Woodworth of AFRL, who coordinated this study and the running of test subjects, and to the eight pilot subjects whose operational experience and professionalism contributed greatly to the success of the study.

This page intentionally left blank

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

LIST OF FIGURES

Figure **Page**

LIST OF TABLES

Table **Page**

SECTION 1.
INTRODUCTION

Background

Distributed Interactive Simulation (DIS) is a synthetic environment within which models and live entities, including humans and systems, can interact through computer simulations and networking. Initially developed as a concept to provide cost-effective, high-fidelity virtual battle environments in which to train warfighters, the DIS architecture supports in-depth data collection and analysis of a broad range of complex engagement scenarios in a controlled simulation environment. As such, DIS has provided effective training and tremendous cost savings when applied to large-scale operational training exercises. Although the phase-out of DIS simulations will begin to occur in September of 1998, DIS will receive continued Department of Defense (DoD) support until the year 2000, at which time High Level Architecture (HLA) will become the required standard geared towards interoperability and re-use of models and simulations. In the meantime, testbed studies are being conducted to identify the limitations and benefits of DIS-compliant simulations such as the Aviation Testbed (AVTB), Land Warrior Testbed, and Mounted Warfare Testbed.

The cost savings that DIS has brought to the training arena also appeals to those seeking more cost-effective means of prototyping, testing, and evaluating weapons systems. For example, a major objective expressed in the "DoD Modeling and Simulation (M&S) Master Plan" (U.S. Department of Defense, 1995) is the development of the capability for credibly examining human / system performance and effectiveness in distributed simulation environments. To support this objective, communications infrastructures such as the Defense Simulation Internet (DSI) have been designed to support long-haul, multi-player interactive simulations. A more long-term objective is to transition to commercial services and operational communications capabilities to meet modeling and simulation needs. However, the challenges that face these network solutions (e.g., the need for additional features, latency reduction, bandwidth reduction, and security improvement) and their impact on accurate determinations of human performance and weapon effectiveness remain largely undocumented. Thus, it is not at all clear whether the DIS environment, as presently employed, can reliably support such test and evaluation efforts.

Even if great care is taken to use high-fidelity crewstation simulations and flight models, detailed out-the-window visual systems, and well-trained crewmembers, the accuracy and validity of conclusions drawn from some DIS exercise outcomes may be questionable. Two major sources of error that can greatly threaten the validity of distributed exercise outcomes are time delay (or latency) and clock error (Katz, 1995a & 1995b). The total time delay, t_D,[1] arises from the data processing and propagation delays as well as synchronization errors inherent in high-traffic, long-haul distributed simulation networks. Clock error, Δt_c,[2] in the dead reckoning solution arises when there is a discrepancy between the sending and receiving simulations regarding the absolute time corresponding to the event or entity-state information contained in a transmitted Protocol Data Unit (PDU).

In distributed exercises requiring precise close-quarter interaction, latency can introduce serious correlation errors. Latency imposes hard limits on the rates of relative motion between interacting entities. If latency-imposed thresholds are violated, correlation of the perceptions of behaviors, reactions, and counter-reactions between interacting entities will be lost (Foster & Feldman, 1995; Foster, 1997). Latency can also degrade the correlation between events in DIS exercises by introducing clock error into dead reckoning solutions (Katz, 1995a; Katz, 1995b; Saunders, 1995). The impact of latency on the dead reckoning solution depends upon how the time stamps are implemented, as is described below.

The DIS standard (Institute of Electrical and Electronics Engineers, 1996a) defines dead reckoning as a method of position / orientation estimation employed to limit the rate at which Entity State PDUs (ESPDUs) are issued. Thus, from the sender's end, dead reckoning is seen

[1] The total delay, t_D, is herein defined to include the time interval between the sender's time stamp and the point in time at which the receiver first uses the transmitted information.

[2] Clock error, Δt_c, depends upon how time stamps are handled. If the sender's time stamp is used, then Δt_c arises from any discrepancy between the clocks of the sender and receiver vis-à-vis an absolute time reference. If the sender's time stamp is ignored, then the sender's clock time is irrelevant and the magnitude of Δt_c is equal to the total delay, t_D.

as a method of conserving network bandwidth. What is not generally appreciated, however, is that from the receiver's end, dead reckoning is a means for achieving precision (Katz, 1995a). The receiver extrapolates the sender's entity state, using entity-state time derivatives to correspond to the time that has elapsed since the time indicated by the time stamp.[3] If the sender's time stamp and the receiver's clock are both synchronized against an absolute time reference, then using the sender's time stamp in the receiver's extrapolation will compensate for the total time delay t_D to a precision consistent with the dead reckoning approximation being employed. However if the receiver's clock time does not agree with the sender's in absolute terms, the extrapolated states as viewed by the receiver will disagree with those as viewed by the sender by an amount determined by the magnitudes of the time derivatives and the clock error, Δt_c. In typical DIS implementations (such as the experiment reported herein) the sender's time stamp is not used and is irrelevant. Instead, the receiver's time-of-receipt of the ESPDU is used in the dead reckoning computation (Saunders, 1995). By ignoring the sender's time stamp, the receiving simulator implicitly assumes that the time corresponding to the data in the ESPDU coincides with the time that it was received, and -- if the ESPDU were subjected to a delay of t_D seconds enroute to the receiver -- the clock time used for dead reckoning will contain an error, Δt_c, equal to the total latency, t_D.

The primary consequence of clock error in dead reckoning is that, at any given moment in time, different simulators participating in an exercise will have different representations of an entity's location in space. This position error increases as a function of the velocity of a simulated entity, such that simulations of rapidly moving objects (e.g., aircraft and missiles) are more significantly impacted. Given the potential for disagreement regarding entity locations among distributed simulators, it is likely that a simulated engagement *outcome* may also be perceived differently by the various participants.

The effects of such errors have been seen in a number of DIS exercises involving rapidly moving entities. Valentino, Lubbers, Thompson, Scribner, and Breeding (1996) stress the

[3] Two types of time stamps are used in DIS exercises, "relative" and "absolute." A relative time stamp is the sender's clock time corresponding to the PDU event, in the absence of any global time synchronization. Absolute time stamps are achieved through time-synchronization of the networked simulators, using techniques such as Global Positioning System (GPS) synchronization (Katz, 1995b; Saunders, 1995).

importance of accounting for and reducing delay in DIS, and note its potential to seriously impact such exercises (assuming that the sender's time stamps are ignored). As evidence, they point to the Warbreaker Zen Regard exercise in which network latencies resulted in "missile failures...target 'jumping' or 'warping', ...and network timeouts." In a study of DIS ability to support test and evaluation efforts for the AIM-9M missile, McKee (1997) examined an air-to-air engagement environment and concluded that simulation errors were too large and created discrepancies in perceived simulation outcomes (i.e., whether or not a target had been "killed") between two simulation nodes. One author of this report personally observed a DIS exercise wherein surface-to-air missiles (Patriots) were launched against incoming tactical ballistic missiles (TBMs). In that exercise, limited network bandwidth caused significant transmission delays leading to serious outcome discrepancies. The Patriot simulation detected the incoming TBMs and successfully launched in an effort to intercept them. Operators of the TBM simulation, however, noted that they did not perceive the launch of the Patriot missiles until well *after* their TBMs had impacted their targets. This difference in perceived engagement outcome clearly illustrates the potential detrimental effects of latency and clock errors in DIS exercises that rely upon simulated engagement outcomes to quantify performance and to drive critical engineering or acquisition decisions.

The importance of timing in DIS was recognized early, and the requirement for a time stamp has been included in the DIS data exchange standard from its first version (Institute of Electrical and Electronics Engineers, 1993). In a further effort to minimize errors arising from delays, IEEE Standard 1278.2 (Institute of Electrical and Electronics Engineers, 1996b) includes a quality-of-service requirement regarding acceptable delay in DIS exercises. The IEEE Standard 1278.2 distinguishes between "loosely coupled" and "tightly coupled" simulation environments in regard to delay tolerances as follows:

Loosely Coupled: A condition that exists when simulation entities are not involved in very close interaction such that every action of an entity does not need to be immediately accounted for by the other entities. Two tanks moving over terrain five miles apart from each other is an example a loosely coupled situation. p. 4.

Tightly Coupled: A condition that exists when simulation entities are involved in very close interaction such that every action of an entity must be immediately accounted for by the other entities. Several tanks in close formation involving rapid, complicated maneuvers over the terrain is an example of a tightly coupled situation. p. 6.

For loosely coupled conditions, IEEE Standard 1278.2 specifies 300 ms as the maximum acceptable delay from the input of the Transport Layer at the sending simulator to the output of the Transport Layer at the receiving simulator. For tightly coupled conditions, this tolerance is reduced to 100 ms. The impact of even a 100 ms delay, however, may be quite pronounced in a tightly coupled simulation with fast moving entities if time stamps are ignored. Saunders (1995) illustrates the relative impact of various delays on position errors for a number of different entity types. For a tank traveling at 100 km/hr, an 85 ms error results in a position error of only 2.36 m. However, this error increases to 23.61 m for an aircraft traveling at 1000 km/hr and 94.44 m for a missile traveling at 4000 km/hr. Even for simulations that adhere to the IEEE delay standard for tightly coupled systems, errors of these magnitudes can be expected with today's typical DIS exercise implementations. Hence, concerns regarding the adequacy of current DIS implementations for supporting critical human / system performance decisions, such as those expressed in the "DoD Modeling and Simulation (M&S) Master Plan" (U.S. Department of Defense, 1995), are justifiable from the perspective of position errors alone.

Another potential threat to validity in the DIS environment is the extent to which human behavior / performance is accurately represented in computer-generated models. Objective #4 in the DoD Modeling and Simulation (M&S) Master Plan (U.S. Department of Defense, 1995) requires the development of authoritative representations of human behavior -- specifically, human capabilities and limitations; individual and group performance; effects of organizational configuration and environment of performance; command, control, and communications; and doctrine and tactics. Although many simulations currently rely on a constructive model or computer-generated force (CGF) instead of using a human operator in the loop, these representations are often extremely limited due to a lack of theoretical robustness in this area and the effects of context on human performance. As Benslay (1996) points out, these constructive models are often ineffective because they fail to exhibit sufficient human behaviors. Thus, when facing a human adversary in a DIS exercise, CGF models are often at a disadvantage.

Developing a single algorithm to model human behavior in complex environments is difficult at best. Considering the complex nature of human cognitive / perceptual processes, it is easy to understand why a CGF attempting to model human behavior may fall short. Although cognitive process modeling tools such as OMAR (Operator Model Architecture) and SOAR (State, Operator and Result) have been successful in predicting human behavior based on decisions that the subject faces in a given task, they require a significant investment of time to develop, and thus, are often overlooked for most DIS exercises. As a result, computer algorithms for modeling weapon system behavior (including the man-in-the-loop) in today's DIS environments are generally simple, rigid, and are based on strict mathematical computations. Actual human performance in operating these weapon systems is far more inconsistent, flexible, and adaptable. Often, in addition to applying defined rules to make a decision, human operators are guided by past experience, hunches, and even political consequences in their decision making. Thus, although results from DIS exercises using a human in the loop and those using a CGF are often assumed to be equally valid, human operator and CGF performance may actually look quite different.

The purpose of the effort described here was to demonstrate and quantify the impact of protocol data unit (PDU) transport delay and entity control (human vs. constructive) on the outcome of DIS engagements involving tightly coupled simulations. In particular, researchers were interested in determining how these factors inherent to DIS might affect the validity of human / system performance evaluations.

Approach

In an effort to better control and manipulate PDU transport delay and entity control, this demonstration was conducted in a laboratory environment. The simulation scenario selected for examining these issues consisted of a surface-to-air missile (SAM) engaging a fighter aircraft. This tightly coupled simulation environment with high-velocity entities represents the type of scenario most vulnerable to simulation delays. In addition, it provides a scenario that easily accommodates either constructive threat modeling (i.e., computer-controlled engagement rules) or human-in-the-loop decision making. To support this simulation scenario, an air defense system simulator and a fighter aircraft simulator were configured in a locally distributed environment within Armstrong Laboratory's Crew-Centered Design Technology laboratory at Wright-Patterson Air Force Base. A series of simulated engagements between the air defense system and aircraft were performed. The resulting engagement outcomes were then evaluated as a function of PDU transport latency ("Delay") and human vs. constructive control of the air defense simulation ("Threat Control"). More detailed descriptions of the scenario and the experimental design are provided below.

Scenario

The engagement scenario began with the aircraft flying at 22,000 ft at 480 kts. The planned route of flight included a series of turns intended to navigate the aircraft around three air defense sites known to possess generic medium-range surface-to-air missiles (SAMs). On each trial, however, an additional air defense site (the "pop-up" threat) appeared on the planned route and launched a missile. (See Figure 1). The pop-up SAM had a maximum effective range of 13 NM and a maximum velocity of approximately 1200 m/s. Both the planned route and the location of the pop-up threat varied across trials, resulting in a different

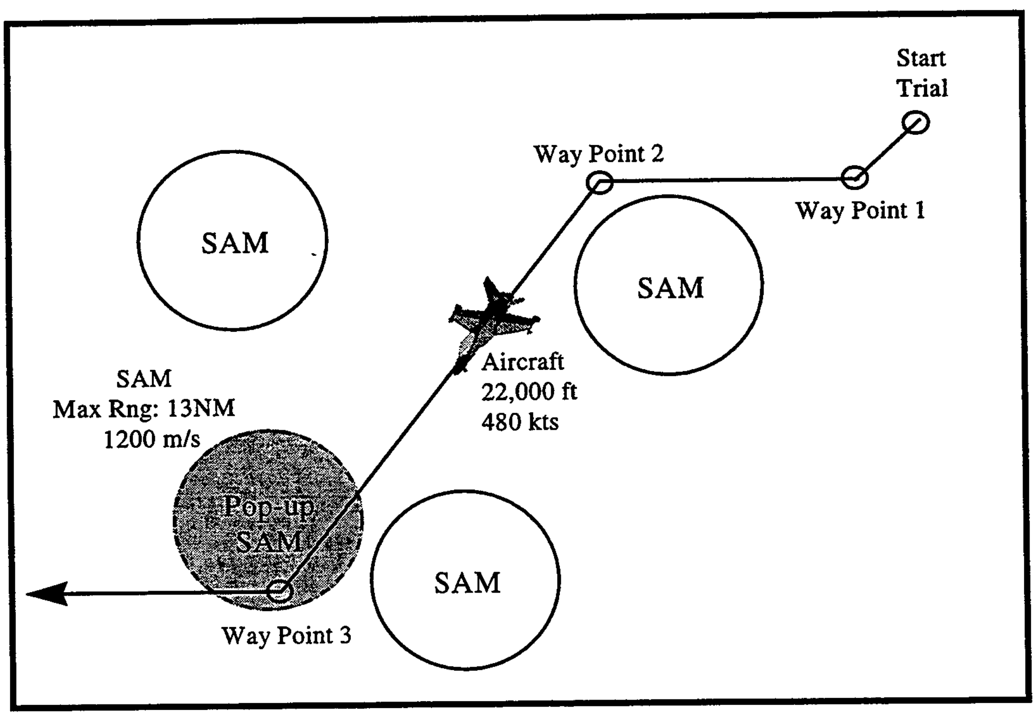


Figure 1. Simulation scenario

route / threat location combination for each of 16 trials. The pop-up threat required the pilot to perform countermeasures in order to decrease the probability of being intercepted by the missile. Countermeasures available to the pilot included maneuvering the aircraft such as to limit the ranging information available to the air defense radar and releasing up to two bundles of chaff. Because the aircraft simulation used in this study had a limited out-the-window visual capability, the scenario assumed that cloud cover precluded visual acquisition of the missile, requiring the pilot to depend solely on the aural and visual cues from the radar warning receiver in the cockpit for situational awareness regarding the air defense system.

Experimental Design

Both study variables, Delay and Threat Control, were examined at two levels, creating a simple 2 x 2 experimental design (see Table 1). The baseline or "0 ms" Delay condition consisted of only that delay imposed across the local Ethernet between the two systems (less than 2.0 ms), with no additional transport delay imposed. In the "100 ms" Delay condition, a simulated transport delay of 100 ms (the maximum delay allowable for tightly coupled simulations as stated in the DIS standard) was artificially introduced using the Delay Manager software described later in the Method section.

Table 1. Experimental Design

Transport Delay	Air Defense Threat Control	
0 ms	Live	Constructive
100 ms	Live	Constructive

To examine potential differences in simulation outcome due to the use of a constructive control model, two Threat Control conditions were also implemented. These conditions were only examined with respect to the missile simulation, as it allowed for a more straightforward implementation of a constructive model. In the "constructive" Threat Control condition, a computer algorithm determined when the missile would be launched. This simple algorithm was set such that the missile simulation would take the first opportunity to launch once it determined that the aircraft was within its high lethality range. Such an algorithm was considered to be representative of most threat models used in today's DIS exercises. In the "live" Threat Control condition, a human operator monitored air defense system displays, made a decision when to launch the missile, and manually initiated the launch sequence. Live vs. constructive control of the aircraft was not manipulated. In all cases, a human operator controlled the aircraft simulation. This resulted in a more realistic engagement scenario, allowing the use of complex maneuvering and the release of chaff in an effort to defeat the missile.

SECTION 2.
METHOD

Participants

Eight pilots served as subjects in this study, performing the flight task and missile avoidance tactics in the flight simulator. All subjects were currently or formerly rated aircraft pilots in the US military. These subjects were drawn predominantly from the F-16 community, however, they had combined experience in over 25 different military aircraft. The experience of these aviators ranged from 1300 hours of military flight time for the least experienced pilot to over 5000 hours for the most experienced. The average number of flying hours across all subjects was slightly over 3000, representing a very experienced subject pool. The air defense operator was also quite experienced, with 13 years experience as an Army officer and 10 years as an air defense simulation developer. His combined military and civilian air defense tactical employment experience included over 14 years of Air Defense tactical firing doctrine analysis, simulation, and field exercise support, focusing on the PATRIOT and HAWK air defense missile systems.

Apparatus

Air Defense Simulation

The air defense simulation platform in this study consisted of the Reconfigurable Tactical Operations Simulator (RTOS) developed by SAIC. Evolved from the Patriot Tactical Operations Simulator (PTOS), RTOS is a DIS-compliant missile system simulator whose operator console and radar / missile properties can be reconfigured to represent a variety of missile systems. The operator workstation, shown in Figure 2, consists of a Silicon Graphics Indigo II workstation with a 19" color monitor and three gas-plasma flat panel touch screen displays that serve as the operator control interface.

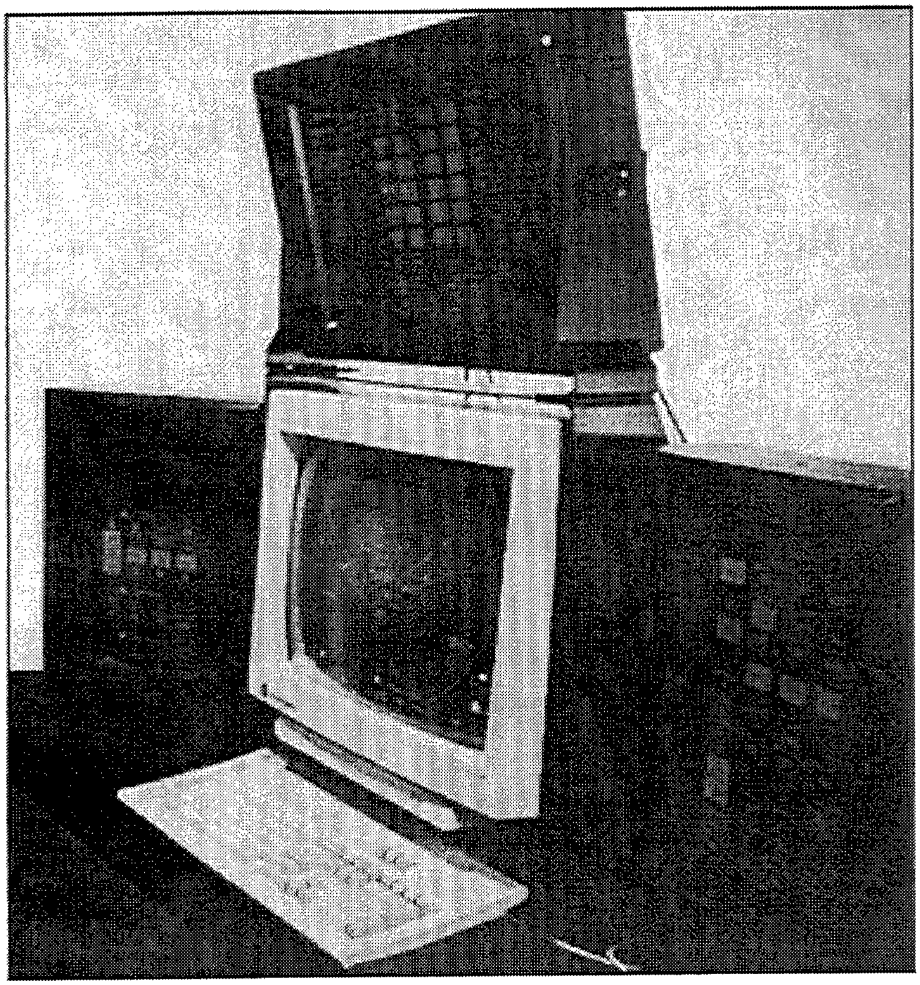


Figure 2. Reconfigurable Tactical Operations Simulator (RTOS)

The RTOS software used for this study was an unclassified air defense engagement model with a battalion level command and control system for air picture coordination and engagement operations. The air defense system was customized with range and emissions characteristics to represent a generic medium-range SAM threat. To provide an active countermeasure for the aircraft pilot, a rudimentary model representing the degradation effects of chaff releases on the air defense radar was also implemented. This was a probabilistic model only and did not model first principles of the radar / chaff / atmosphere interactions. The radar type in the Emissions PDU issued for each fire unit radar was modified such that the aircraft RWR simulator could detect changes between surveillance and illumination modes and generate appropriate radar warning indications.

Aircraft Simulation

The aircraft simulation platform employed in the study was the Engineering Design Simulator (EDSIM) shown in Figure 3. Developed by Veda Inc., the EDSIM is a single-seat, rapidly reconfigurable aircraft cockpit simulator that incorporates a scaleable hardware and software architecture for the rapid prototyping of cockpit designs. This three-layer architecture consists of the simulation system software, the simulation application software, and the cockpit application software. This EDSIM is hosted on a series of Silicon Graphics computers and incorporates MÄK Technologies' VR-Link as the DIS interface software. For the purpose of this study, cockpit controls / displays were limited to only those necessary to perform the required flight and countermeasure activities. These included a stick and throttle, a head up display (HUD), a horizontal situation indicator, a radar warning receiver (RWR), a chaff release button, and a chaff stores display. The aero model driving the simulation was the Silicon Graphics *Flight* demonstration with the selectable F-16 representation. The

Figure 3. Engineering Design Simulator (EDSIM)

a priori assumption used in regard to the lethality radius of a generic surface-to-air missile against a fighter aircraft was 35 m. The EDSIM was programmed with this vulnerability radius such that the EDSIM declared that it had been killed if it perceived a detonation within 35 m of its center point.

Delay Manager

To artificially represent and control the transport delay associated with distributed simulation, a "Delay Manager" software module was developed to regulate the flow of PDUs between the RTOS and EDSIM. The Delay Manager, which ran concurrently on the RTOS host computer, was designed to pass through PDUs after a specified delay had expired. It was also configured to capture a log of the PDU stream between EDSIM and RTOS, with time tags showing the actual delay imposed on each PDU. The transport delay value used during the experiment trials was either 0 ms (representing the no delay condition) or 100 ms. Coupled with the internal *processing* delay inherent in the simulators, the total of which averaged 30 ms, and a delay of 2 ms associated with communication across the local Ethernet; this imposed delay resulted in *total* delay t_D of approximately 32 ms and 132 ms in the 0 ms and 100 ms transport delay conditions, respectively.

Simulation Timing

For the purpose of this study, a decision was made to use a version of VR-Link that ignores the sender's time stamps stored in the PDUs and substitutes the times-of-receipt instead. As discussed earlier, this introduced into the dead reckoning extrapolations a clock error equal to the total delay. Although this implementation provides the worst case dead-reckoning performance in assessing the effects of delay, it is most representative of the implementation used in today's DIS exercises (Saunders, 1995).

Although the sender's time stamps were ignored in the simulation, the clocks of the two simulation computers were synchronized to within 1 ms. This allowed a master time reference to be established against which all system delays could be measured, such that researchers could better characterize the delay environment.

Procedure

Prior to the experimental trials, each subject (pilot) was briefed on the purpose of the study and on unclassified tactics for defeating a radar-guided SAM. The subject was then oriented to the flight simulator and shown the subset of controls and displays critical to performing the required flight task. Once familiar with the cockpit, the subject was asked to review a simple map outlining a route through an area of known threats and was given an opportunity to practice flying the route in the simulator. This route started at a predefined waypoint and did not require the subject to perform a takeoff or landing. Once the subject felt comfortable with the flight task, he was asked to fly an additional set of trials in which a pop-up SAM generated by the RTOS was introduced at a point along the flight path. On these trials, which portrayed the actual experimental conditions, the subject was asked to perform countermeasures in an effort to defeat the missile. Countermeasure tactics included maneuvering the aircraft to a heading tangential to the azimuth of the pop-up threat, maintaining airspeed, and releasing up to two bundles of chaff. The subject was allowed to fly as many practice trials as he wished until he felt comfortable with performing the task, at which point he began the series of 16 experimental trials.

The primary stimulus to the subject in this task was the RWR, which alerted the subject to missile site type, location and mode. Throughout the trials, the subject heard a steady pulse aural tone (approximately 0.5 Hz) in his headset, which indicated the presence of search radar. These radar sites were also identified with a symbol on the RWR visual display. As the aircraft entered a pre-defined lethality radius of the simulated pop-up SAM, the RTOS would begin sending emission PDUs from a previously silent radar site indicating that a pop-up radar had initiated a track mode. This generated an immediate change in the aircraft RWR aural tone, which changed to a rapid pulse (approximately 5 Hz). On each trial, within 5 to 10 seconds of switching to track mode, the RTOS launched a missile. The aircraft RWR indicated a missile launch by presenting a continuous tone at a higher frequency and by drawing a diamond around the symbol representing the launching missile site on the visual display.

On each trial, the aircraft was engaged by only one missile. When being engaged by a SAM, the subject was asked to perform the countermeasure tactics as briefed. Each trial was terminated at the end of the missile flyout, and the EDSIM performed a kill determination. If the EDSIM calculated the missile's position to be within 35 meters of the aircraft at the time of detonation, it scored the engagement outcome as a successful intercept or "kill." However, if the aircraft / missile separation at the time of detonation was calculated by the EDSIM to be greater than 35 meters, the trial outcome resulted in a "no kill." After each trial, the subject was given feedback regarding the trial outcome. Each subject performed a total of sixteen trials. Using a fully randomized, repeated measures design, subjects performed four trials in each of the four experimental conditions.

SECTION 3.
RESULTS

Each of the eight subjects successfully completed all 16 experimental trials, yielding a total of 128 trials. Because subjects were given an opportunity to employ countermeasures, which made the task more interesting and realistic, not all trials resulted in the aircraft being intercepted. On 13 of the 128 trials, countermeasures caused the air defense system to drop the radar track. Because these trials resulted in intentional detonation of the missile at a non-intercept position (missile abort), results from these trials were not analyzed. An initial analysis of the remaining data revealed that one of the 16 flight paths presented to subjects resulted in number of anomalous results (four of eight trials led to position errors greater than two standard deviations above the mean for all trials). In addition, two trials resulted in detonation PDUs being dropped (i.e., the PDUs were issued by the missile simulation but were not received by the aircraft simulation). Thus, data from these trials were deemed to be unreliable and were eliminated from the analyses reported below. Analyses of data from the remaining 105 trials focused on determining how Engagement Outcome and missile Miss Distance at intercept varied as a function of Delay (0 vs. 100ms) and Threat Control (live vs. constructive).

Engagement Outcome

Because the kill assessment in DIS exercises is determined by the entity being attacked, Engagement Outcome (i.e., whether the aircraft survived the missile attack) was determined from the perspective of the aircraft. This binary outcome, which resulted in a "kill" if the aircraft simulation perceived a detonation within 35 m of the aircraft center point, reflected the calculated difference between the missile coordinates contained in the missile detonation PDU and the aircraft position at the time the detonation PDU was received by the aircraft simulation. The results of this assessment are shown as a function of Delay and Threat Control conditions in Table 2.

Table 2. Simulation Outcome by Condition

Transport Delay	Air Defense Threat Control	
	Live	Constructive
0 ms	N=28 22 kills P_k = .7857	N=24 16 kills P_k = .6667
100 ms	N=26 0 kills P_k = .0000	N=27 0 kills P_k = .0000

Within the "0 ms" Delay condition, the live Threat Control condition generated 22 kills, resulting in a probability of kill (P_k) of .7857 across 28 trials. In the constructive Threat Control condition, P_k was reduced to .6667 across 24 trials. Averaged across Threat Control conditions, this resulted in a P_k of .7308 when no additional transport delay was introduced. Within the "100 ms" Delay condition, however, not a single trial resulted in a successful intercept, resulting in a P_k of .0000, regardless of air defense Threat Control. A Chi-Square analysis of Delay yielded a value of 60.70, df=1, p.<.0001; indicating a statistically significant impact of Delay on the probability of kill (P_k). A similar Chi-Square analysis examining the effect of Threat Control did not yield a significant effect (X^2=.931, df=1, p.<.335), suggesting that control of the missile launch (live vs. constructive) did *not* significantly impact P_k.

Miss Distance At Intercept

To better understand why Engagement Outcome varied as a function of Delay, Miss Distance was examined. To isolate the effects of Delay from any error due to dead reckoning algorithms, Miss Distance within a trial was examined for only missile detonation PDUs. These events were chosen for analysis because they are static events (i.e., a detonation happens at a single point is space) requiring no dead reckoning by the aircraft simulation. Furthermore, because the air defense simulation detonates the missile at the very position it perceives the aircraft to be, the miss distance at detonation provides a good estimate of the simulation position error.[4]

Like Engagement Outcome, Miss Distance was calculated based on the aircraft simulation's perspective. That is, Miss Distance was calculated as the difference between the missile coordinates contained in the missile detonation PDU and the aircraft position at the time the detonation PDU was received by the aircraft simulation. Mean Miss Distance across trials is shown in Figure 4 for each of the four experimental conditions.

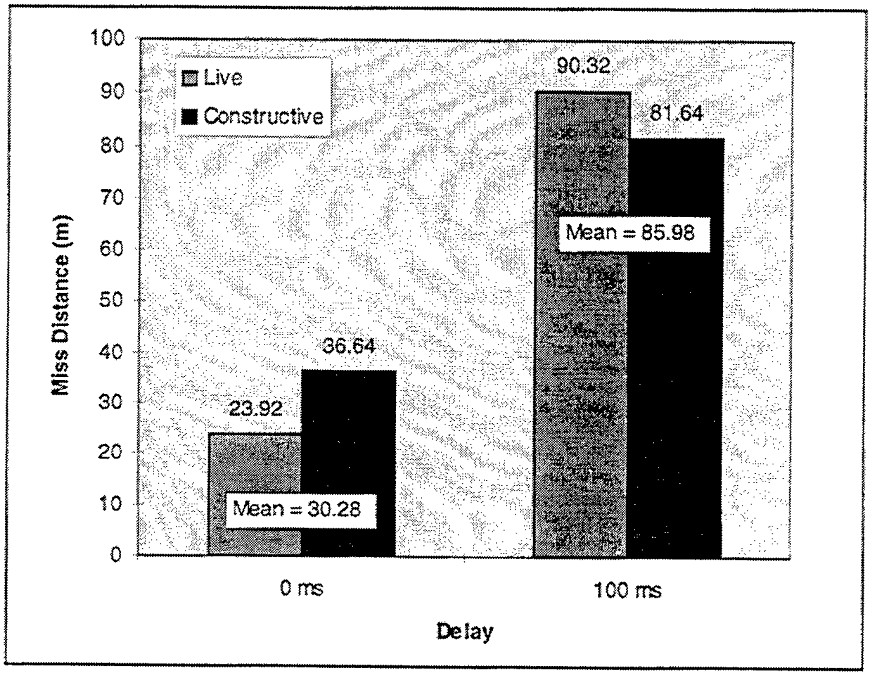


Figure 4. Missile Miss Distance at intercept by condition

Collapsed across Threat Control conditions, Miss Distance averaged 30.28 m and 85.98 m in the 0 ms 100 ms Delay conditions, respectively. A two factor analysis of variance (ANOVA)

---

[4] Use of the sender's detonation time stamp and recent aircraft position history would have permitted better correlation between aircraft position and missile position at time of detonation. However, this was not done because it is not supported by the version of VR-Link used, nor is it done in the typical DIS exercise implementation.

found this to be a significant main effect of Delay (F(1,7)=35.83, p.<.001), indicating that the 100 ms Delay condition resulted in a significantly higher Miss Distance than the 0 ms condition. Across Delay conditions, Miss Distance averaged 57.12 m and 59.14 m in the live and constructive Threat Control conditions, respectively. Neither this difference due to Threat Control (F(1,7)=.13, p.<.731) nor a Threat Control x Delay interaction (F(1,7)=1.16, p.<.317) proved to be statistically significant.

Additional Observations

Human vs. Constructive Model Performance

Although Threat Control had no significant impact on either Engagement Outcome or Miss Distance, the human operator was observed to conduct the missile launch differently than the computer model, waiting slightly longer before launching the missile. This observation was confirmed by examining the total missile flight time by Threat Control condition. Figure 5 shows the average time of missile flight for human-controlled and computer-controlled launch sequences. The average missile flight in the constructive Threat Control condition

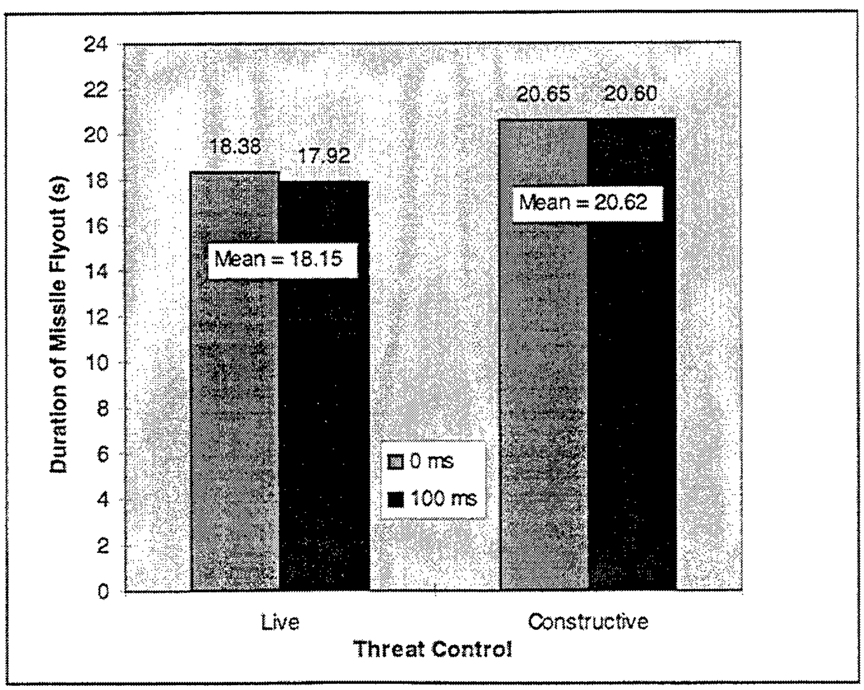


Figure 5. Time of missile flight to intercept by condition

lasted 20.62 s. In the live Threat Control condition, missile flight time was reduced to 18.15 s, as the human operator waited until the aircraft flew deeper into his lethal radius before initiating the missile launch sequence, resulting in a closer range to target at launch. An ANOVA found this to be a significant main effect of Threat Control on missile flight time ($F(1,7)=27.38$, $p<.001$). Neither an effect of Delay on missile flight time ($F(1,7)=1.27$, $p<.297$) nor an interaction ($F(1,7)=.25$, $p<.630$) were observed.

Total System Delay

As described in the Method section, the *total* system delay reflected not only the 0 ms or 100 ms transport Delay conditions, but also the delays associated with internal processing of data and transmission across the Ethernet. The total system delay was identified by measuring the mean elapsed time between the generation of the detonation PDU in the missile simulation and the processing of that PDU in the aircraft simulation. As shown in Figure 6, the average total system delay across all trials was found to be 34.16 ms and 131.20 ms in the 0 ms and 100 ms Delay conditions, respectively. Factoring out the transport delay imposed in the 100 ms Delay condition, these values approximate the 32 ms observed in the preliminary measurements.

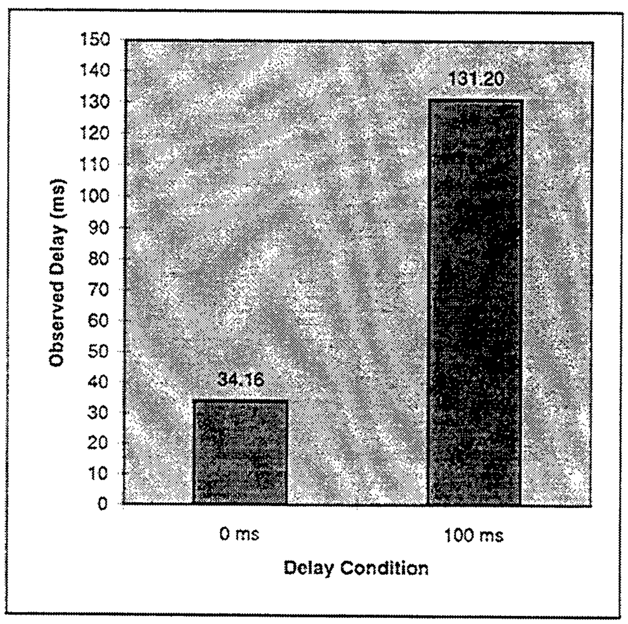


Figure 6. Observed total system delay by condition

SECTION 4.
DISCUSSION

Transport Delay

As illustrated above, the introduction of only a 100 ms transport delay had a tremendous impact on the outcome of the simulated engagement between missile and aircraft. The effect was the reduction in missile Pk from .7308 with no transport delay to .0000 when the delay was imposed. Clearly, DIS artifacts such as PDU transport delays and the resulting position errors can dramatically affect data generated in a test. When critical decisions are to be based on those data (e.g., selection of system / component alternatives in the acquisition process), the potential for costly errors is significant.

When the sender's time stamp is ignored, the delay in DIS (whether caused by transmission of data among distributed systems or by internal processing of PDUs) manifests itself in position error. Resulting problems, such as target "jumping" on visual displays, have been well documented -- however, the potential for impacts to the actual outcomes of DIS exercises seems to be a more critical problem. In this particular simulation, position error and corresponding missile Miss Distance increased from 27 to 85 m as transport delay increased from 0 ms to 100 ms. With a representative vulnerability ring of 35 m around the aircraft used for kill determination, this position error was directly responsible for the change in Engagement Outcome. It is likely that this position error would have been even greater in a closed loop system such as that employed in McKee (1997), in which the two simulation entities react to each other. In the current study, the aircraft was reacting to a general missile threat, but only the missile was making course corrections based on the aircraft maneuvers. Even in this case, position error artificially reduced the effectiveness of the missile when attempting to achieve an intercept.

Although this study demonstrated serious limitations to the validity of simulated engagement outcomes obtained in a tightly coupled distributed exercise with high-velocity entities, steps can be taken to greatly reduce this problem. The use of absolute time stamps on a network architecture with global positioning system (GPS)-coordinated system clocks may offer the best chance to compensate for the effects of latency. With the use of GPS synchronization to implement an absolute clock for an exercise, absolute time stamps can yield accuracies of 2 μs (Katz, 1995a; Saunders, 1995). With a precise time reference, more precise extrapolations can be applied to entity position data. The technology currently exists and is affordable, but does require some minor modification to existing simulation facilities -- and not all simulations have a DIS interface that can support time stamps. Although implementation of absolute timing can reduce clock errors to insignificance (Katz, 1995b), the consequences of clock errors do not appear to be widely appreciated (Saunders, 1995), and there is nothing to suggest that implementation of absolute time stamps will be widely adopted in the near future.

Further down the road, the delay problem will also have to be addressed for HLA exercises. Future HLA federations will have a choice of time management options that may enforce a more rigorous time advance mechanism, regulated by the Run Time Infrastructure (RTI). When RTI implementations are capable of supporting real time interactions of entities with controlled time step advances, the position perception problem can be better controlled by ensuring a consistent view of entity positions among all federates. Even in HLA, however, transport delays will still occur in long haul networks. For high speed maneuvering entities such as missiles and tactical fighters, dead reckoning algorithms will still be challenged. One solution lies with HLA object management. If threat / weapon models are also co-located with the aircraft simulators, ownership of missile attributes can be transferred from the threat simulator to the aircraft simulator after launch so that missile flyout can be generated locally -- thereby eliminating transport delay of entity state data.

In the meantime, what can we do to increase the validity of engagement outcomes within today's commonly used network configurations? Reduction of clock error requires a use of time stamps that account for network delays. The Network Time Protocol (NTP) specification (Mills, 1992) and its infrastructure have been developed and evolved over the past fifteen years to support just that end. NTP is used to synchronize the time of a computer client or server to another server or reference time source, such as a radio or satellite receiver or modem. It provides client accuracies typically within a millisecond on Local Area Networks (LANs) and up to a few tens of milliseconds on Wide Area Networks (WANs) relative to a primary server synchronized to Coordinated Universal Time (UTC). The NTP subnet presently includes over 50 public primary servers synchronized directly to UTC by radio, satellite or modem and located in North America, Europe and Asia.

Threat Control

Although the human operator clearly behaved differently than the constructive model in initiating missile launches (i.e., waiting until the aircraft was at a closer range), this had no significant impact on Miss Distance or Engagement Outcome. This may be due in large part to the fact that, in an effort to achieve experimental control, the air defense operator was given fairly little latitude in how to engage the aircraft. He was limited to a single missile shot with no opportunity to perform a salvo, ripple, or shoot-look-shoot engagement; and he was not permitted to employ tactics such as manipulating power settings or launching prior to illumination. Furthermore, the operator performed the task flawlessly across all experimental trials, consistently launching the missile only when the aircraft was within the missile's high-lethality radius. In this sense, his performance looked identical to that of the constructive model. The only tactic available to the human operator was to attempt to decrease the range of a single-shot engagement by waiting until the aircraft flew *deeper* into the missile's high lethality radius. Looking at only the 0 ms Delay condition, in which results were not confounded with Delay effects, this tactic showed a trend toward decreasing Miss Distance at intercept and a higher probability of kill. However, it apparently did not enhance the effectiveness of the semi-active homing process employed by the missile guidance model, as it failed to yield a statistically significant improvement in performance.

These results suggest that in a simple engagement level simulation, the use of constructive models to represent human-in-the-loop performance may be acceptable in terms of yielding equivalent engagement outcomes. However, the differences in human vs. constructive model behavior observed here illustrate the potential problems with simple constructive models. As the simulation environment becomes more complex, including scenarios where human operators would have a greater degree of freedom in decision making and procedures, it is increasingly likely that increasing behavioral differences will lead to more pronounced differences in simulation outcome.

SECTION 5.
CONCLUSION

Clearly, typical exercise implementations in today's DIS environment are better suited for representing loosely coupled simulations consisting of slower-moving, less dynamic entities whose position data are not significantly impacted by typical transport delays. This effort focused on the effect of clock errors arising from latency and the typical misuse of time stamps in DIS exercises. However, the correct use of time stamps along with absolute timing can reduce clock errors to insignificance -- there is no technical reason to tolerate this source of error (Katz, 1995a; Katz, 1995b). The *real* technical problems associated with latency are in regard to the limits it imposes on the range of motions over which entities in a distributed simulation can interact (Foster, 1997); dead reckoning algorithms can actually exacerbate jitter and exaggerated motions if latency is too large to support plausible interactions to occur for highly dynamic entities (Foster & Feldman, 1995). Therefore -- notwithstanding the existence of mature technology to eliminate clock error -- there remain formidable technical challenges to improving the state of distributed simulation such that it is equally capable of supporting tightly coupled simulations involving high-velocity entities.

Today's DIS exercises typically do not achieve the accuracy necessary for a human-engineering or test-and-evaluation environment. If we are to draw conclusions regarding human / system effectiveness based on the performance data and outcomes of DIS exercises, we must first be assured that these data are valid representations of human / system performance. This requires accurately modeling both the performance of relevant system hardware (e.g., tanks, planes, missiles) and the performance of relevant human operators in the simulated world.

REFERENCES

Benslay, J. L., (1996), "Domain Independent Framework for Developing Knowledge Based Computer Generated Forces," Rept No. AFIT/GCS/ENG/96D-04, Air Force Institute of Technology, Wright-Patterson AFB, OH.

Foster, L., and Feldman, P., (1995), "Limitations of Interactive Behavior for Distributed Interactive Simulation," *Proceedings of the 13th DIS Workshop*, Paper 95-13-027, p. 175-189.

Foster, L. (1997), "Conditions for the Positive Correlation of Dynamic State Data Between Objects in a Distributed Simulation Environment," *1997 Fall Simulation Interoperability Workshop*, Paper 97F-SIW-071.

Institute of Electrical and Electronics Engineers (1993), "IEEE Standard for Information Technology -- Protocols for Distributed Interactive Simulation Applications," *IEEE Std 1278.1-1993*.

Institute of Electrical and Electronics Engineers (1996a), "IEEE Standard for Distributed Interactive Simulation -- Application Protocols," *IEEE Std 1278.1-1995*.

Institute of Electrical and Electronics Engineers (1996b), "IEEE Standard for Distributed Interactive Simulation -- Communication Services and Profiles," *IEEE Std 1278.2-1995*.

Katz, A. (1995a), "Precision under DIS," *Proceedings of the 12th DIS Workshop*, Paper 12-95-090, p. 531-535.

Katz, A. (1995b), "Event Correlation for Networked Simulators," *Journal of Aircraft*, 32 (3): p. 515-519.

McKee, L. (1997), "Latency and its Effects on the Fidelity of Air-to-Air Missile T&E Using Advanced Distributed Simulations," *1997 Fall Simulation Interoperability Workshop*, Paper 97F-SIW-010.

Mills, D.L. (1992), "Network Time Protocol (Version 3) Specification, Implementation and Analysis," Network Working Group Report RFC-1305;[5] 113 pp.

Saunders, R. (1995), "It's About Time, It's About Space - Time and Space in DIS," *Proceedings of the 12th DIS Workshop*, Paper 12-95-015, p. 63-66.

U.S. Department of Defense, (1995), "Modeling and Simulation (M&S) Master Plan," DoD 5000.59-P.

Valentino, G. J., Lubbers, S. A., Thompson, S. T., Scribner, K. W., and Breeding, K. E. (1996), "Techniques for Reducing Latency in Distributed Flight Simulation," Paper AIAA-96-3542-CP, p. 600-610.

[5] Internet Requests for Comments (RFC) are available from the DDN Network Information center, SRI International, Menlo Park CA 94025, USA.

APPENDIX: NETWORK TIME PROTOCOL RESOURCES ON THE INTERNET

The Network Time Protocol (NTP) is used to synchronize the time of a computer client or server to another server or reference time source, such as a radio or satellite receiver or modem. It provides client accuracies typically within a millisecond on LANs and up to a few tens of milliseconds on WANs relative to a primary server synchronized to Coordinated Universal Time (UTC) via a Global Positioning Service (GPS) receiver, for example. Typical NTP configurations utilize multiple redundant servers and diverse network paths, in order to achieve high accuracy and reliability. Some configurations include cryptographic authentication to prevent accidental or malicious protocol attacks. The following provides a brief listing of NTP materials and \ or pointers to such materials available on the Internet, along with the relevant Universal Resource Locator (URL) address and some descriptive material drawn from the Internet web sites.

NTP Specification and Bibliography

The NTP specification and implementation has evolved over the last fifteen years to the current Version 3 of the protocol. This version includes significant enhancements in accuracy and reliability, as determined by experience in an estimated total of well over 100,000 clients and servers in the Internet, while retaining backward compatibility with previous versions. The formal specification of the NTP Version 3 protocol is contained in:

> Mills, D.L. (1992), "Network Time Protocol (Version 3) Specification, Implementation and Analysis", Network Working Group Report RFC-1305, University of Delaware, March 1992, 113 pp.
> (Abstract, Body, and Appendices available in PostScript format.)

http://www.eecis.udel.edu/~ntp/database/html_xntp3-5.90/biblio.html

Time Synchronization Server

This server provides the latest information on NTP and other related clock synchronization products.

http://www.eecis.udel.edu/~ntp/

Building and Installing NTP and Configuring Clients and Servers

The Building and Installing the Distribution page presents an overview of the procedures for compiling the distribution and installing it on a typical client or server. The build procedures inspect the system hardware and software environment and automatically select the appropriate options for that environment. In order to participate in the existing NTP synchronization subnet and obtain accurate, reliable time, it is necessary to construct an appropriate configuration file which establishes the servers and / or external receivers or modems to be used by this particular machine. Directions for constructing this file are in the Notes on Configuring NTP and Setting up a NTP Subnet page.

http://www.eecis.udel.edu/~ntp/database/html_xntp3-5.90/index.html

Public NTP Time Servers and Rules of Engagement

The lists of NTP public time servers provided represents the best information available at the current date. The list of primary (stratum 1) and secondary (stratum 2) designates the NTP time servers available for public access under stated restrictions. Each entry gives the host name, Internet address, approximate location and geographic coordinates (if available), synchronization source (stratum, type of radio or satellite receiver and host type), suggested service area, access policy (as notified) and contact name and e-mail address. Those servers known to be running NTP Version 3 are indicated as well. It is very important that potential clients avoid use of servers not listed as open access, unless approved first by the contact person.

http://www.eecis.udel.edu/~mills/ntp/servers.html