THE CAPABILITY OF THE DISTRIBUTED INTERACTIVE SIMULATION NETWORKING STANDARD TO SUPPORT HIGH FIDELITY AIRCRAFT SIMULATION

Edward P. Harvey
BMH Associates, Inc.
Norfolk, VA

Richard L. Schaffer
BBN Systems and Technologies
Cambridge, MA

ABSTRACT

How will U.S. tactical aviation forces train for future conflict? The prevailing budgetary climate will force a reduction in the frequency of training operations using actual equipment for some time to come. One cost-effective means for U.S. combat forces to conduct training is through the application of distributed simulation technology. A large scale simulation network which is based on the new Distributed Interactive Simulation (DIS) draft military standard for simulator networking and is accessible by the components of all three services will be the likely medium for conduct of this type training.

DIS networking protocols evolved from ground vehicle networking protocols developed during the U.S. Army/ DARPA SIMNET program. It is therefore understandable that some misconceptions may exist over the capability of DIS to provide sufficiently accurate vehicle position and orientation data for high performance aircraft simulation. High performance tactical aircraft simulation requires a high degree of vehicle position and orientation accuracy for conduct of fully effective training. Operational community acceptance is dependent upon the capability of a DIS network to support all potential high performance aircraft combat interactions including air-to-air missile engagements and air-to-air gunnery.

This paper will quantitatively detail DIS vehicle position and orientation accuracies throughout the potential range of simulated aircraft maneuvering capability. Entity State (position/orientation) Protocol Data Unit (PDU) transmission frequencies for differing order Dead Reckoning (DR) algorithms will be empirically derived for the F-16 fighter aircraft performing the dynamic Paris airshow flight routine. Average Entity State PDU transmission frequencies will be presented as a function of dead reckoning algorithm threshold values. This data will show the capability of the DIS networking standard to support high fidelity aviation training tasks, even those requiring precise real-time position updates such as air-to-air gunnery, while achieving significant network bandwidth reductions.

INTRODUCTION

You've just come from the Intell shop and there is no news on NAIL 21, the F-16 which was lost on the strike mission earlier in the day. Brief, launch, transit to the tanker, and ingress were no problem. It sure helped to have had 7 missions under your belt. The first few were a real zoo. Forming up a strike package from a half dozen different fields including the Navy carrier airwing assets, was something you didn't practice for during peacetime training. The morning mission had gone smoothly until the bandit warning from COUGAR, the AWACS controlling the strike. COUGAR had vectored your division of F-15C aircraft to intercept a flight of four bad guys and each of your wingmen splashed a single with AIM-7 shots in the face. Your Sparrow appeared to guide, but must have failed to fuse, and your MiG was the only one of the four to escape. You wanted to chase him down so bad you could hardly stand it but the mission comes first so you broke off the pursuit, reformed your division, and got a vector from COUGAR to rejoin the strike package.

Unfortunately, the target area was hot. Plenty of AAA, but that was going off well below everyone's altitude so it wasn't a big concern. Then at the roll-in point there was a flurry of SAMs launched nearly simultaneously and one managed to guide on NAIL 21. Either NAIL 21's EW pod was on the fritz, or maybe it was just a lucky ballistic shot. Nobody had any RHAW indications. At any rate there was a man in the chute over 200 miles deep in bad guy territory. Hopefully, that wouldn't happen this afternoon.

Back to the present. Time to brief your division for the afternoon go. This time you wouldn't be tied to the strike package and were going on a sweep. Hopefully, everything would go as well as this morning, except for the loss of the F-16 and not getting that kill. You've got to do this right. The boss is watching to see who are the performers. You stop for a moment's reflection. It is a weird feeling that this war is taking place in simulation facilities at twelve different USAF and USN bases in CONUS and Germany. Who would have ever thought....

DISTRIBUTED INTERACTIVE SIMULATION

The scenario recounted in the previous paragraphs is an example of the type of realistic team training which will be possible upon implementation of the Distributed Interactive Simulation (DIS) standard for networking simulators. The DIS standard for networking simulators, now released in draft form, is a direct outgrowth of the simulator networking protocol developed during the

DARPA/U.S Army SIMNET program. The University of Central Florida Institute for Simulation and Training (UCF IST) is the organization responsible for finalizing the content of the DIS standard for release to industry. UCF IST organized a series of semi-annual industry working groups to maximize participation in the DIS standard development process. These DIS working groups first convened in August 1989 to make recommendations for enhancing the existing SIMNET networking protocol to accommodate all classes of simulated vehicles and systems for network team training applications.

Over 70 point papers were submitted for review after the first DIS working group session. Many of these papers contained recommendations for alternate networking approaches; some addressed perceived weaknesses in the SIMNET protocols; and other papers advocated development of additional networking protocols. Many lively discussions by members of industry ensued which resulted in the submission of additional point papers to defend or attack positions taken, or recommend enhancements to the standard. As the DIS standard matured fewer point papers were submitted for each subsequent DIS working group session. Only 8 point papers were submitted for the working group session held in March 1991 with the majority of these papers addressing refinements to the existing draft DIS standard.

The DIS standard development process has been a success. A simulator networking standard has finally been developed by industry. It is interesting to note that after nearly two years of effort, the DIS standard is surprisingly similar to the original SIMNET protocols from which it evolved. This is remarkable since SIMNET networking protocols were developed primarily for an armored vehicle simulator network while the DIS protocols must support all classes of simulated vehicles, including high performance aircraft simulators.

The next section of this paper describes the DIS Entity State PDU and the role of dead reckoning. This is followed by a discussion on network protocol design considerations and a comparison of the performance characteristics of armored vehicles and high performance aircraft which influence network protocol design.

DIS Entity State PDU And Dead Reckoning

DIS simulators exchange messages with each other using a communications network. The message used to communicate vehicle state information is called the Entity State PDU. The DIS Entity State PDU is an enhanced version of the SIMNET Appearance PDU. The Entity State PDU contains all information needed to depict the originating vehicle. This information includes the type of vehicle and its position and orientation. In addition, it contains a set of parameters, called dead reckoning parameters, that are used to extrapolate the position and orientation of the vehicle into the future. This model of the future position of the vehicle is known as the dead reckoning model. When this model is computed in a remote simulator, it is referred to as the remote vehicle approximation (RVA). The procedure by which the dead reckoning model is calculated is known as the dead reckoning algorithm. The inputs to the dead reckoning algorithm are the position, orientation and dead reckoning parameters from the most recent Entity State PDU and the current time. DIS supports more than one kind of dead reckoning algorithm.

The Entity State PDU is not transmitted at the frame rate of the originating simulator. Instead, a simulator transmits an Entity State PDU only when the discrepancy between its own high fidelity model of its position and orientation and that of the dead reckoning model exceeds a certain threshold. In DIS, there are six separate thresholds; a positional threshold along each body axis, and a rotational threshold about each body axis. If any of these thresholds are exceeded, a new Entity State PDU is transmitted to the other simulators on the network.

Benefits of Dead Reckoning. The use of dead reckoning for networked simulation was pioneered by SIMNET. The principal motivation for the use of dead reckoning is to reduce the network bandwidth required to support a given application. From another viewpoint, dead reckoning increases the scale of exercise that can be supported on a given network. In the case of SIMNET ground vehicles, the reduction in network bandwidth requirements is dramatic; approximately 83% of network traffic is eliminated through the use of dead reckoning [Miller, et. al. 1988]. Another benefit of dead reckoning accrues to receiving simulators. In general, there is a computational cost associated with receiving and processing a PDU. By substantially reducing the rate at which PDUs are received, dead reckoning likewise substantially reduces this computational cost.

Finally, the dead reckoning model explicitly defines the state of remote vehicles in the intervals between the reception of Entity State PDUs. This provides an unambiguous definition of the state of remote vehicles, independent of the frame rate of the underlying simulators. Thus, it provides a means for simulators operating at different, and even irregular, frame rates to interact.

Tradeoffs Among Computation, Fidelity, and Network Bandwidth. Dead reckoning allows an explicit tradeoff to be made among three factors; network bandwidth requirements, computation performed in the RVA, and the fidelity of remote vehicle position and orientation. Network bandwidth is consumed by the required rate of transmission of Entity State PDUs. The cost of computation is the cost of computing the dead reckoning model for all remote vehicles within a range of interest. There is also a cost to encode the dead reckoning parameters on the part of the Entity State PDU sender. However, this cost is generally negligible since it need be done only once per Entity State PDU transmission, rather than once per frame per remote vehicle. Finally, the fidelity of representation of remote vehicles is simply determined by the dead reckoning thresholds for position and orientation.

There are two parameters which control these tradeoffs. One parameter is the choice of dead reckoning thresholds. These thresholds provide control of the tradeoff between network bandwidth and the fidelity of remote vehicle position and orientation representation. As the thresholds are decreased, the discrepancy between the dead reckoning model and the internal high fidelity model more quickly reaches its limit, the threshold; consequently, the rate at which Entity State PDUs are transmitted increases.

The other parameter controlling the tradeoffs is the choice of dead reckoning algorithm. This choice determines the tradeoff between network bandwidth and computational cost. More complex dead reckoning algorithms can better model the future path of the vehicle. Consequently, the discrepancy between the internal and dead reckoning models accumulates less quickly. The resulting reduced Entity State PDU transmission rate is bought at the cost of increased computation required by the more complex dead reckoning algorithm. This cost may be significant, since it must be computed each frame for each remote vehicle.

For any system design, and for any particular exercise, these two parameters may be adjusted to maximize fidelity within the limits set by the computational and network resources available.

Simulation Network Protocol Design Considerations

When SIMNET was developed it was postulated that networking armored vehicle simulators would be relatively simple compared to networking high performance aircraft simulators. This seems logical at first thought; after all, tanks are certainly much slower and appear to be less maneuverable than high performance aircraft. Therefore, an armored vehicle simulator network should require a lower entity state update rate than a high performance aircraft simulator network to provide sufficient fidelity for conduct of effective training. Based on a cursory inspection of the problem, it is natural to assume a network communications protocol developed for an armored vehicle simulator network would not have sufficient performance for conduct of effective training in a high performance aircraft simulation network. In fact, this is not the case for reasons which are discussed in the following paragraphs.

Vehicle Operating Characteristics . Each simulated entity must be able to detect the actions of other entities on the network in real time in order to provide adequate training fidelity. This may be achieved by the brute force method of transmitting entity state data at the frame rate of the transmitting simulator. In this manner the maximum delay which may be encountered before an entity action will be transmitted on the network is the frame period of the transmitting simulator. This method of ensuring real time entity state data transmission has its limitations for very large simulator networks. Affordable computational resources and available network bandwidth eliminate this method as a viable approach for a simulator network with many vehicles. A more clever approach than a high entity state data transmission frequency is necessary to provide real time entity state updates over a large scale simulator network. Below, we investigate whether one can take advantage of the characteristics of the various types of simulated vehicles which may be networked.

Vehicle Speed. Armored vehicles operate in a much slower speed regime than high performance aircraft by an order of magnitude or more. Armored vehicles also appear to be less maneuverable than high performance aircraft. It may seem that vehicle speed is the factor upon which the entity state transmission frequency of a simulator should be primarily dependent. If vehicle speed is a primary factor, high performance aircraft simulators would need to transmit entity state data at a much higher rate than armored vehicle simulators. Since aircraft are at least an order of magnitude faster than armored vehicles, the aircraft simulator entity state data transmission frequency should be at least ten times that of on armored vehicle.

This assumption is wrong. For example, a simulator using a simple dead reckoning algorithm can accurately calculate the future position of a simulated vehicle in steady state motion with a constant velocity regardless of the speed of the vehicle. As long as a simulated high speed aircraft remains in steady state flight there is no need for a high entity state data transmission rate to allow other simulated vehicles on the network to accurately calculate the future position of this simulated aircraft. Vehicle speed is not the dominant factor upon which to base design decisions regarding the entity state data transmission frequency of a simulator network.

Vehicle Maneuverability. Although an armored vehicle is maneuverable in two dimensions while an aircraft is maneuverable in three dimensions, there are several similarities between the behavior of the two types of vehicles. The maximum rate at which an armored vehicle can change its orientation through rotation about its vertical axis is similar to the maximum rate an aircraft can change its orientation by rolling about its longitudinal axis. An armored vehicle can only change its orientation at the maximum rate if it is rotating about a point in a constant position. An aircraft performing a maximum rate rolling maneuver about its longitudinal axis does not alter its flight path or future position appreciably. Although each type of vehicle is changing its orientation as rapidly as is possible, the future position of the vehicle is not appreciably affected; the tank remains in the same position and the aircraft direction of flight remains virtually constant.

In addition to the similarity in the rate at which the two types of vehicles can change their orientation, there is a similarity in the rate they are both able to change direction (turn) while in motion. A moving tank and an aircraft in flight are limited in their ability to change their direction of motion as a function of speed. The faster the tank and the aircraft travel within their respective speed ranges, the slower the possible turn rate. High performance aircraft are typically G-limited and rarely exceed turns which generate 9.0 Gs. An F-16 aircraft can generate a 9.0 G turn, which equates to a turn rate of approximately 22°/sec, at a speed of 440 knots. As aircraft speed increases the turn rate of the aircraft decreases at maximum G. At 600 knots an F-16 at 9.0 Gs turns at a rate of 16°/sec, a 27% decrease in turn rate from the possible turn rate at 440 knots airspeed. Tanks are able to generate similar turn rates as high performance aircraft as they are traveling much slower and do not need to generate much radial G to achieve equivalent turn rates. A tank traveling at 20 miles per hour need only generate 1/3 G to achieve a turn rate of 22°/sec.

There is much similarity between the rates at which armored vehicles and high performance aircraft can change their orientations and direction of travel. A networking protocol which takes advantage of the similar manner in which these two different classes of simulated vehicles change orientation and direction of motion should be capable of providing sufficiently accurate remote vehicle position and orientation updates over the network.

Tactical Considerations for Armored Vehicle Simulators

An armored vehicle crew increases their exposure to danger from adversary infantry, armored vehicles, and aircraft whenever they move their vehicle. Therefore, armored vehicles spend much of the time in prepared firing positions hidden from these foes. Entity state data which accurately describes a stationary armored vehicle's position and appearance can be transmitted at a relatively low frequency over a simulator network. If a vehicle remains stationary and does not change its appearance in any way to an outside observer it only need transmit one entity state information package to the rest of the network at the moment the vehicle assumes this state to provide "ground truth". In practice, the DIS and SIMNET networking protocols require stationary vehicles to transmit an entity state update at least every 5 seconds so that simulated vehicles joining the network must wait only 5 seconds to receive all remote vehicle data necessary to calculate "ground truth".

The motion of an armored vehicle is affected by not only the driver of the vehicle, but also by the variations in the terrain over which the vehicle moves. Armored vehicles also have dynamically moving articulated parts including turrets and cannons which may rotate and elevate at a rate greater than the rate which the vehicle itself can change its orientation. The orientation of articulated parts must also be communicated which further increases the entity state data transmission rate for armored vehicles.

Tactical Considerations for High Performance Aircraft Simulators

A manned high performance aircraft is constantly in motion and fully capable of maneuvering, other than the brief time periods spent taxiing before and after the mission. Although an aircraft is in continuous motion while in flight, the motion is predictable for the majority of the time. The flight path of an aircraft is only unpredictable during the time interval in which a pilot repositions the flight controls to perform a maneuver. Once the aircraft enters a steady state maneuver the flight path again becomes predictable. For example, a pilot performs three flight control movements to initiate a steady state high G turn; a roll input to start the turn entry, a pitch input to generate the G, and a roll input to maintain a steady turn attitude while holding the pitch input to maintain G. During an ideal 9.0 G, 360° horizontal turn, an F-16 aircraft is in predictable, steady state flight for all but a few seconds: approximately 2 seconds to enter the turn and 1 second to return to wings level flight after the turn is complete. (It takes slightly more time to enter the turn as the G onset rate is limited by the flight control computer.)

A typical fighter aircraft mission profile includes 2 to 4 hours of transit time to and from the target area with perhaps 4 minutes of air combat maneuvering (ACM). ACM is the flight regime which was once assumed to be impossible to support using the DIS networking protocol. ACM is actually a series of predictable, steady state flight maneuvers such as the high G turn described in the previous paragraph. The aircraft is in unpredictable flight only for a very small percentage of the time. (In this context the term "predictable" is used to describe the flight path of the simulated high performance aircraft for a long period of time in the future for simulation purposes, perhaps 1 - 2 seconds.) The only unpredictable force which can affect the flight path of the aircraft is the pilot and he/she generally points the aircraft and then lets it fly steady state over the small periods of time we are concerned with in simulation.

Protocol Considerations Summary

Other than speed, which is not the dominant factor in protocol design, armored vehicles and high performance aircraft have similar performance characteristics including similar capabilities to change vehicle orientation and direction of travel. High performance aircraft remain predictable for the majority of the time due to the steady state nature of aircraft flight. Vehicle motion for armored vehicles is not quite as predictable as for high performance aircraft as variations in the terrain over which the vehicle travels affect the orientation of the vehicle and armored vehicles typically have tactically relevant articulated parts which must be accurately represented to the rest of the world.

It appears "predictability" is the key to development of a robust simulator networking protocol. Both armored vehicles and high performance aircraft behave predictably for the majority of the time. The position and orientation of "predictable" vehicles can be determined well into the future, several seconds at least, with sufficient accuracy for conduct of fully effective training. Network bandwidth can be efficiently utilized by transmitting entity state data only when there is a change in the state of a vehicle. All other vehicles on the network assume the transmitting vehicle will remain in steady state motion until they receive the next entity state update from that vehicle. Use of dead reckoning to locally calculate the position and orientation of remote vehicles at the local host frame rate should provide a substantial reduction in the amount of entity state data transmitted over the network while providing high fidelity remote vehicle position and orientation accuracy. Design of a dead reckoning algorithm which provides sufficient vehicle position and orientation accuracy for any training simulation network is possible.

As will be seen from the results of the experiments detailed in the following sections of this paper, the DIS networking protocol is a very robust networking protocol which is capable of providing sufficiently accurate vehicle position and orientation data at a sufficient rate over local and long haul networks to support high performance aircraft simulation.

PROTOCOL DATA UNIT TRANSMISSION FREQUENCY EXPERIMENT

The experiment was based on flight dynamics data from a General Dynamics F-16 flight simulator. Time-averaged Entity State PDU transmission rates were measured over a range of thresholds for three different dead reckoning algorithms. These PDU transmission rates document the degree to which dead reckoning can reduce network bandwidth requirements even when relative tight thresholds are employed.

In the late 1970's General Dynamics demonstrated the high performance maneuvering capabilities of the F-16A Falcon fighter aircraft at the Farnborough and Paris Airshows. This airshow flight routine consisted of a series of maneuvers very similar to those performed during classic maneuvering air combat engagements. Some of these maneuvers, in particular the high-G turns, approached the maneuvering limits for a piloted aircraft. Even now, the F-16 is still one of the highest performance fighter aircraft currently flying; therefore, an F-16 flight dynamics simulation should provide the "worst case" for Entity State PDU transmission frequency for a given order dead reckoning algorithm within the high performance aircraft class of networked simulators.

Maneuvering data for this experiment was collected by General Dynamics during a five minute duration F-16 simulator session flown by a fighter pilot with over 1,500 hours of experience in the F-4 and F-16 fighter aircraft. The GD F-16 flight simulator consisted of two Harris 1000™ and three Harris 800™ host processors, and a FPS 5000™ aero model. The simulator visual system was an Evans and Sutherland CT-6™ in a 24 foot dome. The frame rate of the F-16 simulator and the data collected is 50 Hertz.

Figure 1 depicts the actual flight routine performed for this experiment. The pilot conducted a maximum performance take-off transitioning immediately into a slow speed loop. After performing the slow speed loop the pilot performed a series of level rolls then pulled up into a vertical climb followed by a vertical reversal and acceleration in level flight into a high performance level turn. The pilot then reversed heading twice out of the high performance turn and accelerated down the runway in the opposite direction from takeoff. At the departure end of the runway the pilot performed a pull-up and a series of vertical rolls to slow the aircraft before landing.

The flight path data was analyzed by a program which, for a given dead reckoning algorithm, counts how many Entity State PDUs would be transmitted over the duration of the airshow flight routine. For each frame of data, the current position and orientation of the aircraft were compared to the dead reckoning model as applied to the last Entity State PDU transmitted. If the position or orientation exceeded the selected thresholds, a new Entity State PDU was counted as being transmitted. For each dead reckoning algorithm, the position thresholds were varied from 3 meters to 3 centimeters and the orientation thresholds were varied from 10 degrees to 0.1 degrees.

Packet rates were calculated for three different dead reckoning algorithms. These dead reckoning algorithms incorporated either first or second order dead reckoning of position and either zeroth or first order dead reckoning of orientation. "Zeroth order" dead reckoning of orientations simply means orientation was maintained constant over the frame interval. First order dead reckoning of orientation consists of integrating angular velocity over the frame interval to update orientation. Similarly, first order dead reckoning of position consists of simply integrating linear velocity over the frame interval. Second order dead

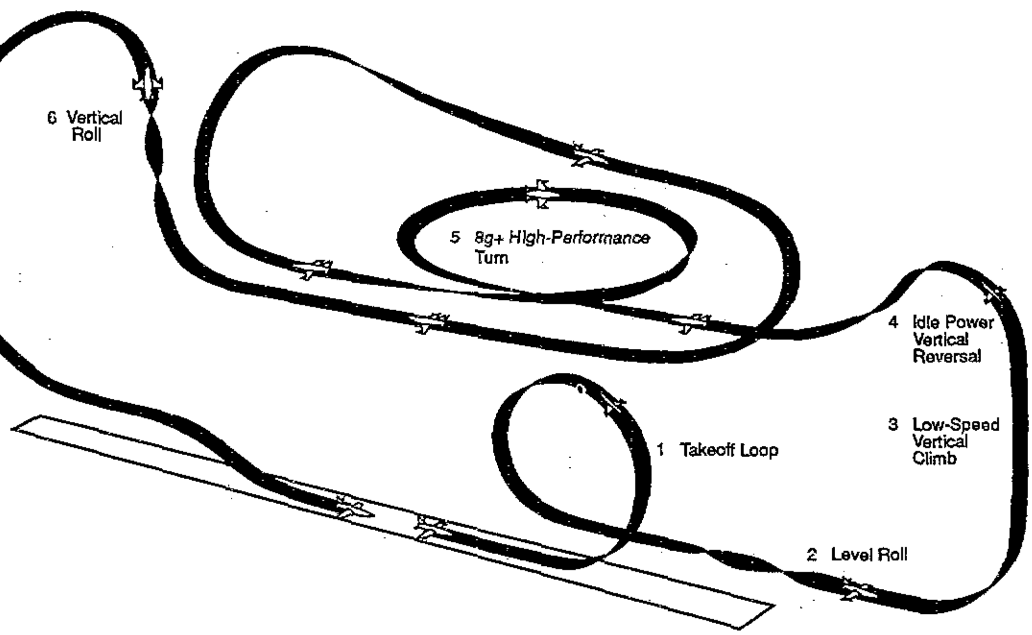

Figure 1: F-16 Paris Airshow Flight Routine

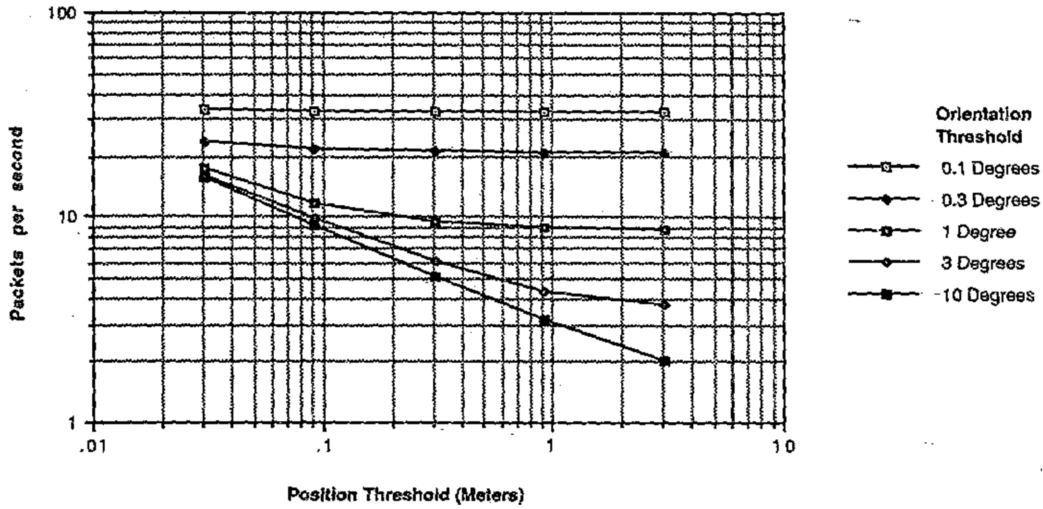

Figure 2: Algorithm 1: First Order Position, Zero Order Orientation

reckoning of position consists of integrating acceleration over the frame interval to update velocity and then integrating velocity to yield position. The average Entity State PDU transmission rate over the course of the flight routine is plotted for the range of thresholds for differing order dead reckoning algorithms in Figures 2 through 4.

Experiment Results

Figure 2 is a plot of the Entity State PDU transmission frequency versus vehicle position thresholds using a first order position, zero order orientation dead reckoning algorithm for varying vehicle position and orientation thresholds. This dead reckoning algorithm is the same one used in SIMNET. The position thresholds were equal for the vehicle x, y, and z axes, and the orientation thresholds were equal for roll, pitch, and yaw. The PDU transmission frequency shown is the average encountered for the complete five minute F-16 Paris Airshow flight profile.

In Figure 2 it can be seen that an average Entity State PDU transmission frequency of approximately 4 packets per second is achieved using the original SIMNET armored vehicle dead reckoning algorithm with a vehicle position accuracy of 1 meter and an orientation accuracy of 3 degrees. It is important to note that the dead reckoned position of the aircraft remains within an 8 cubic meter volume, with the actual aircraft position being the centroid of the volume, for the entire flight profile with only a 4 packet per second PDU transmission frequency. Increasing vehicle position accuracy by an order of magnitude, from 1 meter to .1 meter, only doubles the Entity State PDU transmission frequency for a 3 degree orientation threshold.

The highest accuracy data derived, .03 meter position and 1 degree orientation, equates to an Entity State PDU transmission frequency of approximately 17 packets per second. This accuracy is likely sufficient for even engineering simulation and far exceeds the accuracy required for simulation training exercises. The use of a first order dead reckoning algorithm identical to the SIMNET armored vehicle dead reckoning algorithm for this F-16 flight profile provides a 66% (approximately) reduction in network traffic from 50 Hertz to 17 Hertz even using extremely very tight thresholds of .03 meter for position and 1 degree for orientation.

Figure 3 is a plot of the Entity State PDU transmission frequency versus vehicle position thresholds using a first order position, first order orientation dead reckoning algorithm. The most striking feature of the plot is the significant reduction in PDU rate for tight angular thresholds. Adding first order extrapolation of orientation is so successful that violations of the position threshold are the primary source of PDUs, except for position thresholds looser than about a meter. Thus, the next natural step to reduce PDU rates is to improve the predictive power of the algorithm with respect to position.

Figure 4 is a plot using a second order position, first-order orientation dead reckoning algorithm. Note that the scale on the vertical axis has been reduced by a decade. Packet rates due to positional threshold violations have been dramatically reduced. For example, the curve connecting points of 0.1 degree orientation threshold is now nearly flat. While the angular-threshold-dominated right hand end of this curve has changed very little, the packet rate at the position-threshold-dominated left end of the curve has dropped by over a factor of three. For a 1 meter position and 3 degree orientation threshold the packet rate is only 1.3 packets per second. This is comparable to the packet rates generated by ground vehicles using the first order position, zero order orientation dead reckoning algorithm of Figure 2.

A more comprehensive comparison of simplest and most complex dead reckoning algorithms studied is presented in Figure 5. For each value of the thresholds, Figure 5 shows the ratio of the PDU transmission frequencies using the simple first order position, zero order

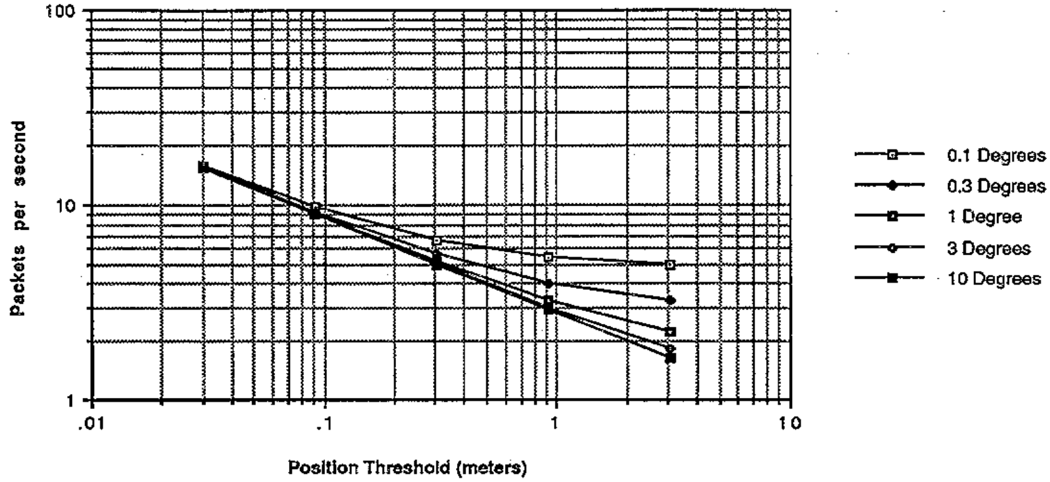

Figure 3: Algorithm 2: First Order Position, First Order Orientation

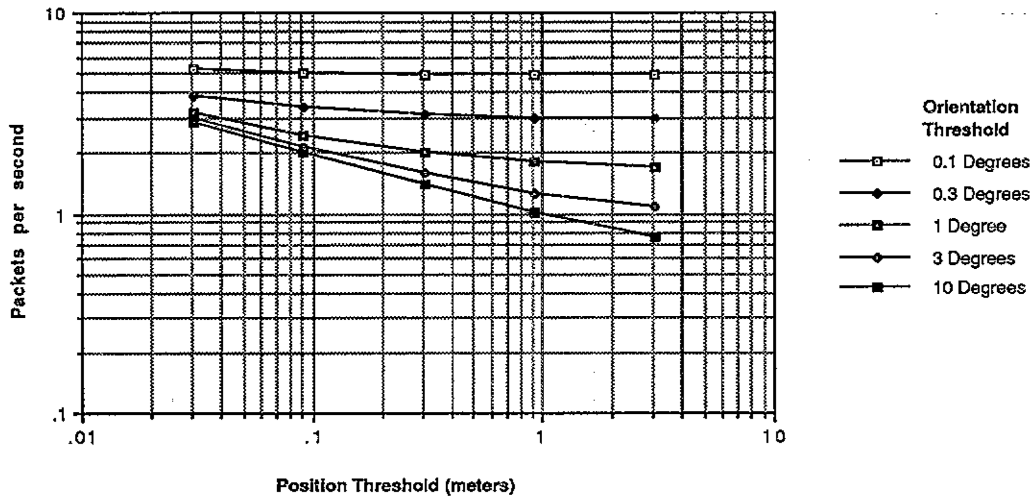

Figure 4: Algorithm 3: Second Order Position, First Order Orientation

orientation dead reckoning algorithm of Figure 2 to that obtained with the more complex second order position, first order orientation dead reckoning algorithm of Figure 4. The improvements in PDU transmission frequencies range from 300% to 700%. Note the pleasing result that the greatest improvements are achieved among the tightest thresholds.

Experiment Conclusions

Even the simplest possible dead reckoning algorithm can dramatically reduce the network bandwidth required to support a distributed simulation. Additionally, even high performance aircraft simulators can achieve high position and orientation accuracies using only the slightly more complex first order position, first order orientation dead reckoning algorithm.

The use of a second order dead reckoning algorithm considerably further decreases the required network bandwidth, particularly for tight position thresholds. By choosing the dead reckoning algorithm to suit available computational and network bandwidth resources, high performance aircraft simulators with high fidelity position and orientation requirements may be incorporated in distributed simulations.

SUMMARY

Entity State PDU transmission frequencies encountered in this experiment are consistent with the results of a similar high performance aircraft simulation experiment using first and second order dead reckoning algorithms conducted by Kenneth D. Morris and Suresh Goel of Northrop Corporation. The results of the Northrop

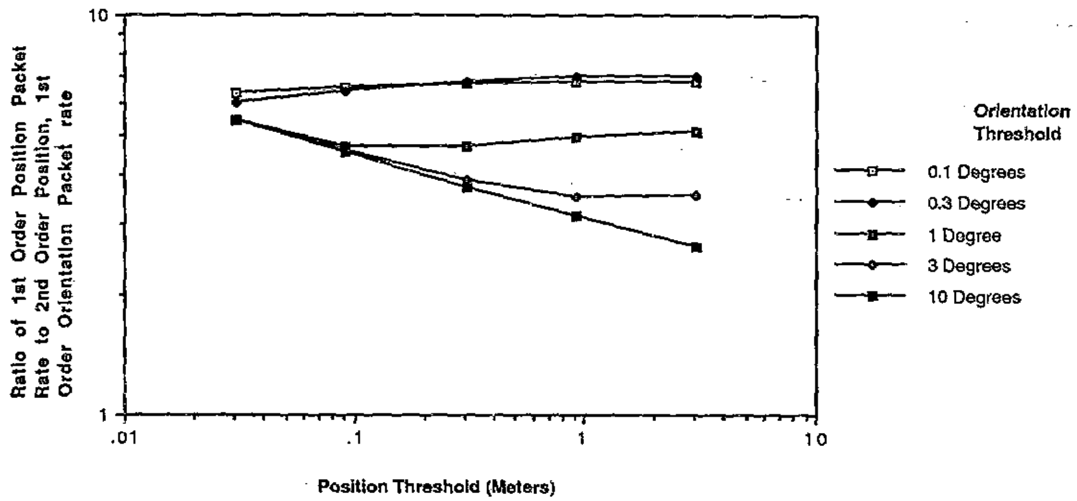

Figure 5: Comparison of Algorithms 1 and 3

PDU transmission frequency experiment and the BBN PDU transmission frequency experiment are consistent. DIS is fully capable of supporting high performance, high fidelity aircraft simulation. Although DIS may have been perceived by some in the simulation industry to lack the capability to provide sufficient fidelity for high performance aircraft simulation, the results of the two independent PDU transmission frequency experiments conducted to date show this is not the case.

The DIS networking standard is a robust, relatively simple, solution for networking hundreds to thousands of simulated entities in a common tactical environment. DIS provides a high validity tactical war fighting training medium which can cost effectively exercise all levels of command and control for U.S. naval, ground and air forces. Conduct of very large scale, joint operations simulation exercises will finally become possible in the 1990's as a result of the implementation of the DIS simulator networking standard.

REFERENCES

1. Schaffer, R., and R. Waters. "Dead Reckoning Algorithms and the Simulation of High Performance Aircraft", BBN White Paper ASD 91-015, UCF IST 4th Workshop on Standards for the Interoperability of Defense Simulators; March 1991

2. Miller, D., A. Pope and R. Waters. "Long-Haul Networking of Simulators," Proceedings: Tenth Interservice/Industry Training Systems Conference; Orlando, Florida; December 1989

3. Pope, A. and R. Schaffer, The SIMNET Network and Protocols; BBN Report 7627; Bolt Beranek and Newman Inc.; Cambridge, Mass.; June 1991

4. Military Standard (Draft), Entity Information and Entity Interaction in a Distributed Interactive Simulation, IST-PD-90-2 (Revised), University of Central Florida, Institute for Simulation and Training; January 25, 1991

5. Morris, K. and S. Goel. "Interim Report for Long Haul Data Transmission System"; Northrop Report NOR 91-28; Northrop Corporation; February 1991

ACKNOWLEDGEMENTS

The authors would like to thank General Dynamics, Flight Simulation Laboratory, Ft. Worth, TX for providing the F-16 Paris Airshow flight simulator data. We would especially like to thank Steve Norman, Doug Gould, Mike Lientz, Matt Landry, David Tedeschi, Ajay Patel, Becky Waren and Dwaine Robinson. The PDU transmission frequency experiment described in this paper would not have been possible without their help.

The initial studies reported in this paper were funded by the Air Force Human Resources Laboratory, Williams AFB. We would like to thank Major Michael Kamrowski, Research Engineer for the Aircrew Combat Mission Enhancement (ACME) - Multiship Network Development for his guidance and support.

The authors would like to thank Dr. Jerry Burchfiel and Rolland Waters of BBN for their helpful advice and assistance in conducting the studies reported in this paper.

ABOUT THE AUTHORS

Richard L. Schaffer is a staff scientist with BBN Systems and Technologies' Advanced Simulation Business Unit. He holds a B.S. in Physics from Massachusetts Institute of Technology. For the past five years, he has worked on the SIMNET Large Scale Simulation project. Mr. Schaffer has also been actively involved in the Distributed Interactive Simulation standards process since its inception. He is a member of two DIS working groups, and chairs a sub-group on articulated parts.

Edward P. Harvey is a military aviation consultant with BMH Associates, Inc. He holds a Chemical Engineering degree from Georgia Institute of Technology. Mr. Harvey is a former F-14A Tomcat Radar Intercept Officer (RIO) and air combat maneuvering instructor with

1,400 hours of tactical fighter experience. He has been involved in a variety of flight simulation projects including the U.S. Navy F-14D Aircrew Training System, the DARPA/U.S. Army SIMNET program, and the DARPA/ U.S. Navy WISSARD program. He is a member of the Distributed Interactive Simulation Time Critical working group.