In the Proceedings of the 1994AI, Simulation and Planning in High Autonomy Systems Conference, held at the University of Florida, Gainesville, 7-9 December 1994.

Insertion of an Articulated Human into a Networked Virtual Environment

(paper AIS-26)

David R. Pratt[1], Paul T. Barham, John Locke, Michael J. Zyda
Naval Postgraduate School

Bryant Eastman, Timothy Moore, Klaus Biggers, Robert Douglass, Stephen Jacobsen
SARCOS, Inc.

Michael Hollick, John Granieri, Hyeongseok Ko, Norman I. Badler
University Of Pennsylvania

Most Distributed Interactive Simulation (DIS) technology demonstrated over the past several years has focused on the interaction of vehicles. The dismounted infantryman--the individual foot soldier--has been largely ignored or represented by static models. After six weeks of development, The Naval Postgraduate School, SARCOS, Inc., and, University of Pennsylvania, under Army Research Laboratory sponsorship, demonstrated the insertion of a fully articulated human figure into a DIS environment. This paper outlines the network and software architecture of the system.

1.0 Introduction

The Simulation Networking (SIMNET) Project [4][5][3] has shown the results of connecting low-cost, networked, man-in-the-loop simulations by a common protocol. SIMNET focused on simulating an armored battlefield. When it was designed, the emphasis was practical for several reasons, both functional and technical.[2] The U.S. Army, the prime customer for the system, was prepared to fight a Soviet land force on the plains of Germany. Most experts agreed that the conflict, if it erupted into warfare, would have been a large-scale tank battle pitting the armored units of either side. In constructing SIMNET, this paradigm allowed for many simplifying decisions. For instance, a tank crew sees the outside world though small view ports. The largest of these four ports has an eighty-nine degree horizontal Field of View (FoV) and a considerably narrower vertical FoV [6]. Compared to the 180-degree or more horizontal FoV of a human in open space, the attenuated view limits the computational load involved in processing the visual channels. The simulation of the environment was likewise restricted since crew members remained inside the tank. Only the rough experience of being in a tank, and not the full detail of the surrounding environment, needed to be simulated for realism. The terrain database could be constructed so that only features influencing tank warfare were considered.

---

1. Contact author at Department of Computer Science, Code CS/Pr, Naval Postgraduate School, Monterey CA 93943. pratt@cs.nps.navy.mil (408) 656-2865, fax (408) 656-2814

2. BBN did a superb job with the existing technology. We in no way minimize their accomplishments.

Distributed Interactive Simulation (DIS) [2], the successor to the SIMNET protocols, likewise focuses on vehicles.

As the concept of the New World Order evolved, emphasis shifted from large-scale tank battles to small regional conflicts which rely more on individual soldiers.[3] The Dismounted Infantryman (DI) plays several roles in this type of conflict, not all of which are currently cost-efficient or feasible to simulate. However, some roles, as in Special Operations Forces (SOF) and Military Operations in Urban Terrain (MOUT), do lend themselves to simulation. These are operations in which small units of soldiers act in close coordination to accomplish the mission. The actions and team work used by these types of soldiers resemble the actions of urban civil police authorities.

Following a brief discussion of the history of DIs in distributed simulation, we discuss a minimal icon for representation as a DI. We then outline the software and network architecture of the system built by the authors and demonstrated at the INCOMSS-94 meeting at Fort Benning, Georgia in February 1994. However, a full discussion of Jack, a system for modeling 3-D articulated figures; NPSNET-IV, a 3-D virtual world simulation; and the ISMS VME hardware controller, an articulated figure output device, are beyond the scope of this paper. We conclude with a discussion of the scenarios demonstrated at INCOMSS-94.

2.0 SIMNET / DIS Representation of Dismounted Infantry

The SIMNET protocol, as described in [1], was the first standard used in a distributed virtual battlefield. Since the systems being modeled were primarily armored combat entities, the protocols and displays could be optimized for that purpose. The systems was limited to three basic types: Static (non-moving), Simple (no articulated parts), and Tank (two articulated parts, turret and gun). Human figures in the SIMNET world were represented by one of two methods. In the early systems, a texture map was used to represent the individual soldier or fire team. The different postures (standing, prone, kneeling, running, etc.) were represented by different textures. As a result, when the figures moved they appeared to merely slide along the ground. In some later systems, the texture DIs were replaced by fixed models. To represent moving DIs, animations were created for running and crawling, limited by fixed speed and stride. When an entity varied speed, their feet would skate across the terrain.

The DIS protocols have an extensible method of representing articulations[5]. Each of the Degrees of Freedom (DoF) is represented by a ninety-six-bit record. The record contains enumerations detailing articulation type, the changing parameter, and the value of the change. While this is a flexible method of describing the articulation, for entities with large number of DoFs, it is an expensive use of network bandwidth. Table 1 contains a comparison of the relative lengths of comparable packets between DIS and our optimized method.

3.0 Human Figure

For this project, the human figure model we used was created by the University of Pennsylvania for their Jack Program. The model was converted to MultiGen Flight format to be compatible

3. Of the three most recent major U.S. campaigns (Granada, Panama, and Persian Gulf), only one, the Persian Gulf, involved a large amount of armored vehicles.

Component	DI_guy	DIS 2.0.3	Difference
Header / Body	76	190	114
Articulations	156	624	468
Total	232	814	582

Table 1: Comparison of the length in bytes of PDUs needed to represent a 39DoF Human Figure

with the visual system, NPSNET-IV [7]. This allows the model to be easily loaded in by SGI's Performer API and treated like other entity models.

As in Figure 1, the figure has a total of thirty-nine DoFs in seventeen separate joints. The torso contains a single joint at the waist. The neck has a lower joint at both the base and top connecting to the torso and head respectively. Each arm has three joints; shoulder, elbow, and wrist. Each of the legs contain four joints; hip, knee, ankle, and toe.

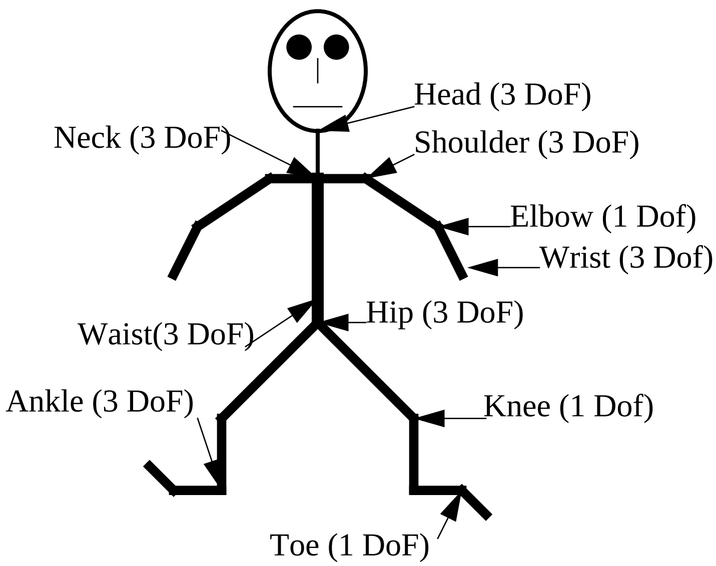

Figure 1. Schematic of Human Figure Showing DoFs

4.0 System Architecture

The basic system architecture balances network loading, computational resources, and system requirements. The overall system was constructed this way due to available equipment and to minimize the risk of failure. The major compromise was the location of Jack and the DI_guy process on separate machines. Since this was the first time we had built a system like this, we were concerned with the computational load of Jack and the DIS conversion process.

Figure 2 shows the actual network connections. All network traffic traveled a single Ethernet segment, reducing the number of physical segments of wire and Ethernet interfaces that would otherwise be required by the number of point-to-point communications in the logical design.

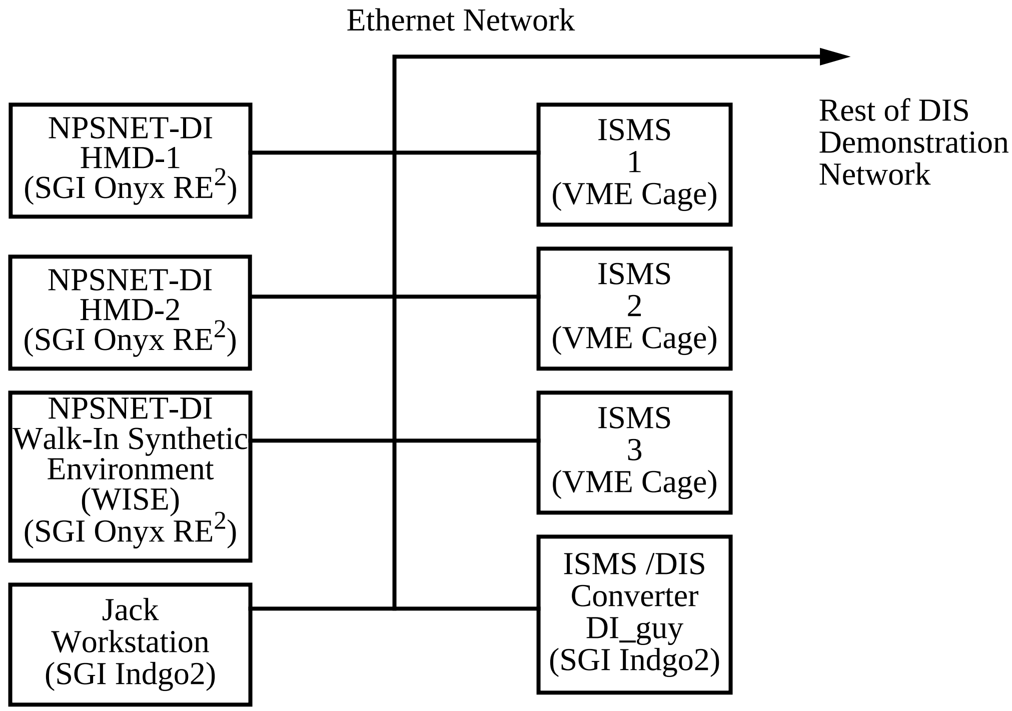

Figure 2. Physical Network Connections

As shown in Figure 3, the system establishes two logical networks, one that uses point-to-point (TCP/IP) communications for protocol formats optimized for the specific implementation, and another for broadcasting (UDP/IP) DIS traffic. While having a single physical network cut down the computational resources, it limits future growth by maximizing potential system bottlenecks. (A detailed discussion of the protocols is presented in Section 6.0.)

5.0 Component Functions

This section contains an overview of the various components of the systems. While each component is a complex system in its own right, we focus on system functionality as it applies to the interface between systems, specifically network message formats. During the description of the system, we refer to the components of the DI_DISPLAY_DATA_MESS_TYPE structure in Figure 4. It represents the complete set of articulations and state data for the human icon.

5.1 ISMS VME Hardware Controller

The Individual Soldier Mobility System (ISMS) controller is a VME-based real-time computer whose primary functions are physical hardware control and monitoring of user input sensors. The user interface consists of three systems, the mobility platform, the sensor suit, and the head-mounted display (HMD).

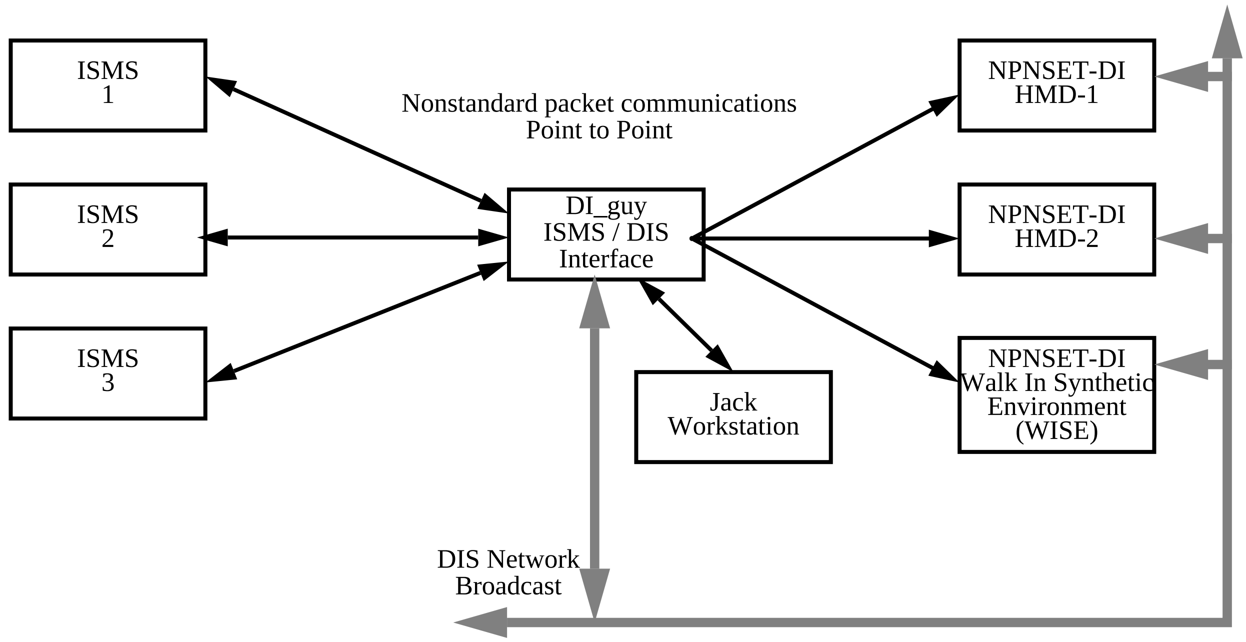

Figure 3. Logical Network Connectivity

The mobility platform resembles an exercise unicycle with a seat and two pedals. The seated user controls the direction by swiveling the top of the seat with his hips. The speed of the corresponding icon in the virtual world is based upon the user's pedal speed. As part of the hardware control, resistance is applied to the pedals based upon pedaling speed and relative terrain slope. Pedaling speed and seat torque are used to compute the new X-Y location of the soldier in the virtual world.

The mechanical sensor suit is worn by the user. It consists of a number of attachments to the user's limbs that output arm position and upper body orientation. The ISMS controller takes the raw sensor input and fills in the data for the left and right arms as contained in the ARM_ANGLES_TYPE structure. Likewise, the waist position is also computed from the sensor suit.

The HMD displays the virtual environment to the user and outputs head position. The HMD sensor data and the sensor suit input are used to compute neck and head position. Eye position, in world coordinates, are computed based on an offset from the icon's world position and the computed offset to the head.

5.2 DI_guy

The ISMS/DIS interface (DI_guy) process is a communications server, elevation server, and a data display device. As a communications server, it dead reckons the human figure icons and formats DIS-compliant Protocol Data Units (PDUs). A copy of the terrain database is maintained to provide the ISMS with ground elevation and slope for a given location. One of the primary uses of the DI_guy is to debug the system by showing current location, status, and parameter values.

```
typedef struct {
   float       wrist[3], elbow[1], shoulder[3];// The DoFs for the Arms
} ARM_ANGLES_TYPE;

typedef struct {
   float      toe[1], ankle[3], knee[1], hip[3];// The joint angles for the legs
joints
} LEG_ANGLES_TYPE;

typedef struct {
   DOF_6      dis_pos;                      // The origin of the entity in world
space
   DOF_6      dis_eyes;                     // Location of the view point
   float      neck[3],                      // Three degrees of freedom
              head[3],                      // Three degrees of freedom
              waist[3];                     // Three degrees of freedom
   LEG_ANGLES_TYPE right_leg, left_leg;     // Angles for the leg joints
   ARM_ANGLES_TYPE right_arm, left_arm;     // Angles for the arm joints
} BODY_ANGLES_TYPE;

typedef struct {
   int        length;                       // The length of the packet.
   int        type;                         // The type, DI_DISPLAY_DATA_MESS
   int        entity;                       // The entity number
   BODY_ANGLES_TYPE body;                   // All the body angles and orientation
   DOF_6      rifle;                        // The rifle location and orientation
   float      velocity[3];                  // Velocity in m/s in world coordinates
   unsigned int status;                     // Flag information
} DI_DISPLAY_DATA_MESS_TYPE;
```

Figure 4. DI_guy Message Format

The ISMS updates the DI_guy process at sixty Hz. Once the packets are received, DI_guy computes elevation of the terrain based upon the X-Y. The normal of the polygon on which the virtual soldier is located is computed and given to the ISMS where it is used to compute the resistive loading on the ISMS pedals. These packets then pass to Jack to compute the remaining joint angles. Once Jack fills in the LEG_ANGLES_TYPE structure for the left and right legs, DI_guy then forwards the packet to the display devices, NPSNET-DI.

5.3 Jack

Locomotion is based on the global velocity vector and global (compass) heading of the soldier. The current time is recorded at the beginning of each footstep, and the time at each update is used to determine the proper frame of the stride to display. A flag in the update packet indicates whether an entity is controlled by an operator in an ISMS, or whether it is from some other source. If the entity is not ISMS based, the figure’s upper body is animated with a naturalistic arm swing.

Locomotion computations are only performed when the figure is in the standing (or upright) posture. The posture can only change when the figure is not walking; thus a figure must stop walking to change posture, and must stand up to start walking. These restrictions make intuitive sense and avoid undesirable system behavior.

Additionally, a mechanism is provided for a forced stop. In normal conditions, the figure comes to a stop in a reasonable manner (by slowing down and taking a final step) when the velocity drops to zero. In the case of a collision with a fixed object, however, this behavior is unacceptable. A flag in the update packet indicates when a sudden stop is required. When set, the figure immediately returns to the default standing posture and the current step is canceled.

Upper body angles of the ISMS operator are measured by the body suit and sent to Jack, which does some simple checks on their validity. The angles are then assigned to the corresponding joints in the human figure.

A special case is the head/neck joint pair. These are not measured by the suit, but are derived from the viewpoint orientation (measured with a head-mounted sensor) and the torso orientation. Since the viewpoint orientation is in the global frame, the head/neck joints are adjusted so the simulated human's head orientation matches that of the viewpoint. This is done by subtracting the torso bend angles from the viewpoint orientation.

A correction is also done on the shoulder and head joints while the entity is prone, or undergoing a posture transition. Since the operator is always upright, not all the measured joints do correspond to the correct simulated posture. For example, if an operator with his arms straight in front (firing a rifle, perhaps) goes prone (indicated by hitting a switch on the rifle), and the raw joint angles are used, his arms will now go *into* the ground since the simulated torso orientation is now roughly parallel to the ground plane. Also, the simulated human is looking into the ground. To correct, the torso orientation is used as a correction factor for the shoulder and neck joints while prone (or in a transition). By using this method, the simulated soldier always has his arms and head in the correct global orientation.

5.4 NPSNET-IV

The three display devices, the two HMDs and the Walk-In Synthetic Environment (WISE) (Figure 5) use the NPSNET-IV [7], a 3D battlefield simulator as the tool for visual display. Since the soldiers can see the other non-ISMS entities in the simulation, they read the DIS network for the status of the other entities. The status of the ISMS humans in the simulation is transmitted over the point-to-point network. Since the ISMS entities are also sent out on the DIS network, they have two paths to arrive at the displays, Figure 3. To avoid showing two icons representing the same DI, the DIS PDUs are filtered out of the DIS PDU stream.

For the Fort Benning demonstration, three variations of NPSNET were used. The first, the WISE, incorporated three large screen projection monitors arranged similar to the CAVE [9]. As shown in Figure 5, the ISMS provided the user an approximately 270 degree FoV of the three screens, each measuring eight feet wide by six feet tall with a resolution of 960x680 pixels. While the low resolution did produce some aliasing artifacts, they were minimal and did not distract from the overall purpose of the system. Arranged around the base of the ISMS were four speakers, driven by a separate process that monitored the network and computed the location and strength of the sound source. Together, the sound, force feedback from the ISMS, and the WISE display produced a convincing environment for the soldier using the ISMS system.

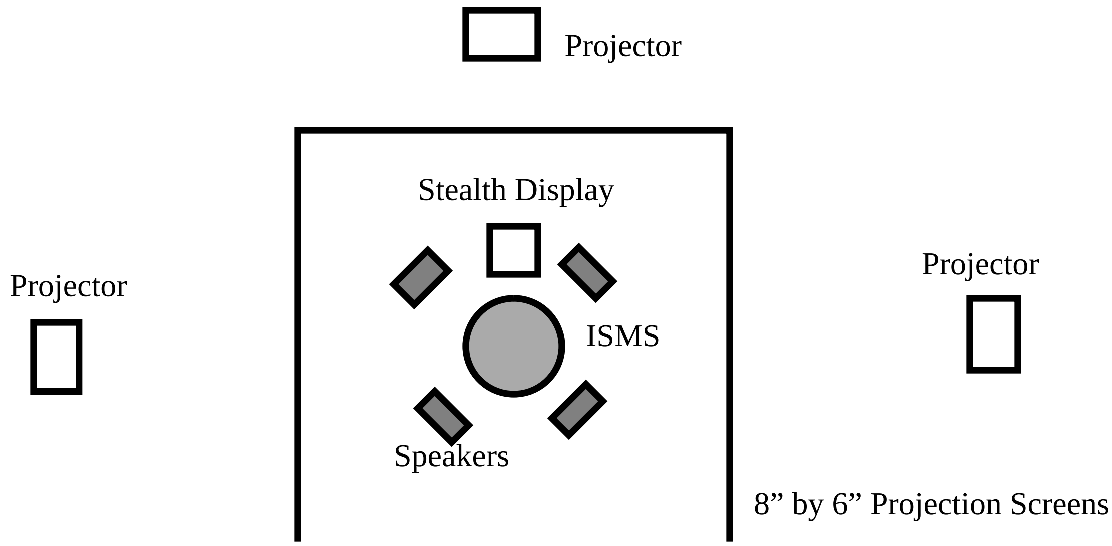

Figure 5. Layout of the WISE Display System

Due to the size of the WISE, only one solider at a time could occupy it. The other two soldiers were wearing Kaiser Electrooptics HMDs equipped with Polhemus head-tracking sensors. The two HMDs, one high and one medium resolution, provided immersive views of the environment to the solders.

The final type of display, not shown in Figure 2, is an observation or “stealth” platform. This system views the simulation without creating message traffic. Shown near the top of Figure 5, the stealth display was placed on the floor in front of the soldier in the WISE. By attaching the view on the soldier, this display mirrored what was being sent on the network, allowing the soldier to verify his upper body positions during hand and arm signals.

6.0 Network Implementation

As shown in Figure 3, there are two logical networks, a point-to-point local network using proprietary protocols and a broadcast network using the DIS protocols. The visual systems, therefore, have two paths to receive the updates for each of the DI entities, one path via the point-to-point connection, the fully articulated model, and the second path from the DIS PDUs put out by DI_guy. To avoid the duplication of icons, a filtering system was set up to discard DIS PDUs coming from DI_guy.

All ISMS subsystem internal connections are socket-based using point-to-point TCP/IP. Ports are assigned dynamically by DI_guy based upon a central request port, eliminating the need to hard-code specific port addresses in the program. The DIS communication was broadcast using UDP/IP on a well-known port. However, the use of point-to-point communications required that the same messages be sent to each visual system individually.

The basic message structure is shown in Figure 4. It is sent from the DI_guy process to update the graphics process and represents the longest and most complete message. As discussed in Section 5.0, for ISMS systems, Jack fills in the lower body angles and checks the limits on the upper body angles. The ISMS controller fills in the remaining information. On non-ISMS systems, Jack fills in all of the joint information and the location is determined from algorithmic computations.

For efficiency's sake, we rejected the DIS round-world coordinate system for our 16X16 kilometer virtual area in favor of a flat world system corresponding to the SIMNET standard. Since the original terrain database was based upon a flat world and over half the visualization systems (SIMNET M-1s and M-2/3s) use flat world coordinates, it made little sense to do an expensive coordinate conversion only to "unconvert" later.

7.0 Demonstrated Scenario

At INCOMSS-94, we demonstrated a multi-soldier system using three scenarios. The first scenario had two of the soldiers "dismount" from a ModSAF controlled M-2 Bradley, run to a building and make sure it was empty. The third ISMS represented an enemy soldier. Since the weapons effects were not implemented, when the friendly soldiers entered the building, the enemy solider ran out the back. The friendly soldiers then ran back to the waiting M-2s.

The second scenario staged at Fort Benning (in both the real and virtual worlds) was similar to the first with one exception. The virtual building being assaulted was the actual building with the demonstration audience. At the end of the scenario, one of the ISMS operators threw a grenade though the door.

The final scenario had one of the ISMS operators give a series of arm and hand signals. These three scenarios were the first time that an articulated icon under human control was shown in an DIS environment.

For the demonstration, the exercise network was divided into two segments based upon the protocols, DIS and SIMNET. Connecting the two was the LORAL PDU translator. The function of this device was to convert DIS PDUs into the corresponding SIMNET PDUs and vice versa. On the DIS side, no noticeable delay between one of the soldiers on the ISMS moving and the corresponding action being shown on the DIS visual displays existed. However, there was a consistent seven second delay representing events on the SIMNET side.

In order to ensure consistent body orientation and posture, both ISMS and Jack updated the displays faster then the frame rate to account for the asynchronous nature of the SGI graphics pipeline and to achieve the minimal delay possible between action and display. The ISMS would send data out as fast a possible--30-60Hz--to the DI_guy process, overwriting any pending messages. The same was done from DI_Guy to NPSNET. While this had the effect of placing more packets then required in the network, it did account for the different cycle times between processes and reduced the apparent latency

8.0 Future Work

Insertion of an articulated human figure into the virtual world is a task that is just beginning. In this section, we briefly discuss some of the continuing and potential projects to insert humans into the virtual world as well as some potential uses of this technology. They are nowhere near a complete list of the applications, but have been chosen as representative capabilities. Not presented here are technical issues like increased articulations, faster processing, and better graphics that we are continually working on.

8.1 Ship Walkthrough

Ships represent one of the worse possible situations for a walkthrough. Not only do they possess the architectural complexities of a building, they also have to be smaller, completely self-contained, more intricate spaces. We envision two fundamental applications to ships. The first is basic human factors design. It is difficult to get a sense of the problems in moving about a ship that is swaying and heaving while trying to read instruments. The ISMS with a HMD can immerse the user and have them maneuver through the environment with the force on the pedals changing to reflect ship motion. From such studies, we could determine the configuration of spaces and equipment. The second use would be that of familiarization. Ships have an extremely large number of cables, valves, compartments (rooms), and piping. It is the job of the Damage Control Assistant (DCA) to know where all are located. To familiarize himself with the layout, the DCA spends a considerable amount of time tracing piping and wiring from compartment to compartment. By digitizing the ship, a virtual ship model can be created. By turning systems off, such as the bulkheads (walls) and highlighting others, such as the fire fighting systems, the DCA can move among the environment and get a better grasp of the ship’s layout.

8.2 Medical Corpsman

In our demonstrations we have shown that is it possible to populate the world with icons that can move under the control of and mimic the actions of humans. One of the side effects of introducing this capability into the synthetic battlefields is that the icons will be shot and require medical care. To address this need, ARPA has started a program to train medics, the military version of a Emergency Medical Technician or Paramedic, in the DIS environment. The basic capabilities of the medic are location of the wounded solider, wound identification and treatment, and patient prioritizing (triage). Since very few real humans volunteer to be shot for medic training, this provides a way for the medic to train in situations they might encounter on a battlefield-like situation.

8.3 Police Training

Unfortunately, many current day police officers require the same urban combat skills as the military. Increasingly, skills like hostage rescue, room clearing, enemy identification, situation response, and team training are becoming a more common part of a policeman’s training. With the insertion of the human into the DIS environment, these types of skills can now be practiced in simulation.

9.0 Conclusions

Fully articulated human figures can be incorporated in DIS, but work is required on articulation parameters. As shown in Table 1,582 bytes per message were saved by not using the DIS messages internally. By using a well-defined structure rather than a generic one, it is possible to reduce the number of bytes needed to describe the articulations by a factor of four. The differences in the length of the header can be attributed to a fundamental difference in the implementation. For instance, we assume that soldiers will not change sides fifteen times a second, an eventuality that is accounted for within the DIS protocols.

Due to the number of articulations and the complexity of human motion, systems can be expected to send out packets at the frame rate. Assuming a frame rate of fifteen Hz, and considering only packet size, each soldier-system produces the same network load as five to eight tanks or three high performance aircraft. This could potentially cripple a large scenario.

The computational load of Jack and the DIS conversation process did not prove to be excessive. Michael Hollick and John Granieri of UPENN have developed a table driven version of Jack that is more suited for the low resolution display of human figures. Bryant Eastman and Tim Moore of SARCOS have placed all of the DI_guy functionality into a heavily modified version of NPSNET and will be porting the table drive version of Jack onto the VME real-time system in the ISMS. These two enhancements represent a significant reduction of the number of machines and packets that are required for the articulated human icon. As pointed out in Section 4.0, the elimination of DI_guy removes a system bottleneck.

The next major demonstration is scheduled for AUSA in San Jose at the end of May. This demonstration will include weapon effects and a fully compliant DIS system.

10.0 Acknowledgments

This whole project would not have been possible without the Human Research and Engineering Directorate, Army Research Laboratory (ARL). Not only were they the project managers, but they came up with the idea and provided technical guidance for the ISMS. It was ARL’s foresight and insight into the required functionality of the system that formed the NPS / UPENN / SARCOS team. Truly, we could not have done it with out them. The authors would also like to thank Farid Mamagahni and Jim Madden for organizing the Fort Benning demonstration. The terrain database was developed at the Topographic Engineering Center (TEC), Fort Belvior under the guidance of George Lukes and Jay Banchero. The HMDs were provided by Kaiser Electrooptics with assistance provided by Frank Hepburn.

11.0 References

[1] Pope, Arthur, “SIMNET Network Protocols,” BBN Systems and Technologies, Report No. 7102,Cambridge, MA, July 1989.

[2] Institute for Simulation and Training, “Protocol Data Units for Entity Information and Entity Interaction in a Distributed Interactive Simulation (DRAFT),” Version 2.0.3, IST-PD-90-2,Orlando, FL, May 1993

[3] Thorpe, Jack A., “The New Technology of Large Scale Simulator Networking: Implications for Mastering the Art of Warfighting,” Proceedings of the 9th Interservice/Industry Training System Conference, Nov. - Dec 1987.

[4] Garvey, Richard E. and Monday, Paul, “SIMNET (SIMulator NETwork),” BBN Technical Note, Ft. Knox, KY., July 1988.

[5] Institute for Defense Analysis, “SIMNET,” Draft, Arlington, VA., May 1990.

[6] Perceptronics, “SIMNET M1 Crew Manual,” SIMNET Manual No. PTUM 001-1250-89-10 (rev-2), Woodland Hills, CA.

[7] Macedonia, Michael R. Zyda, Michael J., Pratt, David R., Barham, Paul T. and Zeswitz, Steven, “NPSNET: A Network Software Architecture for Large Scale Virtual Environments,” Presence, Vol. 3, No. 4. Fall 1994.

[8] Badler, Norman I., Phillips, Cary B., Webber, Bonnie Lynn, “Simulating Humans,” Oxford University Press, New York, 1993.

[9] Cruz-Neira, C., Sandin, D.J., DeFanti, T.A., “Surround Screen Projection Based Virtual Reality: The design and Implementation of the CAVE,” *Computer Graphics*, August 1993.