MODELING INDIVIDUAL HUMANS FOR COMPUTER GENERATED FORCES

Douglas Reece
Institute for Simulation and Training
University of Central Florida
Orlando, Florida

ABSTRACT

IST has been building computer generated humans—combatants and civilians—to populate a virtual battlefield as part of the Team Target Engagement Simulator (TTES) project. This project, which is sponsored by the Naval Air Warfare Center Training Systems Division in Orlando, will train small infantry units to fight in urban terrain. Such a low level simulation with direct human involvement requires detailed models not only of terrain and human behavior but also of human physical characteristics. This paper presents an overview of the problems that a designer of computer controlled humans must address to create realistic entities. The problems span all levels from low level modeling to cognitive behavior. At the simulator infrastructure level we discuss DIS representation and urban terrain databases. At the physical environment level models of visible line of sight, sound generation and propagation, weapons effects, and movement are important. The next level addresses physical entity characteristics and requires modes of vision, hearing, movement, wounds, and fatigue. The last level is cognitive, and comprises two parts: automatic behavior such as perceptual processing, feedback-based motion control, and weapon aiming; and deliberate problem solving and action selection. The paper briefly describes our approach to building all of these models.

ABOUT THE AUTHOR

Douglas A. Reece is a Computer Scientist at the Institute for Simulation and Training. He is the Principal Investigator of the TTES Computer Controlled Hostiles project. His research interests are in artificial intelligence, specifically intelligent agent design and computer vision. He has a Ph.D. in Computer Science from Carnegie Mellon University and B.S. and M.S. degrees in Electrical Engineering from Case Western Reserve University.

MODELING INDIVIDUAL HUMANS FOR COMPUTER GENERATED FORCES

Douglas Reece
Institute for Simulation and Training
University of Central Florida
Orlando, Florida

INTRODUCTION

IST is developing artificial, autonomous individual combatants (ICs) to populate a virtual battlefield as part of the Team Target Engagement Simulator (TTES) project. This project, which is sponsored by the Naval Air Warfare Center Training Systems Division in Orlando, will develop a system to train small infantry units to fight in urban terrain. The users will initially be Marine Corps squads, but the system could potentially be expanded to other services, Special Forces operations, hostage rescue missions, etc. The system is designed to operate as a distributed virtual environment simulation using Distributed Interactive Simulation (DIS) protocols to link simulator nodes.

Developing an IC training simulator is challenging not only in terms of the human trainee interface but also for computer generated forces (CGF). All of the visual and audio detail of the environment and of entities must be encoded for CGF in addition to being rendered for human trainees. The detection capabilities of the humans in the trainer should be duplicated in the CGF ICs. All of the complex movements and actions provided in the interface for the human trainees—posture changes, movement, throwing actions, weapon aiming, arm gestures, spoken commands, etc.—should be allowed the CGF ICs as well. Any of these actions displayed for the humans must be encoded for the CGF ICs. In summary, the overall training system must provide appropriate, detailed representations of the environment to both human and CGF entities, and CGF ICs must be modeled in detail in terms of both physical capabilities and behavior.

This paper discusses CGF IC design issues and presents our approach to these problems. The issues span all levels from low level modeling to cognitive behavior (see Figure 1). At the simulator infrastructure level are DIS representation and urban terrain representation issues. The physical level has two parts: the physical *environment* part, where models of visible line of sight, sound generation and propagation, weapons effects, and movement are important; and the physical *entity* part which requires models of vision, hearing, movement, wounds, and fatigue. The highest level is cognitive, and comprises automatic behavior such as perceptual processing, feedback-based motion control, and weapon aiming, deliberate problem solving and action selection.

SIMULATION SUPPORT

Urban Terrain Representation

The urban battlefield domain of TTES is an increasingly important domain in military operations. Simulating such battlefields requires using detailed databases describing buildings. The database used for TTES is in a raw polygonal format commonly used with image generators; there is no organization or semantic information suitable for CGF. The raw polygonal representation lacks semantic information that one would intuitively expect to be useful for operating in a building. People commonly see buildings as collections of rooms and hallways joined together by doorways; windows and doors connect the inside with the outside. Buildings have levels, and stairs (and elevators, etc.) connect them. Buildings can also be viewed as spaces partitioned by structural walls. The raw polygonal database contains none of this information. Of particular note is the lack of aperture information; in the raw data, an aperture is the *lack* of polygons.

We have been developing software tools to automatically process polygonal building descriptions to produce new representations. The new representations provide an efficient organization of polygons for visibility and height calculations, simplified obstacle models for other geometric calculations, and semantic information for reasoning tasks. Such automatic processing

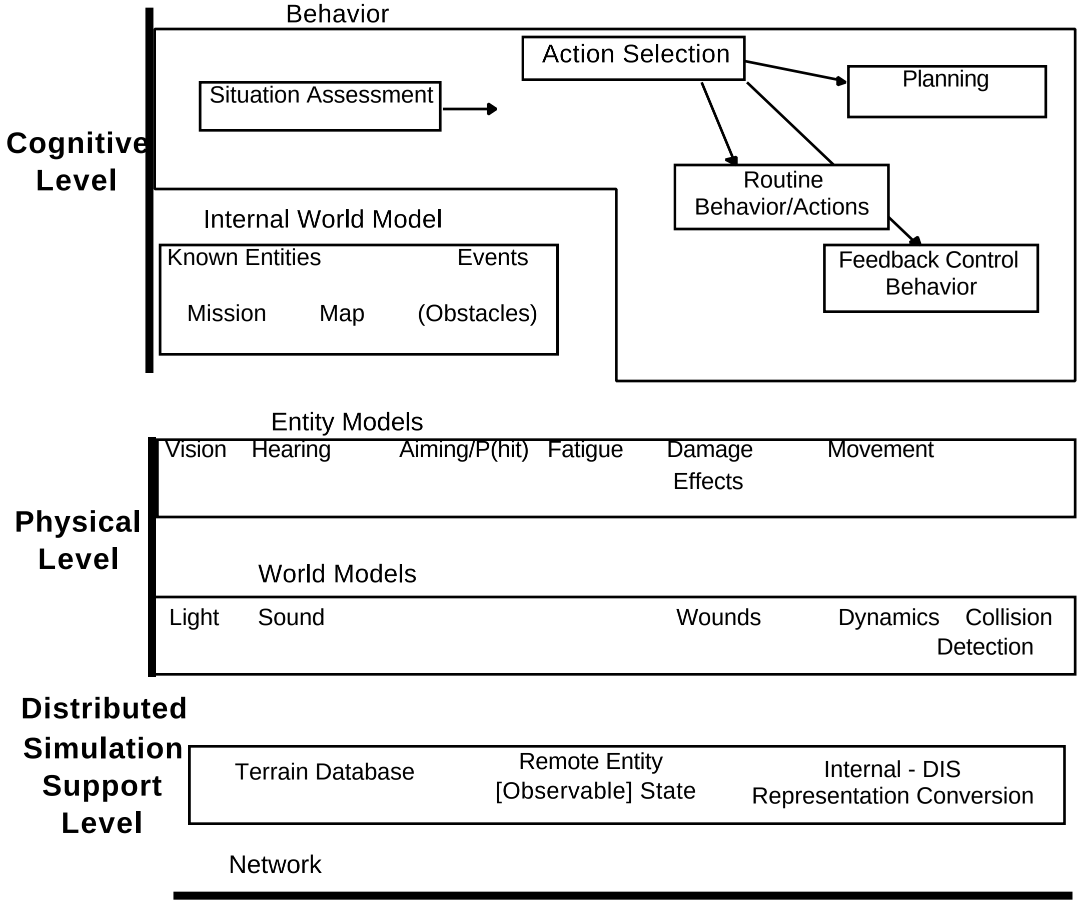

Figure 1. Components of a computer controlled human in different architectural levels

tools will be valuable for rapidly generating urban databases for simulation exercises involving CGF. Further details of our approach can be found in (Reece and Tu, 1996).

Representing Individuals in DIS

While the existing DIS Standard actually has a mechanism for representing a single person, this mechanism is crude and is really only an afterthought to the primary goal of representing tanks and other vehicles. Distributed simulation presents problems representing what individual humans are carrying, what they look like, what they are doing, and what position their body is in (Franceschini 1994, Reece 1994). Of these problems, the body position problem is perhaps the most difficult. The difficulty is that the overall system must generate, transmit, and recreate detailed representations of individuals on both trainee platforms and IC CGF systems (see Figure 2). The two basic approaches are to 1) generate detailed body position data at the source, sending frequent updates in DIS Entity State or Data packets, and 2) generate only abstract data at the source. The ramifications of these choices for trainee stations, CGF systems, and the network are described in Table 1.

The TTES system requires the source platform to generate only abstract data, which has benefits for the network and the CGF component. However, the receiving trainee platform is required to interpolate between abstract (DIS) states by playing a stored animation on the image generator. Since the source entity, whether trainee or CGF, may not actually be doing what the animation indicates, correlation problems arise on occasion. The best way to represent human data is still an open question.

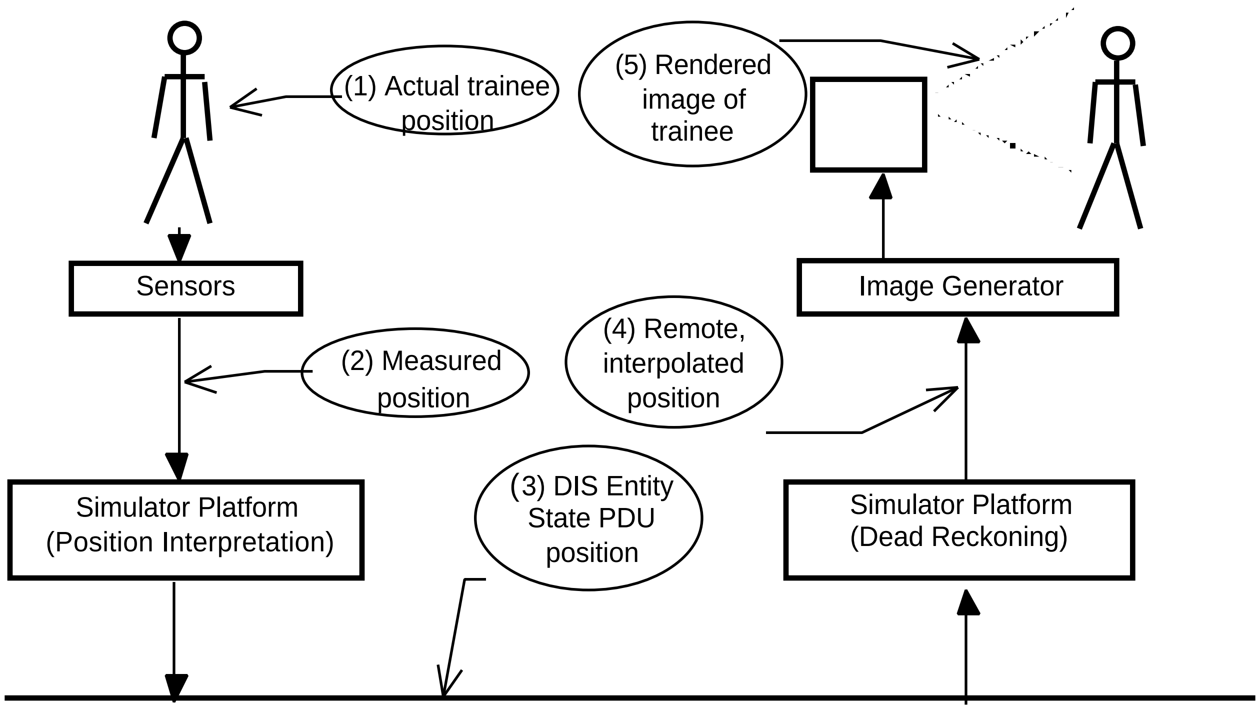

Figure 2. Entity representations in a distributed simulation

Approach to Representing Human	**Source Trainee Platform**	**Destination Trainee Platform**	**Source CGF system**	**Destination CGF system**	**Network**
Generate detailed representation at source	Measures many parts of trainee. Transmits data frequently.	Receives accurate, current data of body position. Renders parts in correct position	Must have high fidelity human movement model to generate body part positions	Must be able to interpret body part positions and motions to understand actions and gestures	High bandwidth required
Generate only abstract body position or activity data at source	Must interpret true body position and produce abstract encoding	Must recreate detailed body position by interpolating between snapshots of abstract data	Produces abstract position information only	Use abstract encodings directly	Moderate-low bandwidth required

Table 1. Requirements on various parts of a distributed simulation from two approaches to representing human body position

PHYSICAL MODELS

Motion and Action Model

The physical level of the IC CGF contains all of the data and procedures that define the characteristics of the entity. The first aspect of the physical level that must be defined for ICs is a model of action. While there is no man-made hull to simulate, there is an analogous body to model with similar physical parameters—maximum speeds, accelerations, etc. Unlike most vehicles, humans can easily move in a direction other than the one they are facing. The human body also has a great many moving parts which potentially increase the complexity of its movement characteristics. Although in TTES body parts are not separately modeled, even the few DIS lifeform (soldier) states mentioned above make the physical model fairly complex. The complexity arises from the interaction of the state variables. For example, what are the constraints on speed while holding the weapon in the firing position? Is it possible to fall prone while running? Is it possible to stow a weapon while rising to a standing position? In addition, the action model must specify how long it takes to perform various actions, and various combinations of actions.

The action model is important not only for CGF IC but for trainee platforms as well. Since the trainees on a TTES platform often cannot physically perform all of the actions they are performing in the virtual world (for example, running), the trainee platform must simulate the capabilities of the soldier using an action model. If high fidelity motions are interpolated in the display system, the speed and range of the motions must correspond to the capabilities and constraints in the action model. All of the models in the system should be the same.

We are developing a model of human action (within DIS) that explicitly describes the state variable interactions, and are making much of it configurable with data files.

Perception Model

A second important aspect of the physical level is the perception model. CGF systems require much more sophisticated visual and audio detection models than do systems that simulate platforms. At the individual level, the sensing characteristics of an entity must reflect the capabilities and limitations of one human. For visual detection, this means modeling multiple, limited fields of view with different acuities and directing visual search with one field based on objects visible in the second field. Hearing also plays an important part in the soldier's awareness of the situation. Sounds such as weapons fire or footsteps indicate the presence of friendly and enemy troops. Typically, the IC recognizes that other soldiers are around him from movement in his peripheral vision or from sound, and turns his gaze to identify the entity.

Our perception model includes both vision and hearing. The vision component, illustrated in Figure 3, includes primary and peripheral fields of view with instant "pop-up" target detection in the peripheral field and foveal search-based detection in the primary field. Detected targets are further identified by fixating the fovea on the target. The hearing component, which includes a simple sound generation and propagation model, allows our CGF ICs to detect and sometimes identify other entities when they move or fire. Loud sounds mask softer sounds. A more detailed description of the perception model can be found in (Reece and Wirthlin, 1996).

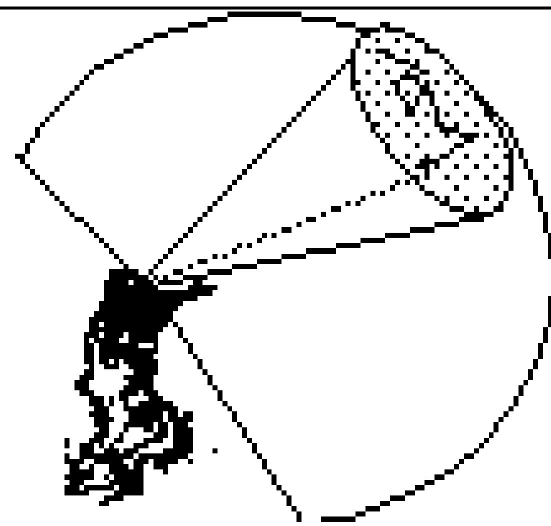

Figure 3. Soldier peripheral field of view with instant detection, and high-resolution primary field of view (shaded) with visual search.

Weapon Control

Since aiming and firing a gun is largely a low level motion control task, we consider weapon control to be part of the physical level. The result of firing a weapon can either be a hit probability or a ballistic round trajectory. The latter is very useful in individual level simulations for several purposes:

- Determining where all of the rounds in a burst from an automatic weapon go. Weapons may have characteristic scatter patterns that reduce the effectiveness of later rounds in a burst. On the other hand, fire sprayed at dense or moving targets might be more likely to hit something.
- Determining where missed rounds go. At close ranges, missed rounds may not land near the target but fly far beyond it. If a CGF IC fires at a crowd of entities and misses its selected target, it still has a good chance of hitting something.
- Determining where unaimed rounds go. Much small arms fire is not aimed at a target, but fired generally at groups of threats or used to suppress threats. Although these rounds are not aimed at a specific target, they may hit one.
- Determining what part of a target is hit.

Our CGF system uses the hit probabilities instead of flying out each round fired. This approach has the advantage of being much less computationally expensive. Hit probability is based on a nominal maximum error radius for the soldier-weapon combination at 100 meters. The resulting hit area is compared with the visible target area projected at 100 meters. This visible target area depends on target stance, aspect, and visible body parts (see Figure 4). Target and firer motion and firer stance are factored in to the error radius. In the future we will also modify the error radius for aim time, wounds, suppression effects, and other factors that affect aim. Missed rounds may be given a probability of hitting other entities within the angle of the error circle if there is a clear line of fire to them.

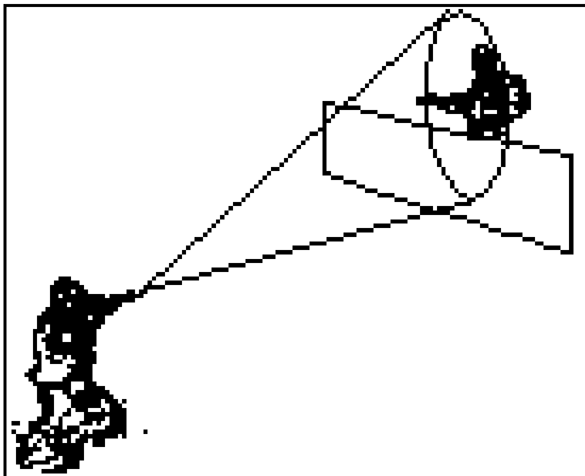

Figure 4. Firer's hit probability is determined by the basic maximum error angle of the soldier with the given weapon and the visible target area.

Wounds

Wound effects are difficult to simulate because they introduce further complications into the action model. For example, if a soldier is wounded in the arm, what actions is he prevented from doing, or from doing as well? It is difficult to find validated models of how wounds affect actions. Instead, wound effects can be modeled in such a way to allow anecdotal data to be reproduced. For example, individuals who continue to function in spite of heavy wounds or individuals who collapse after only a light wound could both be generated with a simple wound effects model.

Our IC CGF system currently has no wound model. A wound model has been a low priority in the project for two reasons: first, there is no wound model for the human trainees; this is probably impossible to implement in any way that is useful for training. When trainees are shot and "wounded," they are informed of that fact aurally, but nothing else happens. The CGF entities are not affected by wounds to make the simulation symmetrical. Second, there is not currently a satisfactory way in TTES to display to the trainee that it wounded a CGF IC.

Fatigue

Unlike platforms which don't fatigue in the short term—other than using fuel—humans are subject to short and long term performance effects from various forms of fatigue. Sprinting, for example, can cause temporary loss of peak performance in less than a minute. Thus fatigue effects can be important even in short exercises. In the longer term, exertion causes a gradual decay of capability—unlike vehicles, which can perform at approximately the same level until they use their last fuel.

We currently use a simple fatigue model that reduces the CGF ICs "energy" as it moves; faster movement uses energy faster. Remaining stationary allows the CGF IC to regain energy. Reduced energy levels cause the CGF ICs movement capability to be reduced. This model is called from the dynamics routine in one place and could be replaced in the future by a higher fidelity model such as IUSS (O'Keefe 1994).

Physical Interactions

Players in an IC-level simulation may come close enough to one another to collide, fight, lift objects together, etc. In addition, entities may wish to pick up objects, use tools, and otherwise manipulate the environment. Providing for this interaction can be extremely difficult and is the subject of ongoing VR research. In our CGF system, we perform collision detection only in two dimensions, effectively treating the ICs as cylinders and obstacles as arbitrarily high walls. The 2.0.4 version of the DIS standard does not provide for physical interactions beyond collisions, so we do not model complex interactions between ICs or object manipulation.

COGNITIVE LEVEL

The cognitive level is responsible for generating all of the entity's behavior. Behavior generation requires situation assessment, terrain analysis, problem solving, action selection, and action control.

Situation Assessment

One component of the cognitive function that is especially important for CGF ICs in urban combat is situation assessment. Due to an ICs limited field of view, the complexity of the terrain, and the fluid nature of urban engagements, the CGF must be able to remain aware of or estimate the locations of entities that it cannot see. These unseen entities should play a part in the CGF ICs tactical reasoning and action selection.

Our CGF uses the fairly standard (e.g. Lind 1995) entity identification levels "detected" (entity presence known, but nothing known about it), "recognized" (class of entity--vehicle, lifeform, etc. is known), and "identified" (everything about the entity is known). In our sound detection model, footsteps and engine noise provide recognition; small arms weapons fire provides identification.

These identification levels are coarse aggregations of facts known about a target and do not really represent knowledge well. For example, it is not clear how useful the distinction between detected and recognized is for ICs. "Identification" is not really identification of specific individual, which could be useful at the IC level. Ideally, all bits of information about an entity could be determined from inference as well as observation. Information includes all publicly observable facts present in an Entity State PDU plus other information such as whether entity is a unit leader, whether it is damaged, what it is carrying, what its age is, how well trained it is, etc. Various observations could fill in different facts about entity. For example, the behavior of a soldier could indicate that it was a unit leader; the motion of a soldier could indicate what load it was carrying; or the path of a vehicle across rough ground could indicate that the vehicle is tracked. This is an area for future work.

We have developed an representation for an entity's mental model of the entities it has seen. When entities are currently visible and detected (to some level), they are "real." Entities that have been detected with sight or sound but are not still visible are given a status of "figment." A complete record of information known about them when last seen is recorded. The positions of figments can be tracked by sound if they continue to make noise. If an observer looks at a location thought to contain a figment but the figment is not observed there or elsewhere, the figment becomes a "ghost." This ghost is known to exist but the observer does not know exactly where it is. (Possible locations may be inferred.)

When an entity is detected, goes out of sight, and then reappears, the observer must determine if it saw the same or different entities. On one hand, there are many details of appearance, equipment and weapons carried, and motion that might allow an observer in the real world to distinguish between individuals. On the other hand, if such discrimination is not possible in the real or virtual world, it could be arbitrarily difficult to determine how many individuals were seen. Sophisticated constraint reasoning would be required to estimate the true situation (e.g. how many individuals were seen at once? Could one individual have moved between the observed locations in the time observed?)

Action Selection

Action selection is the center of cognitive activity for our IC CGF. Decisions at this level initiate all physical and problem solving activities. Outputs can go directly to the physical level to perform actions, to controllers to start continuous monitored activities, or to problem solving modules to start long computations. This system organization is similar to that of other intelligent agent architectures such as (Becket 1993, Gat 1992, and Mettala 1992).

The action selection level consists of a hierarchy of subtasks that allow the entity to decompose its tasks into primitive activities; knowledge, in the form of rules, of what subtasks should be activated to accomplish a task in a given situation; and an inference engine that applies the knowledge to start subtasks and activities.

These processing tasks must be performed under the constraint that the entity stay responsive to changing situations—i.e., reasoning must be done in real time. The action selection computation is intended to be fast so that it (as well as control, physical model, and computations for other entities) may be run frequently and without situation-dependent delays. Long computations are performed at the problem solving level so that the CGF IC always remains responsive even while thinking.

A description of the cognitive level of our CGF system can be found in (Reece and Kelly, 1996).

CONCLUSIONS

We have described a range of areas that are important for simulating an IC in a distributed simulation. The lesson we have learned on the TTES CGF projects is that the level of detail of IC simulation requires more accurate modeling of a number of areas that are not as critical at the larger time and space scales of armored combat. At the individual level, it is literally possible to interact with ICs from a meter away. We have outlined important simulation issues for ICs and presented the approaches taken in the TTES project.

REFERENCES

Becket, W. and Badler, N. (1993) “Integrated Behavioral Agent Architecture”, In *Proceedings of the Third Conference on Computer Generated Forces and Behavioral Representation*. University of Central Florida.

Franceschini R., Petty, M., and Reece, D. (1994). Dismounted Infantry in Distributed Interactive Simulation. *Proceedings of the 16th Interservice/Industry Training Systems and Education Conference,* Section 4-20

Gat, E. (1992) “Integrating Planning and Reacting in a Heterogeneous Asynchronous Architecture for Controlling Real-World Mobile Robots,” In *Proceedings of AAAI-92*, pp. 809-815.

Lind, J., (1995). *Target Acquisition Models for Janus (A)* (NAWCWPNS TM 7811), Naval Air Warfare Center Weapons Division.

Mettala, E. (1992) “The OSD Tactical Unmanned Ground Vehicle Program,” In *Proceedings of the DARPA Image Understanding Workshop*.

O’Keefe, J. (1994). *Factors Governing Dismounted Human Movement in a Synthetic Environment.* U.S. Army Natick Research, Development and Engineering Center.

Reece, D. (1994). Extending DIS for Individual Combatants. *Proceedings of the 5th Annual Conference on AI, Simulation, and Planning in High Autonomy Systems: DIS Environments.* IEEE Computer Society Press.

Reece, D and Tu, H.-K. (1996). Representations of Buildings for Individual Soldier CGF. *Proceedings of the 6th Conference on Computer Generated Forces and Behavioral Representation.*

Reece, D. and Kelly, P. (1996). An Architecture for Computer Generated Individual Combatants. *Proceedings of the 6th Conference on Computer Generated Forces and Behavioral Representation.*

Reece, D. and Wirthlin, R. (1996). Detection Models for Computer Generated Individual Combatants. *Proceedings of the 6th Conference on Computer Generated Forces and Behavioral Representation.*