Proceedings of the 2008 Winter Simulation Conference
S. J. Mason, R. R. Hill, L. Mönch, O. Rose, T. Jefferson, J. W. Fowler eds.

USING AGENT TECHNOLOGY TO MOVE FROM INTENTION-BASED TO EFFECT-BASED MODELS

Andreas Tolk
Robert J. Bowen
Patrick T. Hester

Engineering Management and Systems Engineering Department
Old Dominion University
Norfolk, VA 23529, U.S.A.

ABSTRACT

Following current modeling paradigms, most processes are captured in the form of modeling a desired intent, often using success probabilities. In addition, only special roles that entities are intended to play are modeled. For effect-based modeling, the unintended but nonetheless resulting effects are as important as the intended effects, and they are therefore modeled. Also, uninspected actions based on alternative roles are important. Using agents to not only represent influencers and targets but also the processes, it becomes possible to capture all effects and move from “what I intended to accomplish” to “what I really accomplished,” including side and secondary effects. The agent architecture and a prototype for this effect-based model are presented in this paper.

1 INTRODUCTION

Models are applied to gain insight. Models are developed and implemented as simulations to evaluate systems in situated environments and their interaction with other systems. To this end, modeling is the purposeful abstraction from reality. The two categories that are modeled are entities and processes. These categories can be associated, as processes work on entities, and entities can use processes. Both categories, entities and processes, represent concepts perceived to be important when dealing with the underlying intent for the model application. In other words, modeling is the process of conceptualizing the important entities, processes, and their relations of relevant aspects of the real world. As the model is a purposeful abstraction from reality, it is simpler than the real world. The model only conceptualizes what is needed to represent all relevant aspects of the problem to be solved.

The answer to the question “Why can’t we solve today’s problems with yesterday’s models?” is directly connected with the current modeling paradigm. Most models are *intention-based!* The modeled capabilities of entities are in the majority of cases limited to the intended use and related capabilities. From all possible roles an entity can play in the real world, the main and intended role is selected for the model. The same is true for processes: the desired effect drives how the process is modeled, which is in turn often reduced to the probability of success to have the desired effect. The working hypothesis of this paper is that this modeling paradigm cannot adequately support effect-based modeling. A new modeling paradigm is needed.

This paper describes such a new *effect-based* modeling paradigm based on the agent metaphor. The new paradigm uses agents to represent multiple roles for each of the entities, as well as processes, with their potential effects. In other words: everything becomes an agent, allowing the modeler to capture complex and non-linear systems in which effects and higher-order effects can be generated by the model itself, based on the underlying structures utilizing the emerging features of agent-directed simulation systems. To support the ability of agents to adapt to new environments, more flexible evaluation algorithms are proposed.

This paper introduces the ideas of effect-based models. Following this introduction, the two main shortcomings of the current modeling paradigm are described in the second section. While the examples presented are based on military simulation models, the principles are directly applicable to general research questions. Current evaluations are also explored.

In the third section, a general agent-directed architecture to support effect-based modeling and their evaluation enabling adaptation is described. Focus is given to how this architecture supports a new modeling paradigm that overcomes the identified shortcomings, and a first prototype based on the agent development software NetLogo (Wilenski 1999) is presented. This prototype was developed as a first proof of concept and feasibility. The model implemented is again a military model, but the observed principles are applicable to other domains as well.

978-1-4244-2708-6/08/$25.00 ©2008 IEEE

Finally, results of the work done so far are summarized. It should be pointed out that the research on this topic has just begun and the authors welcome discussions.

2 CURRENT MODELING PARADIGMS

When building a model, the model builder has a predetermined purpose in mind. He wants to support the analysis of several alternatives, optimize a decision based on several constraints, train people using a simulator or a simulation system, etc. In any case, he first builds a model derived from the real world (where the real world may be a hypothetic world describing the constraints and the solution space he is interested in). This world is captured in the form of concepts. The concepts can either be a feature or a fluent, as summarized by Sandewall (1994). While a feature is something that is situation independent and describes entities, a fluent is situation dependent and describes processes (or better said states and situation dependent properties representing this category). In other words, the modeling process results in entities, processes, and their relations.

An entity can play many roles. In modeling processes, an entity is often reduced to a main intended role. This is discussed in the first subsection. A process is directed to change a current situation into a desired direction: starting with the status quo, a process operates on the current entities and changes their properties with the intention that these changes lead to the desired outcome. The intention can be expressed in the form of a desired state change. An example in the military context is to stop an enemy attack; another example in the transportation domain is to transport passengers from one place to another. When developing the model, processes are often reduced to the desired effect, very often modeled in the form of a random experiment (e.g., probabilistically simulating the roll of a die or flip of a coin) capturing the likelihood that a process results in the desired state change. As with entities, this model paradigm reduces the effect of a process to the desired state-change; as an example, a modeler may want to change the status of the player from poor to rich. The resulting shortcomings will be captured in the second subsection.

2.1 Intention-based Capability Modeling

The examples for role-based modeling are taken from the military domain, but they can easily be generalized. Military units can fulfill many tasks. Military history is full of examples where units accomplished a task that they were not originally designed for, but the situation required the use of available capabilities in support of an operation. The use of air defense guns against tanks is a technical example, while the use of logistics units to fight as infantry units against a break-through tank unit is an operational example. Today, nearly every military unit must be prepared to serve as military police or security units, no matter if infantry, engineer, or artillery.

However, traditional combat models reduce the units for modeling to the intended role and the necessary minimal properties to save computational effort. For example, an artillery unit is often modeled as a number of howitzer systems that are equally distributed in a circle, if the dislocation area is modeled at all. The intended role for such a unit is to fire with indirect fire weapons at an enemy. If other capabilities are needed, such as that an artillery unit has many soldiers and therefore can be assigned nearly every task an infantry unit can do, this is not possible.

This reduction to the minimum number of elements needed to model the main intent can be observed in many simulation systems. In transportation models, roads are reduced to lines and networks, evacuation simulations use flow models for the traffic, etc. For example, the main role of a bridge is to provide transportation. At the same time, the same bridge also provides means for the power systems, sewage, telephone landlines, cell phone towers, etc. If this bridge gets destroyed by an accident or an earthquake, similar to what recently happened in Minneapolis in August 2007, not only the primary role gets disrupted, but potentially all secondary roles as well, like a local hospital is suddenly without the necessary power, or emergency workers cannot use their cell phones for coordination, as the area is no longer sufficiently covered. A simulation system used for training or analysis must therefore provide this information.

It should be pointed out that this reductionism is not wrong, as the results are understandable and efficient simulations, but the price is that the use is limited to scenarios in which the modeled units stick to their main role. Thus, using these models to analyze scenarios outside their scope is difficult, if possible at all.

As shown in the example above, this observation is true in other – non-military – domains as well. Models developed to support the design are used for experiments like crash-tests in manufacturing. Static street maps are reused to generate transportation nets for traffic optimization. As with the examples above, the conceptualization of an aspect of reality is reused in a different context, requiring a new aspect – and therefore a new or extended concept. If a simulation system is used to support training or evaluation and analysis, all relevant roles need to be supported. In general, intention-based capability modeling focuses on the main role, which can limit the applicability for new domains with changing scopes.

2.2 Intention-Based Process Modeling

When modeling processes, the current modeling paradigm is even more extreme; however, it is less perceived as reductionism by traditional modeling experts. The example shown in figure 1 is again taken out of the military domain.

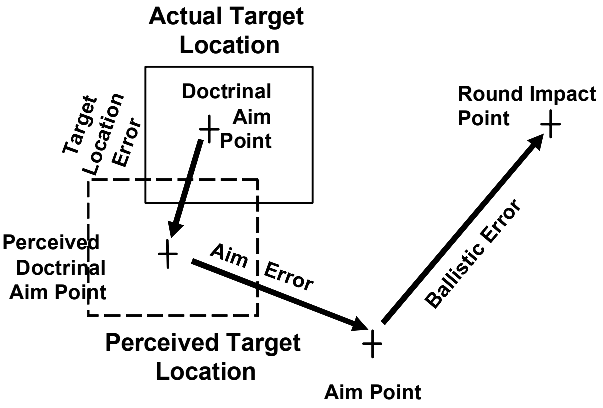

Figure 1: Accumulation of Errors

If a tank engages in a firefight with an enemy – and let us assume this is a tank as well – this process is conducted with the intent to destroy the target. However, not every shot hits, and not even a shot that hits always leads to sufficient damage. Therefore, most combat models use a series of probabilities describing how errors accumulate to find the real impact point first. Similar probability chains (such as Hidden Markov Models) are used to find out if a hit results in damaging the target. Typical damage categories are movement, communications, fire power, or catastrophic damage. These processes to determine the kill-probability and even use error accumulation can become pretty complicated. However, the process is reduced to compute the probability that the intended result of the modeled activity is achieved or not: damaging or destroying the enemy.

The main problem is that this model does not address what happens to the so called "friendly fire," shooting erroneously at one's own units. The so-called hit and kill probabilities are normally only defined for opposing forces, and do not include the chance of damage by an ally. As a result, unintended events are not modeled and secondary effects or side effects are not part of the model or the system.

Why it is generally important to not only model what is intended and how likely it is, but also what can happen that is not intended is obvious in the shooting example in the military domain: it is not only important that a target was missed, it also important what was hit by the bullet. If an air strike against a hostile headquarter destroys a school instead, this is an important event. If the model only supports the evaluation of how often the air strike misses the headquarters, it is not sufficient in support of planning and training procedures.

In general, when following the intention-based paradigm a process is modeled as the likelihood of the desired outcome. Unintended outcomes, side effects, and follow-on effects are normally not modeled. This is not sufficient.

2.3 Intention-based Evaluation

Very similar observations can be made in the domain of metrics as well. The measures of performance used in after action reviews for efficiency evaluation are often tightly coupled with and limited to measuring intended effects. In principle, this is not a bad thing, but if the measures of merit are not extensible to the new task, this is a serious challenge, in particular when parameters needed for evaluations are hardly or not at all obtainable from the system to be evaluated. Nonetheless, kill-ratios, killer-victim-scoreboards, and other attrition-oriented measurements are still the dominant performance measures.

The NATO Code of Best Practice introduced a more flexible system of measures of merit. Green and Johnson (2002) also request more flexibility in measuring model-based battle success. However, while the theory stands, practical applications are not published.

In summary, intention-based modeling reduces entities and processes to the intended main properties, capabilities, and effects. Evaluation procedures are also often fixed and inflexible regarding new objectives.

3 EFFECT-BASED MODELING

The term "effect-based modeling" in the military domain is often tightly connected with effect-based operations as described in detail by Smith (2002). He defines effect-based operations as *"coordinated sets of actions directed at shaping the behavior of friends, neutrals, and foes in peace, crisis, and war."* The application domain of military command and control is no longer limited to the military domain, but must be seen in the context of policy and economy as well. In other words, the scope broadens significantly.

Furthermore, effect-based operations introduce the idea of direct physical effects, related indirect psychological effects, cascading indirect physical effects, and related indirect psychological effects. Figure 2 exemplifies these ideas.

In general, the idea is that not only entities can produce effects, but effects themselves can produce effects. These effects can be intended, or can be unintended as well. It is possible, for example, that a bridge is destroyed to disable an enemies' advancement. The same bridge, however, may also be the only bridge connecting a local village with the local hospital and its destruction has significant negative effects on the relationship between the armed forces and the local community.

As mentioned earlier, effect-based models must also support the variety of capabilities of units. Instead of modeling units with role-derived properties and capabilities, each military capability and means is modeled individually. A unit is then easily defined by the number of possible roles and underlying capabilities.

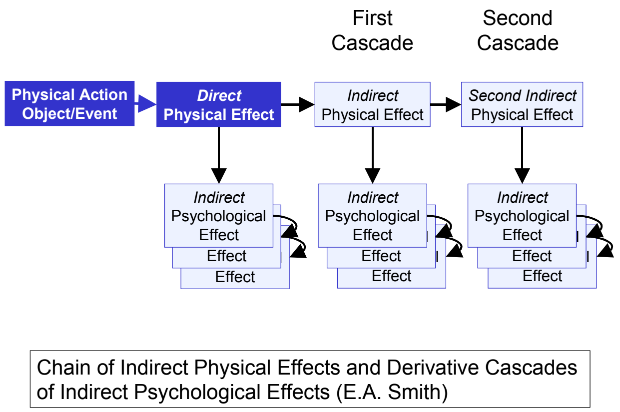

Figure 2: Chain of Effects (Smith 2002)

In the following two subsections, the capability based modeling for features or entities and the effect-based modeling for fluents or processes are defined in more detail, before a first prototype is described in the last subsection of this section.

3.1 Capability-based Modeling for Entities

Modeling results in a formal representation of an analyst's conceptualization of the system and related activities of interest. As a result, each simulated entity is modeled regarding its characteristic properties, its behavior, its functions and capabilities, and its associations.

As stated earlier, traditional modeling focuses on the intended use, not inherent capabilities. The approach recommended as a result of our research is to identify capabilities based on available properties. This allows a modeled entity to not only conduct all intended functions, but it can also provide all possible capabilities. To conduct defense operations against tanks after a successful breakthrough does not require a special anti-tank unit. Each unit is equipped with enough – hopefully educated – personnel and anti-tank weapons. The motto of the US Marines "every Marines a Rifleman" reflects this idea quite well: if a unit has soldiers equipped with rifles, it can support infantry operations.

Another point of interest is to distinguish between potential and actual capabilities. In most cases, it requires resources to instantiate a potential capability, which means to make it an actual, available capability. Finally, it is of interest to distinguish between the capabilities (what a concept can do), the purpose (what a concept was made for), and the use (what a concept is used as). The difference between intended use and actual use is often significant. The creativity of combatants and insurgents regarding how to use objects available for their purpose seems to be without limit, and purposeful abstractions of reality should support this.

In summary, each simulated entity should be described in sufficient detail using properties and associations. Capabilities, purposes, and uses should be connected to properties, not separate modeled units. This allows a simulated unit to support any potential role it may play regarding its characterizing properties.

3.2 Effect-based Modeling for Processes

The second domain that needs significant improvement is the concept of processes. As shown in section 2.2, it is insufficient to only model the intended outcome. Processes need to be described and modeled with the same detail used for entities and their states in traditional modeling. In recent years, the M&S community focused on events and event-driven simulations. M&S formalisms focus primarily on state changes in discrete systems that are triggered by events and that may cause new events, which also result in new state changes.

Sowa (2000) describes processes by their starting and stopping points and by the types of changes in between and categorizes them by distinguishing between continuous change and discrete change as well as if the starting and stopping points are explicit or not. Figure 3 shows the taxonomy of processes resulting from these definitions.

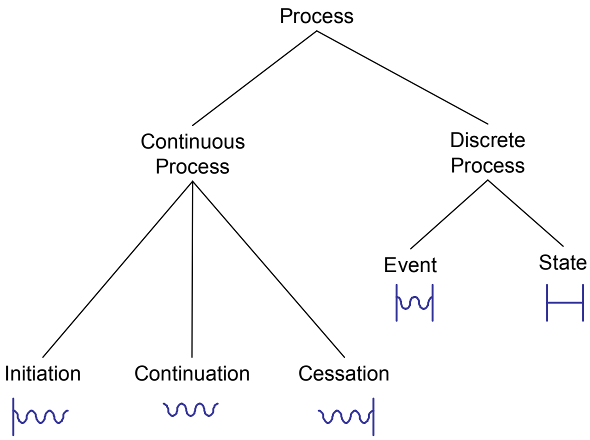

Figure 3: Categories of Processes (Sowa 2000)

What is even more important is the potential that, although the intended outcome is not achieved, another unintended outcome is achieved, as discussed earlier in the paper. In addition, every "kind of change that takes place in between" the starting and the stopping point can trigger another process, which can be as complex as the process, it was initiated by, etc.

It is therefore necessary to model not only the intended outcome, but the whole process and its possible interactions with all entities and other processes. If, for example, one simulated unit jams a hostile radio signal while at the same time a neighboring unit tries to submit an important situation report, it is possible that these two processes will influence each other. Additional examples will be given in section 3.4.

In summary, each process needs to be captured with the same detail as entities, their properties, and associations. Reduction of processes to discrete events only covers a fraction of possible categories. To focus only on the conceptualization of entities and their relations is not sufficient. The M&S community is currently focused on data and data exchange; it is necessary to apply the same diligence to engineer, document, and align processes.

3.3 Flexible Evaluation Algorithms

It would be wrong to say that all current combat simulation models use old metrics to measure intended effects. However, after action review systems are often still very conservative and even more often not usable in support of adaptive processes needed for intelligent systems, such as intelligent agents taking over the role of human decision makers, in particular when many replications of one simulation run are needed or when the simulation runs faster then real time, as is often the case in the analysis community.

In particular, when agent-directed simulation is used in support of computer generated forces and human behavior modeling, the danger to run into structural variances based on misaligned evaluation criteria is imminent, as shown in Tolk (1999). It is necessary that the internal decision logic, the external evaluation logic, and the modeled entities and processes are aligned with each other. The *internal decision logic* controls the behavior of the simulated entities. Based on the perceived situation of its environment (including other entities), the internal decision logic selects (or constructs) the behavior of the entity. If the objectives targeted by these internal decisions differ from what is evaluated by the *external evaluation logic*, structural variances occur. The metrics used, which should be driven by operational requirements and can only measure what is modeled, should drive the internal decisions as well as the external evaluation. Therefore, configurable and adaptable metrics are needed that can be used as fitness functions for internal decisions as well.

In support of NATO evaluations regarding conventional stability in middle Europe, more than 70,000 simulation runs had to be evaluated and used to support the configuration of internal decision rules (Hofmann et al. 1994). An expert team at the university of the federal armed forces developed an evaluation algorithm that was based on the following general principles:

- Objectives can be captured in a utility function that combines multi-criteria for success
- Success criteria are defined regarding the status of own forces as well as opposing forces at the end of a simulated operation. Examples are the objective to destroy an enemy unit using artillery fire (metric: how many forces survive in the target area) or to take an objective area with an infantry attack (metric: how many own forces are in the target area after the operation).
- Success criteria are defined using the descriptions of entities and their properties as well as processes. If the same result is reached faster, the success is normally higher. Accordingly, if opposing forces are denied from reaching their objectives for a longer period of time, the success factor should increase.

These ideas were successfully presented in NATO (1995), but are currently gaining more attention than 10 years ago, as they can be used for internal adaption of learning rules for intelligent agents as well as external evaluation of success of operations.

4 FIRST PROTOTYPICAL EXPERIMENTS

In a thesis conducted in the M&S program of Old Dominion University, prototypical implementations of these ideas have been implemented. Cares (2004) Information Age Combat Model (IACM) built the theoretic frame for an agent-based simulation implementing multi-role agents as proposed in section 3.1, agent-based processes as proposed in section 3.2, and flexible evaluation algorithms as proposed in section 3.3. Although this model is very limited regarding operational functionality, it establishes an extensible framework that the authors are willing to share for academic collaboration (Bowen 2008).

4.1 Using the Information Age Combat Model

The IACM was developed with the intention to model the concepts of Network Centric Warfare (NCW) as a network of interrelated nodes representing the key activities of NCW: sensing, influencing, deciding, and targeting. These activities define the key functions of all combat and combat-related platforms and entities and may exist individually or in any combination on each different platform. The IACM furthermore defines links between each node in the network and each of its neighbors by defining the directionality of each link or the lack of a link if the nodes do not relate directly.

Figure 4 shows the resulting logic: a target (T) is visible to sensors (S). Sensors provide the input for the decision maker (D), who directs influencers (I) to engage known targets.

With this simple model, Cares was able to show that NCW requires a balanced approach to sensors and influencers. He was also able to show effects beyond the attrition-oriented traditional combat models, such as effects of separating sensors and influencers.

In order to support the creation of an agent-based model to support net-centric and effect-based evaluation, this model was slightly modified. First, the element of communication (C) was introduced as a node of its own. If the communication between sensor and decision maker is not available, the perception of the decision maker will not be accurate. Also, if the communication between the decision maker and an influencer is not available, he cannot order any actions.

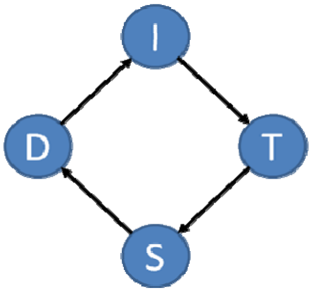

Figure 4: IACM of Cares (2004)

The second enhancement was replacing the secure links with links that are defined probabilistically. The probability of sensing (the likelihood that a sensor really detects a target) is influenced by many factors, such as the effective range of the sensor, the line of sight between the target and sensor, if the sensor is actively sensing, if the target doing something detectable (such as moving, shooting, communicating), types of sensor and target, etc. Similar observations are true for influencing and communicating. These enhancement lead to the model used in the prototypical evaluations shown in Figure 5:

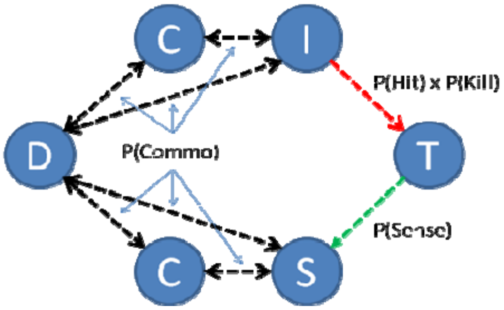

Figure 5: Enhanced IACM Model

These roles in the enhanced IACM Model are defined by properties that are used to describe the capabilities of agents. An agent can expose the properties needed for decision making and influencing, representing, e.g., a tank with the platoon leader on board. Also, autonomous systems having all capabilities can be modeled. Each platform, however, is at least a target to the other sides.

4.2 Implementing the Model in NetLogo

To proof the concepts and the feasibility, the ICAM was implemented using the free software NetLogo. Each modeled entity became an agent with the possible roles defined by ICAM.

As pointed out before, the intention based modeling of desired outcomes of processes is not sufficient. Therefore, the model uses agents to represent these processes: sensing, communicating, and influencing. In the current version, the focus is on sensing and influencing. Figure 6 shows how sensing is modeled as a process.

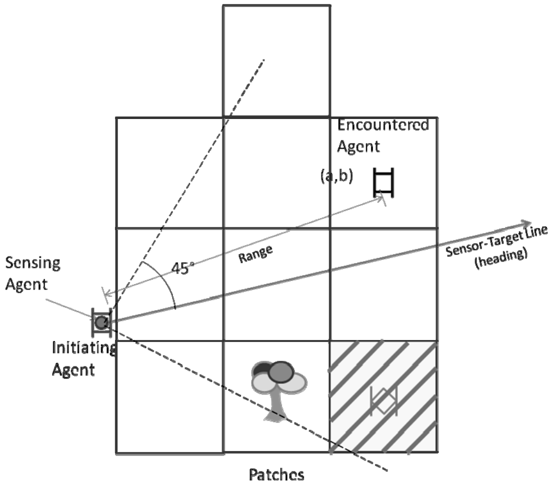

Figure 6: Modeling Sensing

In NetLogo, agents are modeled as turtles or patches. In the prototype, the simulated entities are modeled as turtles, but the embedding environment is modeled by patches. In addition, some objects that are turtles become part of the environment as well, such as smoke. Without going into the details, the process for sensing interacts with all potential other agents to find out if he can sense them or if they influence his sensing in any way. The tree in Figure 6, for example, blocks the view of the tank, if optical senses are used. In the same way, own or friendly forces can "be in the way" of sensing processes. Figure 7 shows the principles for influencing as currently used.

Comparable to the sensing model, the influencing process is an agent that communicates with all potential targets and influencing processes. The model allows the agent to make mistakes, including picking the wrong target. It also allows for friendly forces to become the target. Similar models are implemented for artillery fire as well.

Finally, the ideas of the NATO report (1995) were used to set up configurable metrics, based on the definition of areas and the status of simulated entities within these areas. Each side defines a set of areas in which they want to have own forces and a set of areas in which the want to have no hostile forces. These areas can be weighted using operational values. The success factor of a force is calculated by the percentage of own forces in the desired areas minus the percentage of hostile forces in areas they should not be in, potentially weighted by the operational factor for each area.

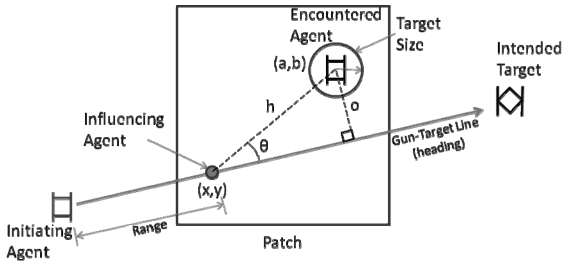

Figure 7: Influence Modeling

It is worth mentioning that this allows for individual accounts of success for all participating parties. If the objective of an attacker is an area the defender is not interested in, both can evaluate their operation as being successful.

4.3 First Results Using the Prototype

The main focus of this paper is to introduce the reader to the ideas of effect-based modeling of entities, processes, and metrics versus the traditional intention-based approach. The first application is documented in Bowen (2008). The task of his work was to set up a simulation framework enabling the evaluation of effect-based and net-centric operations. Therefore, the data used and the processes implemented are based on the subject matter judgment and modeling assumptions of the authors. This may happen in a future phase, but the current experiments were designed to show the feasibility of the ideas presented in this papers. The authors are actually considering making the model available as open source to generate more feedback and encourage collaboration. Currently, these first results provide a tool to use to show and derive effects that could not be modeled and evaluated with traditional models.

As stated before, the NetLogo model (Wilenski 1999) was used as the development tool in order to implement the IACM under a range of environmental conditions. Other agent based modeling development tools are possible, but have not been evaluated by the authors for this purpose. The first group of experiments relied on a series of similar base scenarios starting with a completely unconstrained environment and progressively adding more environmental complexity while constraining the ranges and effectiveness of each of the modeled combat functions to observe the results. It relied on both quantitative and qualitative metrics to assess the effectiveness of the model. The quantitative measures assessed the consistency of the model across the spectrum of environmental conditions to determine if the base model is independent of the constraining effects of the environment or if the environment produces a significant change in the model results. The qualitative measures compared the natures of the interactions and maneuver of the combat systems to assess if their emergent behaviors are realistic in the sense that they appear to act and interact in a manner consistent with what one would expect to see in the real world.

The implementation relied on three major agent types, environmental agents, combat system agents, and effect agents.

- The *environmental agents* represented trees, walls, and smoke. They are inanimate and served the role of impediments to the combat tasks of move, sense, and influence in order to create an adaptable environment in which to test the IACM.
- The environment was populated with *combat system agents* representing tanks, scouts, commanders, and signal trucks. The combat system standard framework gives them the capability to move, sense the environment, communicate, influence other combat systems, and make decisions. Their primary role within the model defined the degree to which they were able to perform these capabilities.

  The agents followed simple commands according to a set of rules that governed how they moved, and interacted with the environment and each other. These agents served the roles of the *targets* in the IACM and were the moving platforms that carried the capabilities of performing the functions of sensing, influencing, and communicating.

  In the experiments, tanks were capable of all three functions; scouts could simply sense and communicate; commanders could only communicate; and signal trucks could detect communications and rebroadcast them in order to extend communications ranges.
- All of the interactions between agents were enabled through the use of *effect agents* representing sensing, influencing, and communicating.

  Sensing served the role of enabling perception. The model relied solely on direct line of sight visual sensing as opposed to more advanced thermal and RF detection capabilities. This was to simplify battlefield conditions in order to scope the model to a reasonable level of complexity. The sensors evaluate a given area for a period of time with effectiveness attenuating as the distance from the source increases.

  Influencing agents model the actual projectile in flight allowing it to evaluate each contact encountered in turn. The projectile could hit environmental agents or combat system agents of both sides.

  Communicating has an attenuating range of effectiveness. Communicating is not directional and cauld be detected from anywhere.

  Effect agents were created as needed within the

> model. When manifested these functions are intended to complete the control and combat decision cycles of which they are a part. If for some reason the intended result does not occur due to missing the target, fratricide, etc. then the network cycle is completed with the effect applied to the unintended target, if the related effects could be observed and were perceived correctly.

The essential element of these first experiments was for command agents to evaluate the current state of the model and make appropriate decisions about where and how to best employ the capabilities of each combat system against a competitive opponent. Each experiment produced a unique result as environmental conditions changed. Each result – intended or unintended - triggered a new decision cycle enabling the combat system to take the appropriate action within the environment to rectify problems and try again or to take advantage of a situation.

Generally, the result of this type of interaction was to enable agents to make decisions that benefited from the use of the environment or to modify behavior based on changing effects. Here is one of the observed examples: if a target area becomes concealed by smoke, the sensing agent informs the scout that the target cannot be seen. The agent now must determine a new location from which to view the target that accounts for the smoke, move there, and try again.

The use of effects agents also enabled agents not directly involved with the interaction to still benefit from it. If a target was engaged in close proximity to other agents who were actively sensing, the sensors could detect the influencer, smoke, and the impact produced by the interaction and pass the information to the combat system agent, who in turn could use the information to become aware of and react appropriately to the contact without actually being in a position to detect the shooter. An interesting effect of this was the resulting close quarter battles that would occur in tree covered areas. Combat system agents were forced to jockey for position to gain a clear line of sight to their target through the trees as well as having to avoid friendly systems that were now legitimate targets for the influencing agents. At the same time agents would react to new contacts hidden from view by the trees based on the bullets impacting in the area.

The results of the first experiments were compelling and demonstrated the effectiveness first of agent based modeling to represent network centric operations and secondly the employment of effect agents to allow a more realistic agent interaction with the environment and to extend the effect of an interaction beyond the two primary parties involved. Although the model is just a demonstrator, it serves as a proof of principle, proof of concept, and proof of feasibility for the ideas presented in the section 2 and 3 of this paper, including supporting a new paradigm for modeling of effect-based operations.

5 SUMMARY

In order to support net-centric and effect based operations, intention-based modeling falls short. Models must be driven by elementary capabilities connected with properties describing simulated entities as well as processes. Features and fluents as defined in the artificial intelligence communities are needed. Discrete event simulation is not sufficient. Agent-directed simulation provides the metaphors needed to build the necessary models. Computational challenges exist, but they seem to be easier to overcome than conceptual weaknesses of alternatives. An enhanced model is proposed for use to solve the discussed problems.

REFERENCES

Bowen R. J. 2008. *An Agent-based Combat Simulation Framework in Support of Effect-based and Net-centric Evaluation.* Thesis at Old Dominion University, Norfolk

Cares, J. R. 2004. *An Information Age Combat Model.* Technical Report, Alidade Incorporated, produced for the Director, Net Assessment, Office of the Secretary of Defense under Contract TPD-01-C-0023

Green J. M. ; Johnson BW. 2002. Towards a Theory of Measures of Effectiveness. *Proceedings of the CCRTS*

Hofmann, H. W., Schnurer, R., Tolk, A. 1994. *KOSMOS Simulation Experiments on Stable Defense,* Technical Report NATO Research Group 18, AC/243(Panel 7)TR/5(IV), The Hague, The Netherlands

North Atlantic Treaty Organization (NATO). 1995. *Symposium on Coping with Uncertainty in Defense Decision Making*, NATO Technical Proceedings AC/243 (Panel 7) TP/9, The Hague, The Netherlands

North Atlantic Treaty Organization (NATO). 2002. *NATO Code of Best Practice for Command and Control Assessment.* Rev.Ed. CCRP Press, Washington, D.C.

Sandewall, E. 1994. *Features and Fluents*. Clarendon Press, Oxford

Smith, E. A. 2002. Effect Based Operations: Applying Network Centric Warfare in Peace, Crisis, and War. *CCRP Press*

Sowa, J. F. 2000. *Knowledge representation: logical, philosophical, and computational foundations*. Brooks/ Coole

Tolk, A. 1999. Non-Monotonicities in HLA-Federations. *IEEE Proceedings of the Spring Simulation Interoperability Workshop,* IEEE CS Press

Wilenski, U. 1999. NetLogo. Center for Connected Learning and Computer-based Modeling, Northwestern University, Evanston, IL `<http://ccl.northwestern.edu/NetLogo>`

AUTHORS BIOGRAPHIES

ANDREAS TOLK is an Associate Professor for Engineering Management and Systems Engineering of Old Dominion University's Modeling, Simulation, and Visualization Faculty. He is also a Senior Research Scientist at the Virginia Modeling Analysis and Simulation Center (VMASC). He holds a M.S. in Computer Science (1988) and a Ph.D. in Computer Science and Applied Operations Research (1995), both from the University of the Federal Armed Forces of Germany in Munich. His e-mail address is `<atolk@odu.edu>`.

ROBERT J. BOWEN is a Major in the U.S. Army (FA-57) and graduate student in the Modeling & Simulation Program of Old Dominion University. In his Master thesis (2008), he implemented first ideas presented in this paper. His e-mail is `<robert.bowen@us.army.mil>`.

PATRICK T. HESTER is an Assistant Professor for Engineering Management and Systems Engineering of Old Dominion University's Modeling, Simulation, and Visualization Faculty. He holds a Ph.D. in Risk and Reliability Engineering (2007) from Vanderbilt University and a B.S. in Naval Architecture and Marine Engineering (2001) from Webb Institute of Naval Architecture. His e-mail is `<pthester@odu.edu>`.