Multi-Level Hybrid Behavior Model of Computer Generated Forces

(Extended Abstract)

Hyeoneun Kim
KAIST, Korea
hekim@ai.kaist.ac.kr

Bongseok Goh
KAIST, Korea
bsgoh@ai.kaist.ac.kr

Bowon Nam
KAIST, Korea
bwnam@ai.kaist.ac.kr

Kanghoon Lee
KAIST, Korea
khlee@ai.kaist.ac.kr

Junseok Lee
KAIST, Korea
junseok.lee@kaist.ac.kr

Jeong Hee Hong
KAIST, Korea
jeonghee.hong@kaist.ac.kr

ABSTRACT

Computer Generated Forces (CGFs) refer to the simulation models of combat entities. While the holy grail of CGFs is the realistic reflection of the entities, it is difficult to achieve since the model is often too sophisticated to be replicated. Traditional models which translate field manuals to descriptive models generally produce reliable behaviors, but concern about being brittle in undescribed or unexpected situations is still remaining. In this respect, automated planning approaches can produce robust behaviors for dynamic situations, but the computational resource is too demanding to compute full-scale solutions. This paper proposes a multi-level behavior modeling approach that adopts the knowledge engineering approach to describe high-level tactical behavior rules and the automated planning approach to compute low-level combat actions in dynamic combat situations. We show that this two-level approach ensures reliable behaviors with moderate computation time.

Categories and Subject Descriptors

I.6 [**Simulation and Modeling**]: Model Development—*modeling methodologies*

General Terms

Design, Experimentation

Keywords

CGF, DEVS, POMDP, Agent-Based Simulation

1. INTRODUCTION

Combat simulations require behavior models of combat entities, such as weapon systems and human soldiers, which are referred to as Computer Generated Forces (CGFs). As a complex multiagent model, CGF model can be useful for estimating the military policy making if it describes agents' behavior realistically and reflects interactive behaviors among agents including both mechanical and interpersonal skills [5][6].

Ideal CGF models should generate realistic behavior with sophistication. On the other hand, we should be aware of computational resource requirements to replicate the simulation for testing statistical analyses. The objective of our work is to seek the balance of the realism and the efficiency through the multi-level behavior modeling approach.

The traditional approach to CGFs can be characterized as a knowledge engineering approach [7]. This approach mostly relies on subject-matter experts and field manuals to build an explicit set of behavioral rules for CGFs. Admitting successes in this rule-based modeling, there has been a persistent concern about the limitation of describing the complex human behaviors in every detail. There exists a combined battle framework of mission and engagement level models [3], but lower level model of this framework still focuses on the desciption of military doctrine in depth.

A recent alternative is a decision-theoretic modeling, e.g. Markov Decision Processes (MDP) [1]. If the MDP model has realistic state transitions and a suitable reward function, this approach should result in a rational behavior model, but computation takes significant time hindering replication.

This paper proposes a hybrid approach that combines the best of the two worlds. Specifically, we adopt a two-level hierarchical model where the high-level strategic behaviors are encoded as a Discrete Event Systems Specification (DEVS) [9] and the low-level tactical actions are obtained from a Partially Observable Markov Decision Process (POMDP) planner [8]. We develop a combat simulation by following this hybrid approach, and present the simulation result highlighting realistic combat actions.

2. DEVS-POMDP HYBRID MODELING

The CGFs inherit the basic structure and characteristics of agents (Fig. 1). They have a decision-making module to exhibit autonomous combat behavior under limited perception of the environment and limited scope of action to interact with other models. Our hybrid modeling mainly expand the decision-making module with a hierarchy of multiple levels. The upper level handles the tactical guidelines that CGFs are bounded. The lower level models the rational combat actions constrained by the guidelines. This can be thought of as a hierarchical planner [2] with adaptation to the application of agent-based simulations.

The upper-level decision-making takes role to show trained behavior following the field manuals. Despite the ambigu-

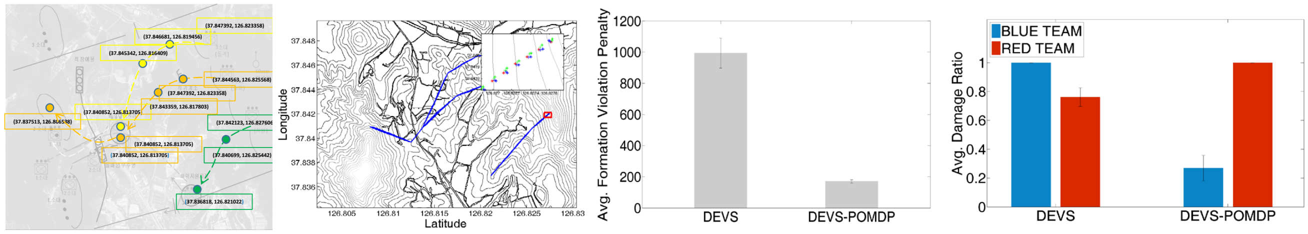

Figure 2: (Leftmost) Combat scenario of interests, (Left) Maneuver agents and magnified view of V-shaped formation of a fire team, (Right) Formation violation penalty that is low when V-shaped fire team formation is better maintained, (Rightmost) Comparison of standalone and hybrid simulation combat outcomes

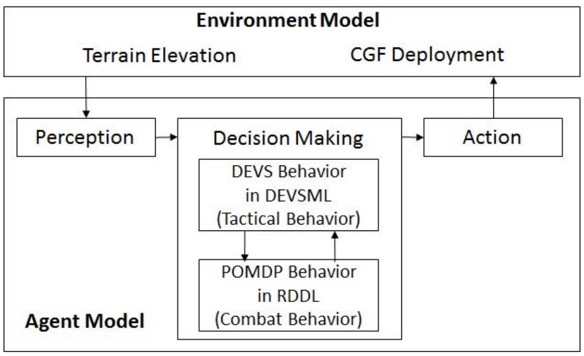

Figure 1: Architecture of Combat Simulator

ous guideline about combat actions, the description at the tactical level is complete enough to constrain the agents to behave within the tactical guidelines. We use DEVS, a formal model for discrete event systems (DESs). DEVS defines the system behavior using state transition tables by a DEVS atomic model, and supports modular and hierarchical specification of complex behaviors by DEVS coupled model. Benefit of using DEVS is a clear and formal specification of what the agent does in which events, allowing us to transcribe the doctrines relatively easily. In our implementation, the model is represented by the DEVSML and the model is executed using DEVSim++.

Fig. 3 is Maneuver DEVS atomic model. This model transfers the maneuver information from soldier agent to behavior planner (control and observation message) and vice versa (planned action to be done).

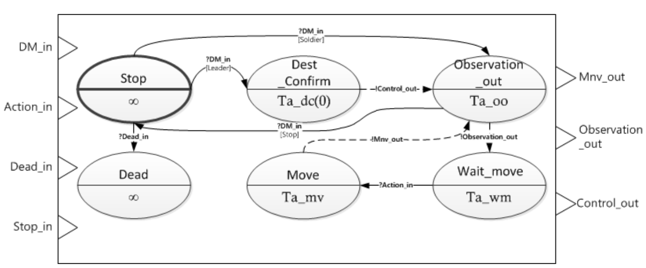

Figure 3: DEVS Diagram of Manuever Model

The lower-level decision-making fills in the detailed combat behavior. We use POMDP, a model for automated planning in partially observable and stochastic environments. Though it provides a rigorous mathematical framework for

(1) Define nominal belief-state sequence $(\hat{b}_0, \ldots ,\hat{b}_{H-1})$
(H: given planning horizon. $\hat{b}_0 = b$)
$\hat{b}_{t+1} = \Phi(\hat{b}_t, a_t)$
where Φ : nominal belief state update assuming
(2) Define approximate value function

$$V^{*}(b) \approx \sum_{i=0}^{H-1} R(\hat{b}_i, a_i, \hat{b}_{i+1})$$

(3) Optimize over action sequence $(a_0, \ldots, a_{H-1})$

Table 1: Pseudo-code for NBO algorithm

planning, it is known to be computationally intractable. We used Relational Dynamic Influence Diagram Language (RDDL) to represent the combat tasks and Nominal Belief Optimization (NBO) algorithm [4] to compute approximately optimal combat actions. Table 1 is the pseudo code for NBO algorithm. It gets the action list $(a_0, \ldots, a_{H-1})$ given from belief state b. Then it calculates the reward function with sampled next belief states to choose the proper action.

3. BATTLE EXPERIMENTS

Our combat scenario consists of 150 agents (90 blues and 60 reds). The maneuver paths of three platoons are illustrated in Fig. 2 (Leftmost). These paths are tactical decisions modeled in DEVS, and at the higher resolution, we modeled the V-shaped formation in POMDP. Fig. 2 (Left) shows Matlab visualization results of maneuver paths and formation of teams adjusted by POMDP. In contrast to the standalone model, the hybrid model was better at keeping the formation with less amount of modeling effort (Fig. 2 (Right)). Other tactical missions such as engagement with rifles were also modeled using the DEVS-POMDP hybrid approach in a similar fashion. Fig. 2 (Rightmost) depicts that these hybrid models resulted in significant changes to the final battle outcomes compared to the standalone model.

4. CONCLUSIONS

We presented a DEVS-POMDP hybrid approach to modeling CGFs in military simulations. The advantages of our approach are (1) DEVS effectively restricts the search space for the POMDP planner for efficient planning, (2) DEVS does not need micro-details on which action to execute in every situation since the POMDP planner takes the responsibility of filling in low-level actions, and (3) the overall behavior is guaranteed to be consistent with the high-level strategy specification which is important for verification and validation purposes.

5. ACKNOWLEDGEMENTS

This work was supported by Defense Acquisition Program Administration and Agency for Defense Development under the contract UD110006MD, Korea

6. ADDITIONAL AUTHORS

Additional authors: Jaehyun Park (Agency of Defense Development, Korea, email: forehand@add.re.kr), Il-Chul Moon (KAIST, Korea, email: icmoon@kaist.ac.kr) and Kee-Eung Kim (KAIST, Korea, email: kekim@cs.kaist.ac.kr).

7. REFERENCES

[1] D. Aberdeen, S. Thiébaux, and L. Zhang. Decision-theoretic military operations planning. In *Proceedings of ICAPS*, 2004.

[2] P. Gorniak and I. Davis. Squadsmart: Hierarchical planning and coordinated plan execution for squads of characters. In *Proceedings of AIIDE*, 2007.

[3] J. Kim, I.-C. Moon, and T. G. Kim. New insight into doctrine via simulation interoperation of heterogeneous levels of models in battle experimentation. *Simulation*, 2012.

[4] S. A. Miller, Z. A. Harris, and E. K. P. Chong. A pomdp framework for coordinated guidance of autonomous uavs for multitarget tracking. *EURASIP J. Adv. Signal Process*, 2009.

[5] I.-C. Moon and K. M. Carley. Modeling and simulating terrorist networks in social and geospatial dimensions. *Intelligent Systems, IEEE*, 2007.

[6] I.-C. Moon, M. Schneider, and K. M. Carley. Evolution of player skill in the america's army game. *Simulation*, 2006.

[7] P. E. Nielsen. SOAR/IFOR: intelligent agents for air simulation and control. In *Simulation Conference Proceedings, 1995. Winter*, 1995.

[8] E. J. Sondik. *The optimal control of partially observable Markov processes.* 1971.

[9] B. Zeigler, H. Praehofer, and T. G. Kim. *Theory of modeling and simulation: Integrating discrete event and continuous complex dynamic systems.* Academic Press, New York, 2nd edition, 2000.