ORNL/TM-2007/20

OAK RIDGE NATIONAL LABORATORY

MANAGED BY UT-BATTELLE FOR THE DEPARTMENT OF ENERGY

Particle Swarm Social Adaptive Model for Multi-Agent Based Insurgency Warfare Simulation

Date: January, 2007

Prepared by:
Xiaohui Cui
Associate Staff

UT-BATTELLE

ORNL-27 (4-00)

ORNL/TM-2007/20

Computational Science and Engineering Division

PARTICLE SWARM SOCIAL ADAPTIVE MODEL FOR MULTI-AGENT BASED INSURGENCY WARFARE SIMULATION

Xiaohui Cui and Thomas E. Potok

Date Published: August, 2007

Prepared by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37831-6283
managed by
UT-BATTELLE, LLC
for the
U.S. DEPARTMENT OF ENERGY
under contract DE-AC05-00OR22725

CONTENTS

Page

LIST OF FIGURES

ABSTRACT

To better understand insurgent activities and asymmetric warfare, a social adaptive model for modeling multiple insurgent groups attacking multiple military and civilian targets is proposed and investigated. This report presents a pilot study using the particle swarm modeling, a widely used non-linear optimal tool to model the emergence of insurgency campaign. The objective of this research is to apply the particle swarm metaphor as a model of insurgent social adaptation for the dynamically changing environment and to provide insight and understanding of insurgency warfare. Our results show that unified leadership, strategic planning, and effective communication between insurgent groups are not the necessary requirements for insurgents to efficiently attain their objective.

1. INTRODUCTION

Insurgency warfare is dynamic, adaptive and non-linear warfare. The study of insurgency warfare can be traced back to the Chinese strategist Sun Tzu's publications in 400BC [9]. In the last century, relatively modern studies about insurgency warfare have provided general insights and practical guidance into the perspective of insurgents and counter-insurgents [5, 8]. The U.S. military, particularly the Army, has a long history of counter insurgent activity. Until the past decade though, this has not been an area of focus for the U.S. military. The events happened in Somalia, Iraq, and Afghanistan make the military re-assess the 21st century insurgency and revisit its strategy, operational concepts, organization, and doctrine. The investigation of insurgency in the U.S. has reached it's height since the end of the Cold War. But insurgency remains a significant challenge for the U.S. and other governments because of its two dominant characteristics: protractions and ambiguity. One possible solution is to represent insurgency and counter insurgency (COIN) warfare as a complex adaptive system (CAS) [1, 4]. CAS is a non-linear dynamical system of many interacting agents continuously adapting to a changing environment. Multi-Agent System (MAS) provides an architecture and platform for the implementation of relatively autonomous agents. This greatly contributed to the establishment of the agent-based CAS simulation. Numerous empirical-based multi-agent simulators [11] have been constructed over recent years to model complex dynamic systems in numerous disciplines. Currently, most ongoing work in the insurgency warfare research area is typically concerned with enhancing existing military capabilities for COIN rather than building scientific understanding of the insurgency. In terms of modeling, there does not appear to be any mature and widely used methodology addressing insurgency warfare.

In this paper, we present a modified particle swarm model for simulating the insurgent groups' social interaction and adaptation in a complex insurgency warfare system. Our simulation results indicate that even without a centrally controlled leadership to coordinate the action of each insurgent member, the particle swarm modeled insurgent swarm can emerge a highly coordinated behavior that optimizes their attacking results. This paper is organized as follows: Section 2 incorporates an introduction to the canonical particle swarm optimization algorithm. Section 3 provides related work on the agent based insurgency warfare simulation; Section 4 describes the particle swarm social adaptive model for our agent based insurgency warfare simulation. Section 5 contains the experimental setups and results. Result discussion and conclusion are presented in Section 6.

2. PARTICLE SWARM OPTIMIZATION ALGORITHM

Particle Swarm Optimization (PSO) is an important part of swarm intelligence. It was originally developed by Eberhart and Kennedy in 1995[1], inspired by the social behavior of the bird flock. In the PSO algorithm, birds in a flock are symbolically represented as particles. These particles can be considered as simple agent swarm “flying” through a problem space. A problem space in PSO may have as many dimensions as needed to model the real problem space. Each particle has a location X-vector and a velocity V-vector. A particle’s location in the multi-dimensional problem space represents one solution for the problem. When a particle moves to a new location, a different problem solution is generated. This solution is evaluated by a fitness function that provides a quantitative fitness value of the solution’s utility.

Each particle also has memory to record the “best location” in the problem space that it has experienced so far, and the knowledge of the best location found so far by all the particles of the swarm. The “best location” means the problem solution generated on this location has the best fitness value. Particles of a swarm communicate the best location with each other and adjust their own location and velocity based on this best location. It is the particle’s personal experience combined with its peers’ experience that influences the movement of each particle through a problem space. For every generation, the particle’s new location is computed by adding the particle’s current velocity V-vector to its location X-vector. Mathematically, given a multi-dimensional problem space, the ***i***th particle changes its velocity and location according to the following equations[7, 9]:

$$v_{id} = w*(v_{id} + c_1 * rand_1 * (p_{id} - x_{id}) + c_2 * rand_2 * (p_{gd} - x_{id})) \tag{1a}$$

$$x_{id} = x_{id} + v_{id} \tag{1b}$$

where $\boldsymbol{V_{id}}$ indicates the speed of the particle moving along the dimensions in a problem space; $\boldsymbol{x_{id}}$ is the particle’s current location; p_{id} *(personal best)* is the location of the particle experienced its personal best fitness value; p_{gd} *(global best)* is the location of the particle experienced the highest best fitness value in the whole population; *d* is the number of dimensions of the problem space; $rand_1$,and $rand_2$ are random values in the range of (0,1). c_1 and c_2 are two positive acceleration constants; $\boldsymbol{w}$ is the constriction coefficient[7] and it is computed according to Equation 2a:

$$w = \frac{2}{2 - \varphi - \sqrt{\varphi^2 - 4\varphi}} \tag{2a}$$

$$\varphi = c_1 + c_2, \varphi > 4 \tag{2b}$$

Equation 1a requires each particle to record its current coordinate $\boldsymbol{x_{id}}$, its velocity $\boldsymbol{V_{id}}$, its personal best fitness value location vector P_{id}, and the whole population’s best fitness value location vector P_{gd}. The best fitness value X_i is updated at each generation based on Equation 3, where the symbol *f()* denotes the fitness function; X_i *()* denotes the best fitness values; and *t* denotes the generation step.

$$X_i(t+1) = \begin{cases} X_i(t) & f(P_d(t+1)) \le X_i(t) \\ f(P_d(t+1)) & f(P_d(t+1)) > X_i(t) \end{cases} \tag{3}$$

The P_{id} and P_{gd} and their coordinate fitness values $f(P_{id})$ and $f(P_{gd})$ can be considered as each individual particle’s experience or knowledge and Equation 3 is the particle’s knowledge updating and learning mechanism. In PSO, the knowledge of each particle will not be updated until the particle encounters a new vector location with a higher fitness value than the currently stored value in its memory.

3. RELATED WORK

A relevant multi-agent based model has been built and studied by Epstein [4]. Epstein reported a simple cellular automata (CA) model for simulating civil violence. In this idealized spatial model, a central authority seeks to use police officers to arrest (remove) actively insurgency from the society for a specified jail term to suppress a decentralized rebellion. This model contains three types of agents: the general citizen, the insurgency, and the police officer. All agents possess local vision and can randomly move to a new unoccupied site within its limited vision over a two-dimensional lattice. By using this simple CA simulation, Epstein showed how the complex dynamics resulting from simple assumptions can generate empirically interesting macroscopic regularities that are difficult to analyze using standard modeling approaches. The MANA model [11], an extension of Epstein's model, introduced specific movement strategies that are aimed at correcting the purely random movement of agents. However, the agent interaction and cognition in both simulations are too rigid and simplistic to be psychologically plausible. The behavior of the software agent is strictly rule-based: If a particular condition appears in the agent's environment, the agent can only respond with a particular preprogrammed action. There is no direct communication between agents and each agent does not have any capability of learning from its previous experience. This is clearly a very unrealistic representation of the social world.

A new insurgency warfare model that can provide a better understanding of the insurgent communication and learning activity is needed. The individual particles in the PSO model are capable of both individual learning and social learning through the interactions between particles. We propose a particle swarm social adaptive model to simulate insurgency warfare. This report presents a pilot study of an integration of particle swarm social knowledge adaptation and multi-agent approaches for modeling the collaboration of insurgent groups while attacking multiple military and civilian targets.

4. PARTICLE SWARM SOCIAL ADAPTIVE MODEL FOR INSURGENCY WARFARE SIMULATION

In this agent based insurgency warfare simulation study, we extend the use of PSO on human social model to simulate the interaction behavior between insurgent agents. Although the PSO algorithm has been widely used as a function optimization tool since it was first published in 1995, the initial research target of the PSO was to develop a human social model and the algorithm itself represents an abstract model of human knowledge social adaptation behavior [6, 7]. Researchers from Europe have applied the PSO model to the simulation of the social behavior in animals [7, 8] and the strategic adaptation in organizations [9]. The research of applying PSO models to the evolution of human society behavior and social cognitive modeling is still unavailable in scientific publications. In this study, we use PSO to model the social adaptive behavior in insurgency warfare. In the following section, we will present a detailed description of each component of the PSO model in our agent based insurgency warfare simulation.

4.1 SIMULATION SCENARIO

Different groups of insurgent agents seek efficient attacking methods to strike the dominant power's targets. The insurgent agents do not have any prior-knowledge about the targets. The insurgent agent that attacks the authority's targets will receive a feedback on the results of the current and historic attack strategy.

4.2 AGENTS

Two types of agents are specified in the particle swarm social adaptive model – the insurgency and the target. There can be multiple insurgencies and targets in the simulation. The insurgent agent can be affiliated with different groups. The objective of the insurgent agents is to locate and attack the highest valuable targets. In contrast, the targets seek to avoid being detected and to increase protection to reduce their loss from insurgent attacks. All agents behave, act, and react in accordance with the environment they have detected.

4.3 INSURGENT INFORMATION EXCHANGE RULE

Insurgents belonging to the same group can exchange information without any restriction. But the information exchanged between different groups will be delayed for a pre-defined number of time-steps and some noise will be added to the value of the information to reduce the information's accuracy.

4.4 INSURGENT AGENT STRATEGIC SEARCHING RULE

The PSO algorithm is used to control the insurgent strategic searching. Under the particle swarm metaphor, each insurgent particle is assumed to move through an attack strategic searching space to search for a functional optimum. Each insurgent particle has two associated properties, a current position ***x*** and a velocity ***v***. Each particle has a memory of its best location where the biggest lose to the authority targets was caused so far (***pbest***) in the attack strategic searching space. Each particle also knows the global best location found by all other particles that belong to the same insurgent group (***gbest***). When the delayed and noisy ***gbest*** value from other groups arrives, the ***gbest*** value from other groups will replace the ***gbest*** value within the group. At each step of the algorithm, an insurgent particle moves from its current position to a new location based on a velocity vector. The velocity vector is influenced by both the particle's previous velocity, its current location and its ***pbest*** and ***gbest*** value. Therefore, at each step, the size and direction of each particle's movement is a function of its own history (experience) and the social influence of its peers.

4.5 TARGET DYNAMICAL ADAPTIVE RULE

Targets randomly move in the environment. When they are attacked, they will gradually increase their protection level to reduce the loss that the insurgent can generate after each attack and to decrease the detectable distance to make itself more difficult to be detected. Once a target has not been attacked for several time-steps, this target will change its protection level to its original value.

4.6 MEMORY UPDATE RULE

Based on the target dynamical adaptive rule, targets can randomly move in the environment and the fitness value of each target in the environment may change over time after insurgent attacks. The location with the highest target fitness value found by a specific insurgent particle will not have the highest fitness value after the targets being attacked for several times. The dynamic change of the highest target fitness value location requires the particle to renew its memory whenever the environmental status does not match its memorized knowledge. However, the traditional PSO lacks an update mechanism to renew the particles' memory when the environment changes. If insurgent particle uses the traditional PSO for directing its attack strategy search, the PSO searching algorithm can cause the particle to continue using the obsolete knowledge to direct its search, which inhibits the particle from following the path of the current optimal solution. As a result, the particle can be easily trapped in the region of the former optimal solution. Therefore, a memory update mechanism [3] is used to renew insurgent particles' memory when it is necessary. In this mechanism, a new notion, a evaporation constant ***T***, is introduced. ***T*** has a value between 0 and 1. The personal fitness value and global fitness value stored in each particle's memory will gradually evaporate (decrease) at the rate of the evaporation constant over time. The update process is formulated as:

$$X_i(t+1) = \begin{cases} X_i(t) & f(P_d(t+1)) \le X_i(t)*T \\ f(P_d(t+1)) & f(P_d(t+1)) > X_i(t)*T \end{cases} \tag{4}$$

In Equation 4, after the value of fitness evaporates for a period, the fitness value, ***X-fitness***, of the current location may be higher than the evaporated fitness values and will replace the old fitness value. Although all particles have the same evaporation constant T, each particle's update frequency may not be same. Depending on the particle's current stored best fitness value ***f(P)*** and the current fitness value ***f(X)*** the particle acquired, the particle will update its best fitness value more frequently by using its current fitness value when the ***f(P)*** is lower and the ***f(X)*** is higher. However, when the ***f(P)*** is higher and the ***f(X)*** is lower in a changing environment indicates the particle's current location is farther away from the current optimal solution compared to the distance between the optimal solution and the best fitness value's position stored in the particle's memory. In this situation, the best fitness value will be kept in the particle's memory till the best fitness value has become too obsolete after several generations. The fitness value update equation enables each particle to self-adapt to the changing environment.

5. EXPERIMENTAL SETTINGS AND RESULTS

Simulations are carried out in the Netlogo [10] agent modeling environment. The insurgent particles use Equation 4 to update their best fitness value and the evaporation constant ***T*** is set as ***exp (-1)***. There are 300 insurgent particles randomly distributed in the environment that consists of a ***100x100*** rectangular grid. An example of the initial environment is shown in Fig. 1. In this initial environment, twenty insurgent groups are simulated and each group has 15 insurgents. Each dot represents an insurgent and different colors are used to help to identify different insurgent groups. The white circuits represent different insurgent attacking targets. The brighter the white color of the circuit, the higher the loss will be when attacked by insurgency. There are eight targets initially deployed in the example environment. When the distance between insurgent and a target is smaller than a predefined value ***k***, the insurgent will attack the target. The loss can be calculated by using the following standard normal distribution equation:

$$f(x) = \begin{cases} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} & x >= k \\ 0 & x < k \end{cases} \tag{5}$$

where ***x*** is the distance to the target and ***k*** is a constant value. The nearer the insurgent to the target, the higher the loss is.

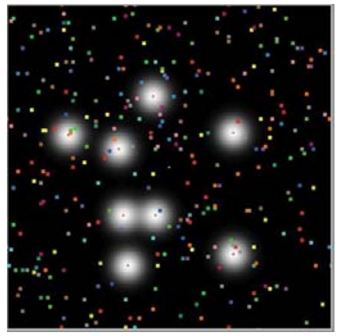
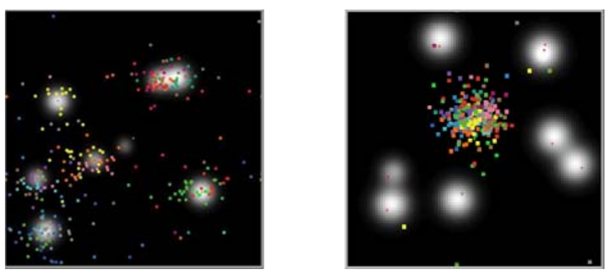


Fig. 1: The initial environment

The movement of each insurgent particle follows Equation 2a and Equation 2b, in which c_1 and c_2 are set to 1.49 and V_{max} is set to 5. The w value is set to 0.72. All targets randomly move in the environment at a speed equal to 2. The delayed time-steps for information exchange between groups is 20 time-steps and there is a 20 percent possibility that the information, including the location of the best fitness value and the fitness value itself, is incorrect. Different numbers of insurgent groups are simulated in this study. To evaluate the performance of the insurgent groups in the simulation, average loss and total loss caused by insurgent attacks at each time step are recorded separately. The simulation results are presented in Fig. 2 - 6.

The trajectories of insurgent particles with updated memory rule (Fig. 2a) are compared with these without updated memory rule (Fig. 2b) and the results are shown in Fig. 3.



(a) (b)

Fig. 2: Trajectories of insurgent particles (a) with update memory rule, (b) without update memory rule.

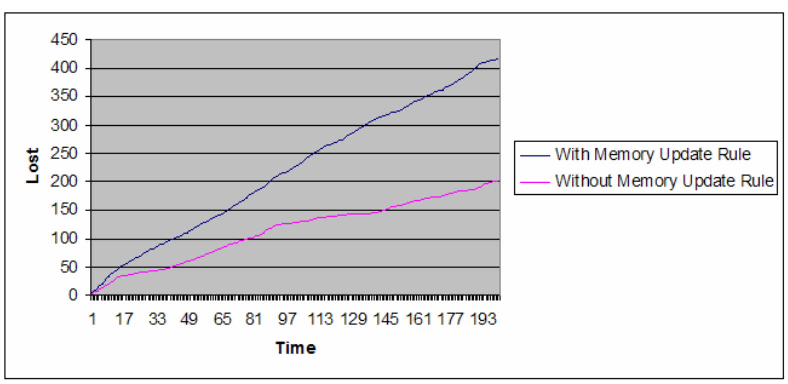
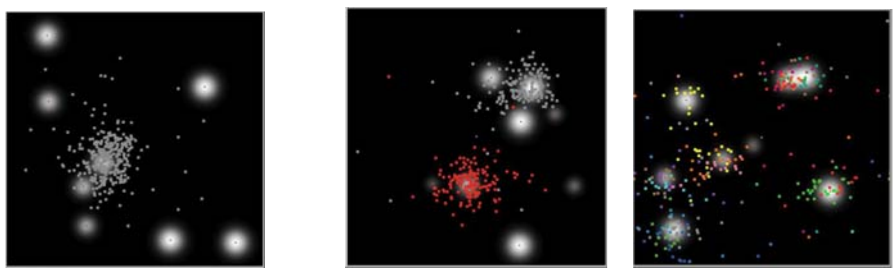


Fig. 3: The average loss caused by insurgents with memory update rule and without memory update rule.

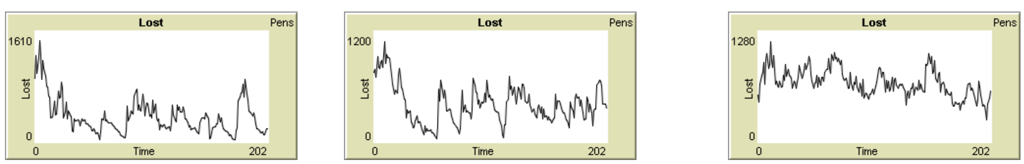
Fig. 3 illustrates that, in the dynamic environment, the insurgent group modeled with PSO model without the memory update rule fails to track the randomly moving optimal solution. As shown in Fig. 2b, all particles are trapped at the center of the environment. However, as shown in Fig. 2a, the insurgent particles that use the memory update rule can, in real time, track the dynamically moving target and surround themselves at the vicinity of the targets. This can help insurgent groups generate higher total loss on the targets.

To investigate whether lacking unified leadership, planning, and effective communication among insurgent groups can inhibit or facilitate insurgents obtaining their goals, the performance of the insurgent particles for one insurgent group with 300 insurgents (Fig. 4a), two insurgent groups with 150 insurgents in each group (Fig. 4b), and twenty insurgent groups with 15 insurgent in each group (Fig. 4c) are compared. The results are shown in Fig. 5.



(a) (b) (c)

Fig. 4. Trajectories of insurgent particles for (a) one group, 300 insurgents, (b) two groups, 150 insurgents per group, (c) twenty groups, 15 insurgents per group



(a) (b) (c)

Fig. 5. Total loss at each time-step for (a) one group, 300 insurgents, (b) two groups, 150 insurgents per group, (c) twenty groups, 15 insurgents per group

The average loss caused by insurgents with different group numbers is shown in Fig. 6. With time increases, the average loss increases in all three situations. As shown in Fig. 6, the simulation of twenty insurgent groups can cause more loss of targets than both the one group and the two groups do, although there is no unified leadership, planning or effective communication among these twenty insurgent groups in the simulation compared to the one group insurgent simulation.

Fig. 6. The average loss caused by insurgents with different group numbers.

6. DISCUSSION AND CONCLUSION

In 1962, President John Kennedy advised the West Point graduates that they would have to deal with "…another type of war, new in its intensity, ancient in its origins—war by guerrillas, subversives, insurgents, assassins; war by ambush instead of by combat; by infiltration, instead of aggression, seeking victory by eroding and exhausting the enemy instead of engaging him....". After more than 40 years, this kind of warfare is still the most significant military tactic being used against American forces around the world. Discerning how insurgent groups interacting, learning and how emergent behaviors emerging from aggregate interactions in a dynamic environment is crucial for understanding insurgency.

In this paper, a modified PSO model is developed to simulate the complex interactions between the insurgent groups and targets and to analyze how an un-organized, un-planned insurgent riot can reach their effect-based operation in a highly dynamic environment. Our primary aim is to demonstrate how individual insurgent violence on dominant power targets can produce effect based operations, which usually requires highly organized and professional planning. We construct a novel agent based simulation model to examine the impact of different attack scenarios that the insurgent groups may conduct. The objective of this research is not to develop a tool for optimizing an optimizing attack strategy that can cause highest loss to authority, but to apply the particle swarm metaphor as a model of insurgent social adaptation for the dynamically changing environment and provide an insight and scientific understanding of the insurgency warfare. Results from our simulation have shown that lack of unified leadership, planning, and effective communication are not the necessary requirements for insurgents to attain their objective.

7. REFERENCES

1. Arquilla, J. and Ronfeldt, D., "Swarming and the Future of Conflict". DB-311, RAND, Santa Monica, CA, 2000.
2. Carlisle, A. and Dozier, G., "An Off-The-Shelf PSO", Proceedings of the 2001 Workshop on Particle Swarm Optimization, 2001, Indianapolis, IN, USA, 2001.
3. Cui X., Hardin T., Ragade R. K., Potok T. E., and Elmaghraby A. S., "Tracking non-Stationary Optimal Solution by Particle Swarm Optimizer", In Proceedings of the 6th ACIS

International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, May, 2005, Baltimore, Maryland, USA, 2005.

4. Epstein, Joshua M., “Modeling Civil Violence: An Agent-Based Computational Approach”, Proceedings of the National Academy of Sciences, Vol. 99, pp. 7243-7250, 2000.

5. Guevara, Ernesto. Guerrilla Warfare. Lincoln: University of Nebraska Press, 2001.

6. Kennedy, J. and Eberhart, R. “Particle Swarm Optimization”, IEEE International Conference on Neural Networks, Perth, Australia, Vol. IV: pp. 1942-1948, 2000.

7. Kennedy, J., “The Particle Swarm: Social Adaptation of Knowledge”, IEEE International Conference on Evolutionary Computation, Indianapolis, Indiana, USA, pp. 303-308, 1997.

8. Mao Tse-tung. “On Guerrilla Warfare Champaign”, Illinois: University of Illinois Press, 2000.

9. Sun Tzu, “The Art of War” (tr. S.B Griffith), Oxford University Press, Oxford, 1963.

10. Wilensky, U., “NetLogo”. Evanston, IL. Center for Connected Learning and Computer Based Modeling, Northwestern University, 1999

11. S. Y. Yiu, A. Gill, and P. Shi, “Investigating Strategies for Managing Civil Violence Using the MANA Agent Based Distillation,” Land Warfare Conference, Brisbane, Australia, 2002.

INTERNAL DISTRIBUTION

1. T. E. Potok
2. J. Treadwell
3. ORNL Office of
   Technical Information
   and Classification