Particle Swarm Social Model for Group Social Learning in Adaptive Environment

Xiaohui Cui[1], Laura L. Pullum[2], Jim Treadwell[1], Robert M. Patton[1], and Thomas E. Potok[1]

[1]Computational Sciences and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831
[2]Lockheed Martin Corporation, St. Paul, MN 55164-0525

Abstract This report presents a study of integrating particle swarm algorithm, social knowledge adaptation and multi-agent approaches for modeling the social learning of self-organized groups and their collective searching behavior in an adaptive environment. The objective of this research is to apply the particle swarm metaphor as a model of social learning for a dynamic environment. The research provides a platform for understanding and insights into knowledge discovery and strategic search in human self-organized social groups, such as human communities.

1 Introduction

The notion of social learning has been used with many different meanings to refer to processes of learning and change of individuals and social systems. In this research, social learning refers to the process in which agents learn new knowledge and increase their capability by interacting with other agents directly or indirectly. Research of some social species indicates that these social species have a kind of social learning capacity to use knowledge provided by other group members to help the whole group quickly respond and adapt to a dynamic environment. Swarm Intelligence is the research field that attempts to design computational algorithms or distributed problem-solving devices inspired by the collective social learning behaviors of these social species. Particle swarm algorithm [10] is a type of Swarm Intelligence and was originally developed by Eberhart and Kennedy in 1995. Since 2004, researchers have successfully applied the particle swarm model in the simulation of the social behavior of animals [7] and strategic adaptation in organizations [2]. However, in terms of self-organized group social learning and collective strategy searching behavior for dynamically changing environments, there does not appear to be any mature or widely used methodology.

In this research, a modified particle swarm algorithm model, PArticle Swarm Social (PASS) model, is used to model the self-organized group's social learning

and collective searching behavior in a dynamically changing environment. Different from the randomly changing environment model used in many research efforts, a new adaptive environment model, which adaptively reacts to the agent's collective searching behaviors, is proposed. An agent based simulation is implemented for investigating the factors that affect the global performance of the whole social community through social learning. The objective of this research is to apply the particle swarm metaphor as a model of human social group social learning for the adaptive environment and to provide insight and understanding of social group knowledge discovery and strategic search in a changing environment.

This paper is organized as follows: Section 2 provides an introduction to the canonical particle swarm optimization algorithm. Section 3 describes the PASS model, which covers the individual social learning behavior, the adaptive environment and a social network model for social learning behavior. Experiment results will be discussed in section 4. Section 5 describes a verification and validation approach for social network models. The conclusion is presented in Section 6.

2 Particle Swarm Algorithm

The particle swarm algorithm was inspired by the social behavior of bird flocks and social interactions of human society [10]. In the particle swarm algorithm, birds in a flock or individuals in human society are symbolically represented as particles. These particles can be considered as simple agents "flying" through a high dimensional problem space searching for a high fitness value solution or strategy. It is the particle's personal experience combined with its neighbors' experience that influences the movement of each particle through a problem space. Mathematically, given a multi-dimensional problem space, the i_{th} particle changes its velocity and location according to the following equations:

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

$$x_{id} = x_{id} + v_{id} \tag{2}$$

where, $p_i d$ is the location where the particle experiences the best fitness value; $p_g d$ is the location of the highest best fitness value current found in the whole population; $x_i d$ is the particle's current location; c_1 and c_2 are two positive constants; d is the number of dimensions of the problem space; $rand_1$ and $rand_2$ are random values in the range of (0,1). w is called the constriction coefficient [8]. Each particle will update the best fitness values $p_i d$ and $p_g d$at each generation based on Eq.3, where the symbol f denotes the fitness function; $P_i(t)$ denotes the best fitness coordination; and t denotes the generation step.

$$f(P_i(t+1)) = \begin{cases} f(P_i(t)), & \text{if } f(X_i(t+1)) \leq f(P_i(t)) \\ f(X_i(t+1)), & \text{if } f(X_i(t+1)) > f(P_i(t)) \end{cases} \tag{3}$$

The P_{id} and the coordinate fitness values $f(P_{id})$ can be considered as each individual particle's experience or knowledge. The P_{gd} and the coordinate fitness values $f(P_{gd})$ can be considered as the best knowledge that an individual can acquire from its neighbors through interaction. The social learning behavior in the particle swarm algorithm model is mathematically represented as the combination of an individual particle's experience and the best experience it acquired from neighbors for generating the new moving action. In the following section, we propose a modified particle swarm social (PASS) model for modeling the self-organized group social learning.

3 Particle Swarm Social Model

The PASS model describes individuals that are affiliated with different groups, seeking the highest fitness (profit) value solution or strategy in a changing environment. Individuals use the information provided by other individuals to enhance their capability in finding the highest fitness value solutions. The solution landscape will dynamically change as the group individuals search for the highest profit strategy configuration. This demands that the groups not only find a highly profitable solution in a short time, but also track the trajectory of the profitable solution in the dynamic environment. The group members do not have any prior knowledge about the profit landscape. Following two sections will describe the two elements of the PASS model: the adaptive environment and the individual social learning behavior.

3.1 Adaptive Environment

In the PASS model, the change patterns of the environment will be influenced by the collective behaviors of the social groups when these collective behaviors are effective enough to alter the environment. We define this kind of environment as an adaptive environment. To simulate the movement of the solutions, a test function, DF1, proposed by Morrison and De Jong [12], is used to construct the solution fitness value landscape. DF1 function has been widely used as the generator of dynamic test environments [1, 13]. For a two dimensional space, the fitness value evaluation function in DF1 is defined in Eq. 4. Where N denotes the number of peaks in the environment. The (x_i, y_i) represents each cone's location. R_i and H_i represent the cone's height and slope.The movement of the problem solutions and the dynamic change of the fitness value of different solutions are simulated with the movement of the cones and the change of the height of the cone-shaped peaks. It is controlled through the logic function eq. 5. Where A is a constant and Y_i is the value at the time-step i. The Y value produced on each time-step will be used to control the changing step sizes of the dynamic environment.

$$f(X,Y) = MAX[H_i - R_i \times \sqrt{(X-x_i)^2 + (Y-y_i)^2}];\ (i = 1, ...N) \tag{4}$$

$$Y_i = A \times Y_{i-1} \times (1 - Y_{i-1}) \tag{5}$$

In real-world applications, the evaluated fitness value cannot always be calculated precisely. Most of the time, the fitness value will be polluted by some degree of noise. To simulate this kind of noise pollution in the fitness evaluation, a noise polluted fitness value function [13] can be represented in eq.6. where η illustrate the noise and is a Gaussian distributed random variable with zero mean and variance σ^2. Another dynamic mechanism of the fitness landscape is the fitness value will gradually decrease with an increasing number of the searching group members that adopt similar solution. The eq. 7 represents this fitness value decrease.Where f is the landscape fitness value of strategic configuration (x,y) at the iteration i. N denotes the number of group member that adopts similar strategic configurations.

$$f^n(x) = f(x) \times (1+\eta); \;\; \eta \sim N(0,\sigma^2) \tag{6}$$

$$f_i(x,y) = f_{i-1}(x,y) \times (\frac{1}{e^{(N-1)}}) \tag{7}$$

3.2 The individual social learning behavior

Social learning behavior occurs when individual can observe or interact with other individuals. An individual will combine his individual experience and the information provided by other experienced individuals to improve its search capability. The particle swarm algorithm is used to control the individual's social learning behavior in the dynamical profit space. According to the eq.3, a particle's knowledge will not be updated until the particle encounters a new vector location with a higher fitness value than the value currently stored in its memory. However, in the dynamic environment, the fitness value of each point in the profit landscape may change over time. The problem solution with the highest fitness value ever found by a specific particle may not have the highest fitness value after several iterations. It requires the particle to learn new knowledge whenever the environment changes. However, the traditional particle swarm algorithm lacks a knowledge updating mechanism to monitor the change of the environment and renew the particles' memory when the environment has changed. As a result, the particle continually uses outdated experience/knowledge to direct its search, which inhibits the particle from following the movement of the current optimal solution and eventually, causes the particle to be easily trapped in the region of the former optimal solution. In the PASS model, a distributed adaptive particle swarm algorithm approach [9] is used to enable each particle to automatically detect change in the environment and use social learning to update its knowledge. Each particle will compare the fitness value of its current location with that of its previous location. If the current fitness value doesn't have any improvement compared to the previous value, the particle will use Eq.8 for the fitness value update. Eq.8 is slightly different than the traditional fitness value update

function provided in Eq.3.

$$f(P_i(t+1)) = \begin{cases} f(p_i(t)) \times \rho, \text{ if } f(X_i(t+1)) \leq f(P_i(t)) \times \rho \\ f(X_i(t+1)), \text{ if } f(X_i(t+1)) > f(P_i(t)) \times \rho \end{cases} \tag{8}$$

In Eq. 8, a new notion, the evaporation constant ρ, is introduced. ρ has a value between 0 and 1. The personal fitness value that is stored in each particle's memory and the global fitness value of the particle swarm will gradually evaporate (decrease) at the rate of the evaporation constant over time. If the particle continuously fails to improve its current fitness value by using its previous individual and social experience, the particle's personal best fitness value $f(p_{id})$ as well as the global best fitness value $f(p_{gd})$ will gradually decrease. Eventually, the$f(p_{id})$ and $f(p_{gd})$ value will be lower than the fitness value of the particle's current location and the best fitness value will be replaced. Although all particles have the same evaporation constant ρ, each particle's updating frequency may not be the same. The updating frequency depends on the particle's previous best fitness value and the current fitness value $f(X)$ that the particle acquired. The particle will update its best fitness value more frequently when the previous best fitness value is lower and the $f(X)$ is higher. Usually the new environment (after changing) is closely related to the previous environment from which it evolved. It would be beneficial to use the existing knowledge/experience about the previous landscape space to help particle searching for the new optimal. The Eq.8 enables each particle to self-adapt to the changing environment.

4 Social Learning Simulation Experiment and Results

The implementations of the PASS model and the adaptive environment simulations are carried out in the NetLogo agent modeling environment [15]. Each agent in the NetLogo environment represents one particle in the model. Initially, there are 400 agents randomly distributed in an environment that consists of a 100x100 rectangular 2D grid. The grid represents all the possible strategic configurations or solutions the agents may adopt for their profit. A dynamic profit landscape is generated as discussed in section 3.1 and mirrored on the 2D grid. The initial status of the 2D grid is shown in Fig.1(a). Eight white circuits represent the fitness (profit) values of solutions. The brighter the white circuit, the higher the fitness (profit) value is. These white circuits will dynamically move in the grid base on the Eq. 4 and the fitness values (brightness of the circuits) are dynamically changed base on Eq. 5 and Eq. 6. The agents are represented as the color dots in the grid. Different colors indicate different groups. The social learning of each individual is represented as the highest fitness value and location broadcast within the group. The searching behavior for finding highly profitable solution is represented as the movement of agent in the 2D grid. The movement of each agent is controlled by Eq. 1 and Eq. 2, in which c_1 and c_2 are set to 1.49, V_{max} is set to 5 and the w value is set to 0.72 as recommended in

the canonical particle swarm algorithm [12]. It is assumed that agents 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 pollute the value of the information to reduce the information's accuracy. The delayed time-step for information exchange between agent groups is pre-set as 20 time-steps. There is a 20% possibility that the information, including the location of the best fitness value and the fitness value itself, is incorrect.

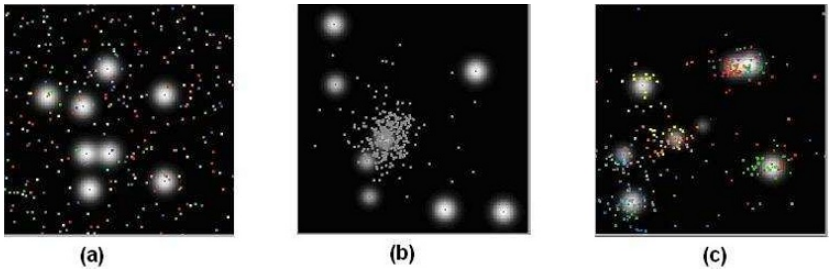
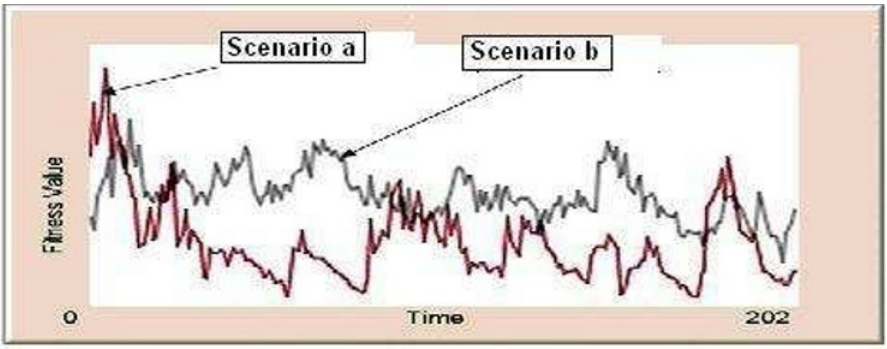


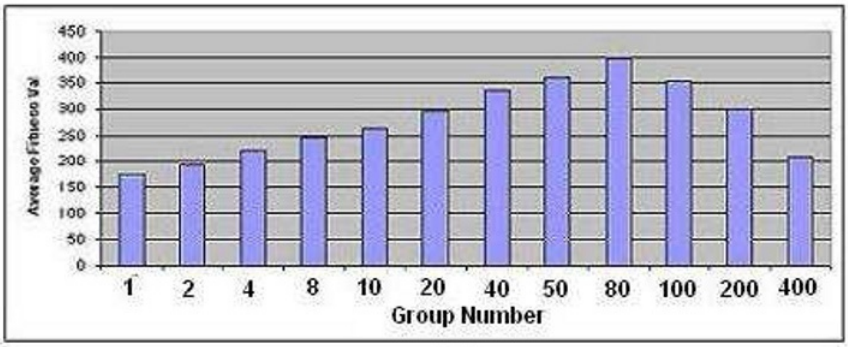
Fig. 1 The distribution of (a) initial environment, (b) scenario *a*: 1 group, 400 agents, (c) scenario *b*: 20 groups, 20 agents per group

In this experiment, we first investigated the change of a particle's social learning performance when the social group structure changed from a single group to multiple groups. The performance evaluation can be generated via computing the average fitness value of all individuals generated in the whole search period. Two different agent group structure scenarios, scenario a and scenario b, are simulated in this study. In scenario a, 400 agents belong to one single group. In scenario b, the 400 agents are evenly distributed into 20 different groups with 20 agents in each group. Each simulation was run for 200 iterations. The final agent distribution maps are presented in Fig 1(b) and (c). As shown in Fig 1(b), for scenario a, all agents belong to the same group. These agents can freely exchange information about their searching performance. Every agent wants to adopt the problem solution that can generate the highest fitness value. This will cause all agents to swarm around the highest fitness value peak in the landscape. However, because of the dynamic adaptation character of the landscape, the fitness value of the problem solutions around the highest peak will gradually reduce when the number of agents around it increases. For scenario b, as shown in Fig 1(c), limited and noised communication between agent groups causes some agents not to receive the newest information about the best solution that other agents have found. Consequently, agents are distributed relatively evenly around different solution fitness peaks.

The searching performance of these two group scenarios is shown in Fig 2(a), illustrating the average fitness value vs. simulation time step. Initially, scenario a has a higher fitness value than the scenario b, because in scenario a, with the help of social learning, all agents can quickly aggregate around the highest peak in the land-



(a)



(b)

Fig. 2 The comparison of the average fitness values of (a) each simulation iteration for group scenario *a* and *b* (b) whole simulation for different agent group scenarios

scape. However, the fitness value in the landscape will adaptively change according to Eq.7. The congregation of the agents around the highest fitness value will cause a quick decrease of the fitness value of the nearby landscape and eventually cause the sum of the fitness value to quickly reduce. As shown in Fig. 2(a), the profit of scenario a reduces quickly from the peak and remains low. For scenario b, because of the delay and inaccuracy of the information between groups, the agents are evenly distributed around all fitness peaks. This distribution makes the fitness value of the nearby landscape not decrease as quickly as scenario a and maintains a higher group fitness value than scenario a in nearly the whole simulation.To discover the social network architecture that can generate the highest performance, we tested the performance of different group structures that varied from fully connected social network, in which all individuals belong to a single group, to no connection social network, in which no individuals belong to same group. The searching performance is recorded as the average fitness value over the whole simulation. The performance chart is shown in Fig 2(b). According to Fig.2(b), the performance gradually in-

creases when the agents are divided into large numbers of groups. The performance reveals the highest value when there are 80 agent groups and 5 agents in each group, then the performance gradually reduces.

5 Verification and Validation

Studies of the effectiveness of evolutionary algorithms (EA) show that verification requires rigorous and standardized test problems or benchmarks. Branke [4, 6] provides a detailed survey of test problems and benchmark functions found in literature. We have a dynamic, continuous environment, characterized by complexity, a requirement for the solution to adapt to changes over time, and that uses floating point representations of domain characteristics. For such continuous environments, some of the most commonly used benchmark problems are the moving parabola problem, the moving peaks benchmark function [5], and DF1 [12]. Test environments such as DEFEAT [11] can also be used in the verification process for EA in dynamic environments. Our use of DF1 is described in Section 3.1. Another approach uses formal methods to specify and verify swarm-based missions and has been used in verifying the NASA ANTS (Autonomous Nano Technology Swarm) mission [14]. Applicability of this approach to our problem domain is to be determined.

There are several ways to validate agent-based systems and their choice depends on access to the actual phenomenon investigated and on model complexity. They are [16]:

- Compare agent-based simulation/system output with real phenomenon. This is a straightforward comparison, with the difficulty being access to complete real data on the relevant aspects of the phenomenon under study.
- Compare agent-based simulation/system results with mathematical model results. This approach has the disadvantage of requiring construction of the mathematical models which may be difficult to formulate for a complex system.
- Docking with other simulations of the same phenomenon. Docking is the process of aligning two dissimilar models to address the same question or problem, to investigate their similarities and their differences, and to gain new understanding of the issue being investigated [3].

The PASS model research provides a platform for understanding and insights in human self-organized human social groups. With respect to comparison with real phenomenon, it is common knowledge that the collection of real self-organized human community data is difficult. The available data is also questionable because of the potential bias existing when they are collected. Table 1 below lists preferred and optional validation data sources and types. The preferred validation data is real-time data gathered in the domain of interest. If one does not have access to real-time data, then options include use of historical data in the domain of interest, historical data in a parallel domain, or, least favorable, generated data. By parallel domain, we mean

one in which the characteristics of the substitute domain parallel, or resemble, those of the domain and phenomenon under study.

Table 1 Preferred and Optional Validation Data Sources and Types

Preferred	Option
Gathered	Generated
Domain of Interest	Parallel Domain
Real-time	Historical

6 Discussion and Conclusion

Most reported searching behavior models only discuss the scenarios in a static environment or a randomly changing environment. The performance evaluation of various approaches is mainly based on how fast an approach can find the optimal point in the benchmark problems. However, the real social world is rarely static and its changes are not random. Most of time, the changes are influenced by the collective actions of the social groups in the world. At the same time, these influenced changes will impact the social groups' actions and structure. In this paper, a modified particle swarm social learning model is developed to simulate the complex interactions and the collective searching of the self-organized groups in an adaptive environment. We constructed a novel agent based simulation to examine the social learning and collective searching behavior of different social group scenarios. Results from the simulation have shown that effective communication is not a necessary requirement for self organized groups to attain higher profit in an adaptive environment.Part of the hesitance to accept multi-agent and swarm-based modeling and simulation results rests in their perceived lack of robustness. The next step in this research will be focused on the model verification and validation.

Acknowledgements Prepared by Oak Ridge National Laboratory, P.O. Box 2008, Oak Ridge, Tennessee 37831-6285, managed by UT-Battelle, LLC, for the U.S. Department of Energy under contract DE-AC05-00OR22725; and by Lockheed Martin, partially funded by internal Lockheed research funds.

References

1. Angeline P. J. (1997): Tracking extrema in dynamic environments. In Angeline, Reynolds, McDonnell and Eberhart (Eds.), Proc. of the 6th Int. Conf. on Evolutionary Programming, LNCS, Vol. 1213 , Springer, 335–345

2. Anthony B., Arlindo S., Tiago S.(2004), MichaelO. N., Robin M. , and Ernesto C.: A Particle Swarm Model of Organizational Adaptation. In Genetic and Evolutionary Computation (GECCO), Seattle, WA, USA 12–23
3. Burton R., (1998): Simulating Organizations: Computational Models of Institutions and Groups, chapter Aligning Simulation Models: A Case Study and Results. AAAI/MIT Press, Cambridge, Massachusetts.
4. Branke, J., (1999): "Evolutionary Algorithms for Dynamic Optimization Problems - A Survey", Technical Report 387, Institute AIFB, University of Karlsruhe .
5. Branke, I., (1999): "Memory Enhanced Evolutionary Algorithms for Changing Optimization Problems", Proceedings of Congress on Evolutionary Computation CEC-99, pp. 1875-1882, IEEE.
6. Branke, J., (2002): Evolutionary Optimization in Dynamic Environments, Kluwer Academic.
7. Cecilia D. C., Riccardo P., and Paolo D. C., (2006): Modelling Group-Foraging Behaviour with Particle Swarms. Lecture Notes in Computer Science, vol. 4193/2006, 661–670
8. Clerc M. and Kennedy J., (2002): The particle swarm-explosion, stability, and convergence in a multidimensional complex space. IEEE Transactions on Evolutionary Computation, vol. 6 58–73
9. Cui X., Hardin C. T., Ragade R. K., Potok T. E., and Elmaghraby A. S., (2005): Tracking non-stationary optimal solution by particle swarm optimizer. in Proceedings of Software Engineering, Artificial Intelligence, Networking and Parallel/ Distributed Computing, Towson, MD, USA 133–138
10. Eberhart R. and Kennedy J., (1995): A new optimizer using particle swarm theory. In Proceedings of the Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan 39–43
11. Etaner-Uyar, Sima A., and Turgut U. H., (2004): "An Event-Driven Test Framework for Evolutionary Algorithms in Dynamic Environments," IEEE, pp. 2265-2272 .
12. Morrison R. W. and DeJong K. A., (1999): A test problem generator for non-stationary environments. In Proceedings of the 1999 Congress on Evolutionary Computation, Washington, DC, USA 2047-2053
13. Parsopoulos K. E. and Vrahatis M. N., (2002): Recent approaches to global optimization problems through particle swarm optimization. Natural Computing 1 235–306
14. Rouff C. A., Truszkowski W. F., Hinchey M. G., Rash J. L., (2004): "Verification of emergent behaviors in swarm based systems", Proc. 11th IEEE International Conference on Engineering Computer-Based Systems (ECBS), Workshop on Engineering Autonomic Systems (EASe), pp. 443-448. IEEE Computer Society Press, Los Alamitos, CA, Brno, Czech Republic .
15. Tisue S., (2004): NetLogo: A Simple Environment for Modeling Complexity. In International Conference on Complex Systems, Boston, MA
16. Xu J., Gao Y., and Madey G., (2003): "A Docking Experiment: Swarm and Repast for Social Network Modeling," .