Introduction of Acquiring Method for Agents' Actions with Simple Ant Colony Optimization in Multi-agent System

- Rescue Simulation League Team Description anct_resq_2011 -

Hisayuki Sasaoka[1]

[1] Asahikawa National College of Technology

Shunkohdai 2-2, Asahikawa, Hokkaido, 071-8142, JAPAN.

anctsasa@gmail.com

Abstract. This paper has presented acquiring method for agents' actions using Ant Colony Optimization (ACO) in multi-agent system. ACO is one of powerful meta-heuristics algorithms and some researchers have reported the effectiveness of some applications with the algorithm [1-4]. I have developed fire brigade agents using proposed method in RoboCup rescue simulation system. The final goal of my research is an achievement of co-operations for hetero-agent in multi-agent systems. Then this research for implementation for fire brigade agents in my team is the first step of this goal.

Keywords: *Machine Learning, Swarm Intelligence, Ant Colony Optimization.*

1 Introduction

Recently, some researchers have reported the effectiveness of systems installed swarm intelligence algorithms [1]. Especially, Ant Colony Optimization (ACO) and Ant Colony System (ACS) have become a very successful and widely used in some applications [1-4]. Real ants are capable of finding the shortest path from a food source to their nest without using visual cues by exploiting pheromone information. The real ants exploit pheromone to find shortest path between two points. The behavior of the real ants has inspired ACO and ACS. The system based on ACO and ACS are used artificial ants cooperate to the solution of a problem by exchanging information via pheromone. In this paper, I would like to study that Simple-ACO (S-ACO) algorithm is applied to agents of fire brigade agents in my team.

The travelling salesman problem (TSP) has no noise for solving and all of distances between each city are given in advance. Moreover their situations have never changed for each simulation steps. However situations or outer information in environment is always changing in the real world, dynamically. In some cases, we are disable to know cues to solve a problem in advance. In other case, some outer noise gets information erased or interpolation them.

On the other hand, in a situation of RoboCup rescue simulation system, agents need to handle huge amount of information and take actions dynamically. Therefore, a simulation system of RoboCup rescue is a very good test bed for multi-agent research.

2 Our Approach

In our previous studies [5] and [6], our agents in RoboCup soccer simulation league are able to decide a direction of kicking soccer ball using S-ACO. In the studies, we have done some experiments with the soccer agents implemented in our method and the agents using our method have improved the abilities of getting scores in soccer games. From them, we have confirmed the effectiveness of our method. In next subsection, I would like to the basic idea for our method using Simple-ACO.

2.1 Outline of simple-ACO

Initially, agents have decided their actions on randomly. At each construction step, they have taken a probabilistic choice to decide their direction in their moves. The probabilistic is calculated by (1).

$$p_{ij}^{k}(t)=\begin{cases}\dfrac{\left[\tau_{ij}(t)\right]^{\alpha}\left[\eta_{ij}\right]^{\beta}}{\sum_{l\in J_i^k}\left[\tau_{il}(t)\right]^{\alpha}\left[\eta_{il}\right]^{\beta}} & (if\ j\in J_i^k)\\ 0 & (\text{in other case})\end{cases} \quad \text{...(1)}$$

In (1), τ_{ij} means the value of the associated pheromone trail on arc (i, j).A value i means a current position on the simulation field. A value j means a choice to move to the position. η_{ij} is a heuristic value that is available a priori, α and β are two parameters which determine the relative influence of pheromone trail and the heuristic information. The η_{ij} s have to be decided in advance.

The pheromone trails are updated and pheromone evaporation is able to be calculated by (2).

$$\begin{aligned}&\tau_{ij}(t)\leftarrow(1-\rho)\tau_{ij}(t)+\Delta\tau_{ij}^{k}(t)\\ &\Delta\tau_{ij}(t)=\begin{cases}\sum_{k=1}^{m}\dfrac{Q}{L^k(t)} & (if\ (i,j)\in L)\\ 0 & (other)\end{cases}\end{aligned} \quad \cdots(2)$$

ρ means the pheromone evaporation rate, which is between 0 and 1. $\Delta\tau$ is amount of deposited pheromone. Moreover value of Q is scored by results of agents' action.

2.2 Basic idea of our algorithm for acquiring agents' action

We have applied our algorithm to searching actions to a water point of fire-brigade agents in our team. The searching algorithm has two steps. It has shown below:

1. In the case that the agents has no water to extinguish a fire,
   (1-a) in the case that the agent has known a way to a water supply position, it heads along the way.
   (1-b) in the case that the agent has not known a way to a water station, it heads a way in random order.
2. On the other hands, the agent has enough water, it heads for a fire point.

Moreover, the action of updating pheromone has two steps. It has shown below:

1. After the agent is able to get water, it does "say" command to broadcast a point of water supply position.
2. Other agents which do not have water track back.

3 Evaluation for agents based on my method

I have developed experimental agents based on sample agents whose source codes are included Robocup rescue simulator-package file [7]. However the process of development is on the way.

I have run agents for ten times on test map which is also included Robocup rescue simulation simulator-package file. A score of the map is 117.828 points at the start of simulation and the average score is 9.106 points with my agents. Figure 1shows one example of result on the map.

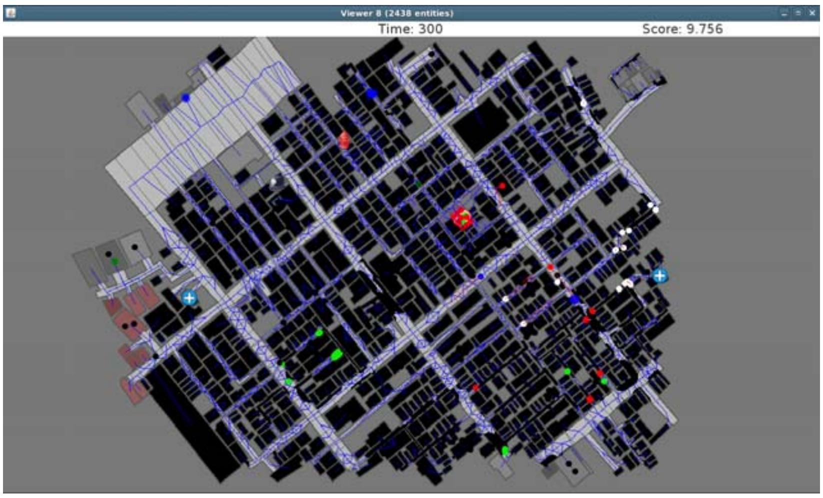

Figure 1. Results of evaluation experiments with my agents on test map.]

Moreover I have run agents on three maps, which are Virtual City 2011, Paris 2011 and Berlin 2011. The scores of these results are shown in Table1. The maps and log files are open to public in the competition's web site[8]. My team "anct2011_resq" have achieved 2nd place finish on the competition.

Table 2. Results of evaluation experiments with my agents.

Names of map	Scores
Virtual City 2011	26.608
Paris 2011	26.221
Berlin 2011	74.546

4 Conclusion

This paper has presented outline of my method and introduced some results with my agents on some maps for Robocup simulation league. In the near future, I would like to consider the values τ and η and to improve agents' abilities.

Acknowledgments. Our programs of agents have developed based on source codes which are included in packages of simulator-package file [7]. This work was supported by Grant-in-Aid for Scientific Research (C) (KAKENHI 23500196).

References

1. Dorigo, M., Stützle, T.: Ant Colony Optimization, The MIT Press, (2004).
2. Hernandez, H., Blum, C., Moore, J. H.: Ant Colony Optimization for Energy-Efficient Broadcasting in Ad-Hoc Networks, in *Proc. 6th International Conference, ANTS 2008,* Brussels, pp.25 –36(2008).
3. D'Acierno, L., Montella, De Lucia, B. F.: A Stochastic Traffic Assignt Algorithm Based Ant Colony Optimisation, in *Proc. 5th International Conference, ANTS 2006,* Brussels, pp.25 –36(2006).
4. Balaprakash, P., Birattari, M., Stützle, T., Dorigo,, M.: Estimation-based ant colony optimization and local search for the probabilistic traveling salesman problem, *Journal of Swarm Intelligence*, vol. 3, Springer, pp. 223–242(2008).
5. Sasaoka, H.:Proposal of Acquiring Method for Agents' Actions using Ant Colony Optimization in Multi-agent System, in *Proc. of the 2010 International Symposium on Intelligent System ,FAN 2010,* (2010).
6. Sasaoka, H.: Approach for Agent Actions' Acquiring Method using Ant Colony Optimization, in *Proc. of Joint 5th International Conference on Soft Computing and Intelligent Systems and 11th International Symposium on Advanced Intelligent Systems, TH-E1-1 (2010).*
7. Robocup Rescue Simulation Project Homepage, http://sourceforge.net/projects/roborescue/
8. Robocup Simulation League Spring Competition 2011 in the 73th National Convention of Information Processing Society Japan, http://rc-oz.sourceforge.jp/pukiwiki/