Multi agent routing to multi targets via ant colony

Conference Paper · March 2010

DOI: 10.1109/ICCAE.2010.5451346 · Source: IEEE Xplore

2 authors:

Mandana Eghbali

Maziar Ahmad Sharbafi
University of Tehran

Multi Agent Routing to Multi Targets Via Ant Colony

Mandana Eghbali
Young Researchers Club, ECE department
Azad University of Qazvin
Qazvin, Iran
mandana.eghbali@gmail.com

Maziar Ahmad Sharbafi
Electrical and Computer Engineering Department
Azad University of Qazvin
Qazvin, Iran
sharbafi@qiau.ac.ir

***Abstract*—Ant colony optimization (ACO) is a cooperative, population-based technique for optimization. Ant algorithms were designed on the base of the behavior of real ant colonies. Real ants can always find the shortest way between the nest and food source, using the environment as the communication tool, named stigmergy. In this paper we focus precisely on the process of finding an optimal path by ant colony optimization algorithm in the multi agent environment. Rescue Simulation, one of the fields of RoboCup competitions, was chosen as the test bed. We have proposed this algorithm to find the best path between basis and destination. Calculating the 'reachablitiy' of a set of targets is the main problem which should be solved with constraints like uncertainty and time limits. Some characteristics of ACO which is utilized as a social algorithm encouraged us to implement it in such problems. Numerical result comparisons with A* algorithm demonstrate the effectiveness and efficiency of this algorithm in Rescue Simulation Environment. The bright background of our team in several world competitions is another support of this work.[1]**

Keywords- Rescue Simulation Environment; path planning; ACO algorithm; A* Algorithm .

I. INTRODUCTION

Path planning is a key step in the programming of mobile agents in Rescue Simulation (RS) environment. And the quality of path influences the efficiency of mobile agents because excess movement causes more temporal and computational expenditures. So designing an efficient path planning algorithm is very critical. Presently, there are two kinds of algorithms for routing in literature. The first are methods like A* and Dijkstra's and their improved versions which are vastly used in path finding [1]. Other approaches are implementing artificial intelligence based methods like Fuzzy Logic [2], Neural Networks [3], and specifically random search techniques such as Genetic Algorithm [4][5] and Ant Colony Optimization (ACO) [6]. The problem is finding the best route from source to destination by selecting the shortest total roads. Existence of many choices in every crossroad may produce a huge search space which results in *curse of dimensionality*. In such conditions suboptimal solutions, obtained by random search methods, are preferred to classic techniques.

Routing will be more complex when you should route to many buildings from one point and it is significant to your decision making. Then the computational cost will be increased and many methods will become unusable. Heuristics which prevents growing computation in spite of increasing the targets donates a precious preference to the routing mechanism. Rescue Simulation environment was selected as a test bed, because it is a complex and time critical multi agent system, in which this problem has a key role on agents' performance. From this point of view ACO has advantages which are gained by the definition of stigmergy determined by the pheromone laid down by ants.

II. OVERVIEW OF ANT COLONY OPTIMIZATION

A. Selecting a Template

Ant Colony Optimization proposed by Marko Dorigo [7], is a new class of natural algorithms inspired by the foraging behavior of natural ant colonies [8]. The development of these algorithms was inspired by the observation of real ant colonies. Ants are social insects and they live in colonies. Ant behavior is governed by the goal of colony survival rather than concentrating on the survival of itself [9]. Ants can find shortest paths between food sources and the nest using tacit communication. At the first steps the search is completely random; so the ants explore the area surrounding their nest in a random manner. While moving, ants leave a chemical pheromone trail on the ground which they can smell it. Searching for their way, they tend to choose paths marked by strong pheromone densities with more probability. As soon as an ant finds a food source, it determines the quantity and the quality of the food and carries some of it back to the nest. During the return trip, the quantity of pheromone that each ant leaves on the ground may depend on the quantity of the source food. The pheromone trails will lead other ants to the food sources. It has been shown in [10] that the indirect communication between the ants via pheromone trails—known as *stigmergy* [11]—enables ants to find shortest path between their nest and food sources. The ACO algorithm has three important steps:

1. Generate solution
2. Update action probability
3. Pheromone update

With these steps ACO can be implemented on every discrete optimization problem. At first all solutions will be distinct and then we should compute probability and update pheromone for each of them. Finally the paths will be chosen

978-1-4244-5586-7/10/$26.00 © 2010 IEEE

based on their probability. The probability is the most important feature helping us to find the shortest and best path.

More probability of choosing the path with more pheromone increases its density during passing them which makes positive feedback named *'local update pheromone'*. The final step performed at the end of each cycle is *'global update pheromone'*. In this process the pheromone of all paths will decrease because of liquid evaporation. The coefficient of evaporation can be tuned by the designer based on problem features, which is 0.01 in our problem.

B. Definition of main variants of ACO

The formulation of ACO will be presented in this section. Every road is limited between two nodes and assumes that the map has N nodes. A node might be a crossroad or the end of a dead end road. In ACO, each road between nodes i and j (r_{ij}), will contain Pheromone showed by τ_{ij} Equation (1) displays the updating rule for Pheromone of the paths after a complete tour of paths was found by an ant.

$$\tau_{ij}\left(t+1\right) = (1-\rho).\tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \qquad (1)$$

Here ρ is the evaporation rate, m is the number of ants and $\Delta\tau_{ij}^{k}$ is the quantity of pheromone per unit length, laid on path r_{ij} by the k^{th} ant:

$$\Delta\tau_{ij}^{k} = \begin{cases} \dfrac{1}{L^{k}} & if\,(i,j)\in T^{k} \\ 0 & otherwise \end{cases} \qquad (2)$$

where L^k is the tour length of the k^{th} ant and the transition probability (P_{ij}^{k}) of the k^{th} ant moving from path (i, j) is given by (3).

$$P_{ij}^{K} = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha}.[\eta_{ij}]^{\beta}}{\sum_{m\in S_k}[\tau_{im}(t)]^{\alpha}.[\eta_{im}]^{\beta}} & if \quad j\in S_k \\ 0 & otherwise \end{cases} \qquad (3)$$

In this equation, S_k is the list of unvisited connected paths which is determined for k^{th} ant. α and β are two parameters to control the importance of the pheromone and the heuristic information η_{ij} respectively. Heuristic is another tuning parameter to insert important goals to the search process. In this paper it is defined by (4), and when d_{ij} is the length of path which connecting points i and j, it means the preference of selecting shorter paths. In our implementation α and β were set to 1.2 and 1 respectively.

$$\eta_{ij} = \frac{1}{d_{ij}} \qquad (4)$$

The most interesting contribution of ACO is the introduction of a *local pheromone update* in addition to the pheromone update performed at the end of the construction process. The local pheromone update is accomplished by all the ants in each cycle. Each ant applies it only to the last Path traversed (5).

$$\tau_{ij} = (1-\varphi).\tau_{ij} + \varphi.\tau_{0} \qquad (5)$$

In this equation $\varphi\in(0,1)$ is the pheromone decay coincident and τ_0 is the initial value of the pheromone. The main target of this step is to vary the exploration process performed by ants, every cycle. This pheromone reduction is done after each action and results in selecting different paths by different ants. The reason is the effect of probability based selection in Ant Colony Optimization. Combining exploration and exploitation is one of the main properties of this searching system which can find suboptimal solutions very fast and the effects of each solution can be utilized in finding other solutions which will be described more in the section IV.

III. WHY RESCUE SIMULATION?

A. Rescue simulation envionment

The Rescue Simulation Project subject is to develop an emergency decision support system by integration of disaster information, prediction, planning, and human interface [12]. To do this, a generic urban disaster simulation environment is created that simulates the urban environment and several disasters that may happen after an earthquake. Besides the unique properties of this simulation environment as a multi-agent system that can be used as a good research problem in this area, the possibility of extending the system by adding necessary disaster modules is the next important feature that must be mentioned. In other words, we have mapped each type of disaster to a requirement and each type of rescue simulation mobile agent to a facility [12].

This problem has led us to a multi-agent system. Based on this statement, each type of mobile agent (Police Force (PF), Ambulance Team (AT) and Fire Brigade (FB) Agents) has a separated view from the disaster environment and a set of requirements defined in this view.

The PF agents just explore the roads and clear the blocked ones. They try to open the roads to fire, collapsed building containing buried civilians, refuge and other important places. Obviously, opening the banned roads is a necessity and the roads priorities are determined by their usage for other agents. Fire Brigades should put out fire buildings to avoid expanding the fire and dying the civilians. Finally, Ambulance Team agents should survive the injured civilians and bring them to the refugee. Less dead and injury means better performance of AT agents.

Such a challenging environment attracted the attention of many researchers recently [13, 14]. Extending scientific methods specifically with Artificial Intelligence backgrounds makes more progresses in this research area [13-15]. In such a challenging condition finding the state of the art methods which are applicable in this environment coping with its constraints is the main bottleneck of designing algorithms [16].

B. Firebrigade agent problems

One of the first steps in decision making of agents in RS is routing. It means when an agent wants to select a target, except the PF agents, firstly, he should check its availability and then find a suitable route to it. Therefore routing should be utilized two times for each option. You can see these places in decision process in Fig.1. One of the events that are simulated in rescue simulation is spread of fire after occurrence of earthquake. If we don't control the fire it can quickly spread to other buildings and may cause serious damage in the city. The main purpose of Fire Brigade agents is extinguishing burning buildings.

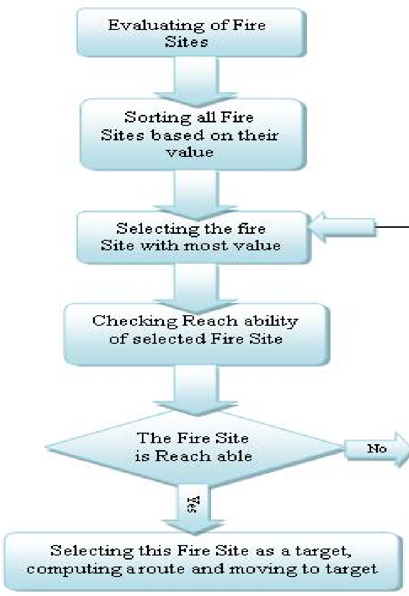

Figure 1. Decision Making Flowchart

Considering that there are numerous buildings and different situations in different maps, it is hard to find the most important building every cycle[2]. Also it is time consuming to change the position for extinguishing buildings which are far from each other. Therefore we apply a hierarchical decision making defining different sets of building, named *fire site, fire zone* and *burning building,* shown in Fig.2. The first step of decision making is selecting the best *fire site* which should be the most important one from different viewpoints from fire spreading to ability of the agents to extinguish it. The next question is that *"is the selected site reachable?"*

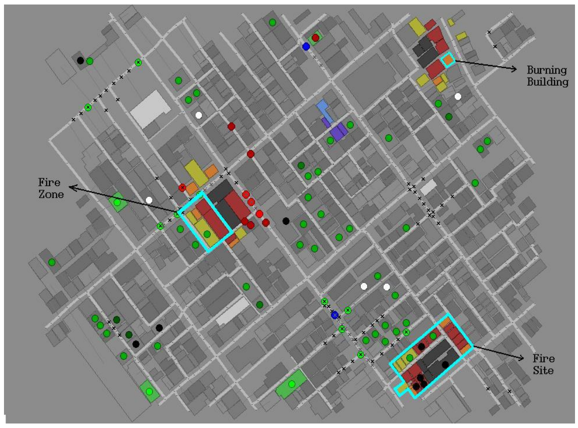

Figure 2. Definition of Fire Site, Fire Zone and Burning Building in Kobe

In this paper we want to find the answer to this question via Ant Colony Optimization. A fire site may have more than 20 buildings and only one of them may be reachable. Checking this property in other methods should be done by finding path to each of the buildings and if the path could not be found the next one should be checked. This process will be very time consuming. In next section you can see our approach to solve this problem based on ACO.

IV. Results and Discussion

As described in previous section our main problem is finding a path to one of the buildings of a *fire site*. In this section this problem will be solved and comparing to other methods presented to evaluate it.

A. Why does ACO work better in this problem?

As described before routing to many buildings will be time consuming and using the experience of path planning to each building can be helpful in routing to other buildings. Therefore a technique that has this property intrinsically will be more suitable to implement in this problem. The advantages of this method are explained in two parts in the following. The pseudo code of our approach which is applied in this paper is presented in the below either.

1) Signing the paths: In ACO the main characteristic is putting effect on paths with better chance to reach the goals. When the target is investigation about the existence of the open route to one of the buildings of *fire site* and next finding the reachable buildings, stigmergy prepares a significant preference. To reach a set of targets the pheromone laid down by ants can be utilized in routing to each of the targt buildings. In other words, determining a set of buildings instead of one, never increase the computation cost in this method and there is no need to route n times to n building placed in a *fire site*.

TABLE I. THE PSEUDO-CODE

```
Pseudo Code of the algorithm
definition of target( Fire Site)
for iteration n= 1, 2, …
   while termination condition is not met do
      for each colony in parallel do
          for k=1 to m ants do
               determent current positions of ant
               get valid connected paths (identity blockades and dead locked paths)
               compute probability for each connected path
               construct solutions and update pheromone
          end for
      end for
   end while
end for
display the best solution found so far by all colonies
```

2) *Probabilit based searching:* When the path selection in each step is based on probability distribution computed by pheromone quantity, the exploration importance will be more. It means that comparing with other methods with greedy policy, this algorithm gives chance to paths far from the targets to consider in ants tour. When blocked roads in future steps may banned the routes, considering more options will results in reaching the targets sooner very frequently. This is the common preference of random search approaches.

B. Simulation Results

In this section the quality of this approach is analyzed from two viewpoints. Comparing the time of routing to a fire site with 20 buildings in different maps is presented in Table II. These results are computed in different situations and they are the mean values of 100 cycles. This table shows that the average time required in A*, which is a common method in this field, is more than one second. This means that the action cannot be performed in many cycles and so the time will be wasted. To solve this problem most of the teams route to less building; e.g. our team, in the previous approach, selected four buildings around the site to check the reachability. In this manner many mistakes may be made and an attainable fire site may be considered as unreachable one. Better time consumed in ACO can be observed clearly in this table. Being less than 1 second (1000 ms) means we do not lose any cycle.

TABLE II. RESULTS OF COMPARING TWO DIFFERENT APPROACHES, THE TOTAL CONSIDERED TIME(MS) TO CHECK REACHABILITY OF FIRE SITE CONTAINING 20 BUILDINGS.

Method	Map							
	Foligno Med	*Foligno Hard*	*VC Med*	*VC Hard*	*Kobe Med*	*Kobe Hard*	*Random Small*	*Random Large*
*A**	1370	2110	1061	2085	1545	2131	1721	1039
ACO	801	853	775	830	731	880	651	693

The main evaluation in Rescue Simulation is the score which is computed based on the rescued civilians, and unburned buildings. Figure 3 displays difference between the influences of two routing methods on scores. This comes from 10 different maps' runs showing the preference of our approach.

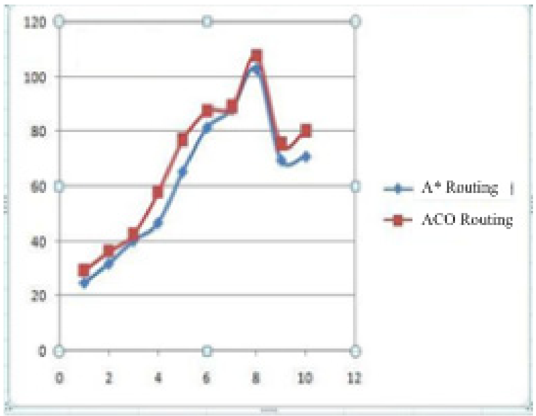

Figure 3. Result of comparing final score of previous approach and new approach in 10 simulations.

Finally figure 4 depicts two maps after running two approaches. The size of burned building shows better performance of ACO in total evaluation of agents work.

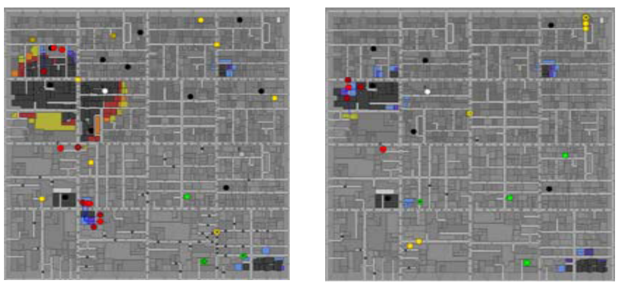

Figure 4. Simulation results in VC map (Left: Routing is based on A^* algorithm. Right: Routing is based on ACO algorithm)

2- The time period considered for making decision and sending it to the server is defined as cycle. One cycle is one second in Rescue Simulation.

REFERENCES

[1] S. J. Russell, P. Norvig, "Artificial Intelligence: A Modern Approach," Upper Saddle River New Jersey : Prentice Hall, pp. 97–104, 2003.

[2] H. R. Beom, H. S. Cho, "A Sensor-Based Navigation for Mobile Robot Using Fuzzy Login and Reinforcement Learning," IEEE Trans, On : SMC, vol.25(3), pp. 464~477,1995

[3] N. Bin, C. Xiong, Z. Liming, X. Wendong, "Recurrent Neural Network for Robot Path Planning," PDCAT 2004, LNCS 3320, vol.2, pp. 188–191, 2004.

[4] L. Xiong, F. Xiao-ping, Y. Sheng, Z. Heng, "A Novel Genetic Algorithm for Robot Path Planning in Environment Containing Large Numbers of Irregular Obstacles," ROBOT, vol 26, no.1, January 2004.

[5] Sh. Liu, Y. Tian, J. Liu, "Multi Mobile Robot Path Planning Based on Genetic Algorithm," Proceedings of the 5th World Congress on Intelligent Control and Automation, vol.5, pp. 4706-4709, 2004.

[6] J. E. Bell, P. R. McMullen, "Ant colony optimization techniques for the vehicle routing problem," Advanced Engineering Informatics, vol.18, no.1, pp.41–48, 2004.

[7] M. Dorigo, G. DiCaro, M. L Gambardella, "Ant Algorithms for Discrete Optimization," Artificial Life, vol. 5, no. 2, pp. 137-172, 1999.

[8] M. Dorigo, T. Stützle, "Ant Colony Optimization," Cabridge, MA: The MIT Press, 2004.

[9] Ch. Blum, "Ant colony optimization: Introduction and recent trends ," physics of life reviews, vol. 2, no.4, pp. 353-373, December 2005.

[10] J. L. Deneubourg, S. Aron, S. Goss, J. M. Pasteels, "The self-organizing exploratory pattern of the Argentine ant," Journal of Insect Behaviour , vol.3, no.2, pp. 159–168, March 1990.

[11] P. P. Grassé, "La reconstruction du nid et les coordinations inter-individuelles chez bellicositermes natalensis et cubitermes sp" La théorie de la stigmergie: Essai d'interprétation des termites constructeurs Insectes Sociaux , vol. 6, pp.41–81, 1959.

[12] T. Morimoto, "How to Develop a RoboCupRescue Agent," http://robomec.cs.kobe-u.ac.jp/robocup-rescue.

[13] O. Aghazade, M. A. Sharbafi, A. T. Haghighat, "Implementing Parametric Reinforcement Learning in Robocup Rescue Simulation," RoboCup 2007: Robot Soccer World Cup XI, Springer Berlin, Heidelberg, vol. 5001, pp. 409-416, 2008.

[14] J. Habibi, M. Ahmadi, A. Nouri, M. Sayyadian, M. Nevisi, "Utilizing Different Multiagent Methods in Robocup Rescue Simulation," In: D. Polani, B. Browning, A. Bonarini, K. Yoshida (eds.) RoboCup 2003, LNCS (LNAI), Springer, Heidelberg, vol. 3020, 2004.

[15] M. A. Sharbafi, C. Lucas, O. AmirGhiasvand, O. Aghazade, A. T. Haghighat, "Using Emotional Learning in Rescue Simulation Environment," Engineering, Computing and Technology, vol.13, pp.333-337, 2006.

[16] M. A. Sharbafi, O. A. ghiasvand, Z. Vosough, M. Eghbali, S. Mehrabi, R. MirSharifi, "RoboCupRescue 2008 – Rescue Simulation League Team Description MRL (Iran)," 2008.

1-This work is sponsored by Mechatronics Research Laboratory (MRL). MRL is an experienced team which gained many awards in RoboCup competitions. From 2005 to 2009, MRL could gain 4th, 1st, 1st, 4th and 2nd place in RoboCup WorldCup competitions respectively.