2009 World Congress on Nature and Biologically Inspired Computing, NABIC 2009; Coimbatore; 9 December 2009 through 11 December 2009; Category number CFP0995H; Code 79534;Article number 5393818, Pages 1145-1149
DOI- http://dx.doi.org/10.1109/NABIC.2009.5393818

Navigation of Multiple Mobile Robots Using Swarm Intelligence

Dayal R. Parhi
National Institute of Technology,
Rourkela, India
E-mail: dayalparhi@yahoo.com

Jayanta Kumar Pothal
National Institute of Technology,
Rourkela, India
E-mail: jayantapothal@yahoo.co.in

Mukesh Kumar Singh
Govt. Engineering College,
Bilaspur, India
E-mail: mukesh3003@yahoo.co.in

Abstract— **This research investigates the application of swarm intelligence principles for the co-operative behavior of autonomous collective robots. Using swarm intelligence technique robots are able to get their optimized path during navigation. A successful way of structuring the navigation task deals with the issue of co-operative behavior among multiple mobile robots. Using Ant Colony Optimization, robot path planning in two-dimension environment is studied. In this paper the current study introduces the intelligent finding optimum mechanism of ant colony. The theory has been tested both in simulation and experimental modes.**

Keywords—**Mobile Robot, Ant colony optimization, Swarm Intelligence, Navigation**

I. Introduction

Swarm intelligence is a particular form of collective intelligence which relies on the capabilities of several minimally intelligent but autonomous individual agents. The swarm intelligence approach implies distributed control strategies, i.e. strategies where coordination is taken over by redistributing most of the information gathered by the individual robots. Inspired by social insect societies, this approach exploits robot-robot and robot-environment interactions to develop robust, goal-oriented, and sometimes emergent collective behaviors.

According to Bonabeau et al. [1], an entity must satisfy its “viability” conditions while interacting in its environment. An intelligent individual must maintain an “identity” throughout its existence. The first category group is the subsumption architecture developed by Brooks [2]. These are behavior oriented control strategies where simple behaviors performed by Augmented Finite State achines, are hierarchically organized so that more complex behaviors emerge. It uses low-level behaviors like responses to sensors that are called upon and build on each other. As the situation changes, the robot switches between behaviors. The alliance system [3] is an example of this type of architecture. The motor-schema architecture proposed by Arkin [4] uses primitive behaviors (called motor schema) as the consumption architecture. The developed methodology has been applied to different robot morphologies, for different complex tasks, subjected to different environmental conditions. The main objective of this research is to achieve autonomous navigation, behavioral control, and speed control, of robots in a cluttered environment. The series of simulation test are conducted using MATLAB to demonstrate that the method have a great potential to solve the proposed problem. The experiments are done with the help of four khepera II mobile. The simulation results and experimental results are in very good agreement.

II. ANT COLONY OPTIMIZATION

ACO is an approximate optimization algorithm. It is useful on problems which are impossible or very time consuming to solve exactly. The field of ‘‘ant algorithms’’ studies models derived from the observation of real ants’ behavior, and uses these models as a source of inspiration for the design of novel algorithms for the solution of optimization and distributed control problems. Such problems include difficult logistics problems, routing in telecommunication networks and scheduling of school classes.

Simple-ACO (S-ACO)

In the following section it is explained how the ants’ behavior, as well as pheromone evaporation, is implemented in an algorithm that we call Simple- ACO. Here considered a static, connected graph G = (N, A), where N is the set of n = | N | nodes and A is the set of undirected arcs connecting them. The two points between which we want to establish a minimum cost path are called source and destination nodes.

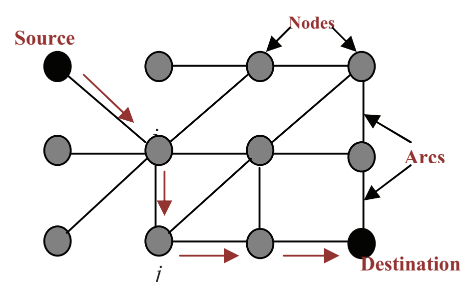


FIGURE 1: Path from a source to a destination node

To each arc (i, j) of the graph G = (N, A) we associate a variable τij called artificial pheromone trail, shortened to pheromone trail in the following. Pheromone trails are read and written by the ants. The amount (intensity) of a pheromone trail is proportional to the utility, as estimated by the ants, of using that arc to build good solutions.
Each ant builds, starting from the source node, a solution to the problem by applying a step-by-step decision policy (Eq. 1). At each node, local information stored on the node itself or on its outgoing arcs is read (sensed) by the ant and used in a stochastic way to decide which node to move to next. At the beginning of the search process, a constant amount of pheromone $(e.g., \tau ij = 1, \forall\ (i,j)\ \epsilon\ A\ )$ is assigned to all the arcs. When located at a node i an ant k uses the pheromone trails τij to compute the probability of choosing j as next node:

$$p_{ij}^{k} = \begin{cases} \dfrac{\tau_{ij}^{\alpha}}{\sum_{l \epsilon N_i^k} \tau_{il}^{\alpha}}, & if j \in N_i^k; \\ 0, & if j \notin N_i^k; \end{cases} \tag{1}$$

where N_i^k is the neighborhood of ant k when in node i. Where $\propto$ is the parameter which determines the relative influence of the pheromone trail and the heuristic information. In S-ACO the neighborhood of a node i contains all the nodes directly connected to node i in the graph G = (N, A), except for the predecessor of node i (i.e., the last node the ant visited before moving to i). In this way the ants avoid returning to the same node they visited immediately before node i. Only in case N_i^k is empty, which corresponds to a dead end in the graph, node i's predecessor is included into N_i^k.

Ant System (AS)

The two main phases of the AS algorithm constitute the ants' solution construction and the pheromone update. In AS a good heuristic to initialize the pheromone trails is to set them to a value slightly higher than the expected amount of pheromone deposited by the ants in one iteration.

In AS, m (artificial) ants are put on randomly chosen nodes. At each construction step, ant k applies a probabilistic action choice rule, called random proportional rule, to decide which city to visit next. In particular, the probability of ant k, currently at node i, chooses to go to node j is

$$p_{ij}^{k} = \frac{[\tau_{ij}]^{\alpha}[\eta_{ij}]^{\beta}}{\sum_{l \in N_i^k}[\tau_{ij}]^{\alpha}[\eta_{ij}]^{\beta}}, \text{ if } j \in N_i^k, \tag{2}$$

Where $\eta_{ij} = {}^{1}/_{d_{ij}}$ is a heuristic value that is available a priori, d is the distance between nodes, α and β are two parameters which determine the relative influence of the pheromone trail and the heuristic information, and N_i^k is the feasible neighborhood of ant k when being at city i, that is, the set of cities that ant k has not visited yet. By this probabilistic rule, the probability of choosing a particular arc (i,j) increases with the value of the associated pheromone trail τ_{ij} and of the heuristic information value η_{ij}. The role of the parameters $\propto$ and β is the following. If α = 0, the closest cities are more likely to be selected: this corresponds to a classic stochastic greedy algorithm (with multiple starting points since ants are initially randomly distributed over the cities). If β = 0, only pheromone amplification is at work, that is, only pheromone is used, without any heuristic bias. This generally leads to rather poor results and, in particular, for values of α > 1 it leads to the rapid emergence of a stagnation situation, that is, a situation in which all the ants follow the same path and construct the same tour, which, in general, is strongly suboptimal.
Each ant k maintains a memory M^k which contains the cities already visited, in the order they have visited. This memory is used to define the feasible neighborhood N_i^k in the construction rule given by equation (2). In addition, the memory M^k allows ant k both to compute the length of the tour T^k it generated and to retrace the path to deposit pheromone.

After all the ants have constructed their tours, the pheromone trails are updated. This is done by first lowering the pheromone value on all arcs by a constant factor, and then adding pheromone on the arcs the ants have crossed in their tours. Pheromone evaporation is implemented by

$$\tau_{ij} \leftarrow (1-\rho)\tau_{ij}\ ,\ \forall\ (i,j)\ \in\ L \tag{3}$$

Where 0< ρ ≤1 is the pheromone evaporation rate. The parameter ρ is used to avoid unlimited accumulation of the pheromone trails and it enables the algorithm to ''forget'' bad decisions previously taken. In fact, if an arc is not chosen by the ants, its associated pheromone value decreases exponentially in the number of iterations. After evaporation, all ants deposit pheromone on the arcs they have crossed in their tour:

$$\tau_{ij} \leftarrow \tau_{ij} + \textstyle\sum_{k=1}^{m} \Delta\,\tau_{ij}^{k}, \quad \forall\ (i,j) \in L \tag{4}$$

Where $\Delta\tau_{ij}^k$ is the amount of pheromone ant k deposits on the arcs it has visited.

It is defined as follows:

$$\Delta\tau_{ij}^{k} = \begin{cases} {}^{1}/_{C^k}, & if\ arc\ (i,j) belongs\ to\ T^k\ ; \\ 0, & otherwise\ ; \end{cases} \tag{5}$$

Where C^k, the length of the tour T^k built by the k^{th} ant, is computed as the sum of the lengths of the arcs belonging to T^k. By means of equation (5), the better an ant's tour is, the more pheromone the arcs belonging to this tour receive. In general, arcs that are used by many ants and which are part of short tours, receive more pheromone and are therefore more likely to be chosen by ants in future iterations of the algorithm.

III. PROBLEM DESCRIPTION

In this paper a 2-D square map overlaid with a uniform pattern of grid points has been considered. The size of a map can be changed arbitrarily; here, the map consists of a 20 x 20 grid. The left bottom corner of the map is the starting point for a path while the right top corner of the map is the destination point for a path. The shape of an obstacle is always a circle, but the size of the obstacle is variable from 1 to 5 grid points. The positions of the obstacles are randomly selected and can be located at any grid point in the map except at points close to the starting point region or close to the goal point region. Furthermore, multiple obstacles are possible. FIGURE 2 shows an example of such an arrangement.

IV. PROPOSED SOLUTION

The Travel salesman problem (TSP) is a paradigmatic optimization problem since it is used to demonstrate the original AS (Ant System) problem. Since then it has often been used as a benchmark to test the new ACO concept. The proposed path-finding algorithm employs the original concept of the ACO algorithm with some modification. The idea of the proposed algorithm is as follows.

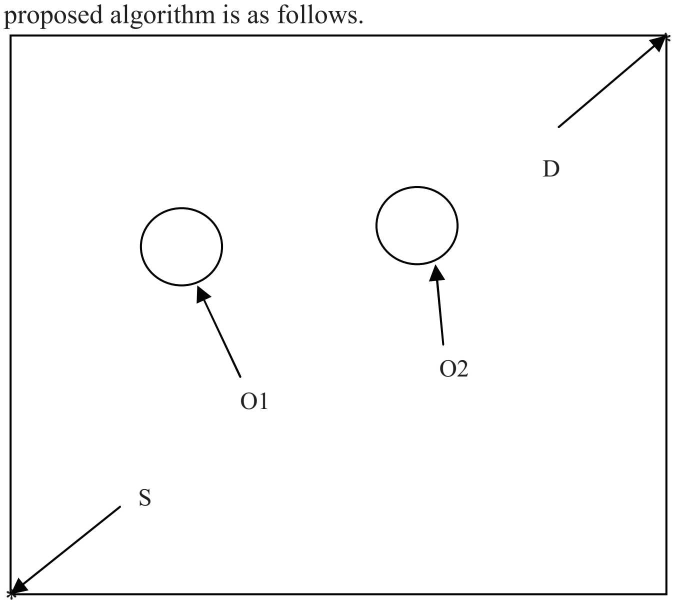


FIGURE 2: Problem configuration (S: starting point, O1, O2: obstacles, D: destination point)

Starting from the grid point (0, 0), an ant iteratively moves from a grid point to one of its neighboring grid points. When at the ith grid point (x, y), ant k can choose the next (jth) grid point by choosing one of its 8 neighbor locations: [(x+1, y+1), (x+1,y), (x, y-1), …..] ∈ N,where N in general is the set of all neighboring locations of the current location.

The ant takes its next step randomly, based on the probability given by

$$P_{ij}^{k}(t) = \frac{[\tau_{ij}]^{\alpha}[\eta_{ij}]^{\beta}}{\sum_{l \in N_i^k}[\tau_{ij}]^{\alpha}[\eta_{ij}]^{\beta}} + \alpha * \Omega_{ij} \quad (6)$$

Where $\tau_{ij}(t)$ is accumulated pheromone on the jth grid point when the ith grid point is the ant's current location at time t. The quantity τ_{il} indicates every possible lth neighbor point when the ant is in ith position. Ω_{ij} is the dot product of the vector from i to j with the vector i to destination point. The first term is associated with the pheromone amount in the 8 neighboring grid points (a local pheromone). The second term is associated with a global attraction, where α is a scale parameter. The global term is defined as the dot product of the agent's heading direction and the food (destination) direction. The purpose of the global term is to guide the agent in the desired direction so that a large number of ants can reach the goal point in a limited number of steps.

TABLE 1: An agent's current position (bold) with 8 possible next positions

(x-1, y+1)	(x, y+1)	(x+1, y+1)
(x-1, y)	**(x, y)**	(x+1, y)
(x-1, y-1)	(x, y-1)	(x+1, y-1)

The solution construction ends after each ant reaches the destination or the ant took too many steps (50 or 100 steps, for example). However, the pheromone will be deposited only if the ant reaches to the destination position in less than a certain number of steps. Thus, after a group of agents finished its tour, the pheromone in the entire map will be updated by

$$\tau_{ij}(t+1) = (1-\rho) * \tau_{ij}(t) + \sum_{k=1}^{m} \Delta\gamma_{ij}^{k}(t) \quad (7)$$

Where 0 < ρ < 1 is the pheromone forgetting parameter. This parameter prevents the map from unlimited accumulation of the pheromone. The quantity $\Delta\gamma_{ij}^{k}$ is the amount of pheromone that ant k deposits on the map. It is defined as

$$\Delta\gamma_{ij}^{k}(t) = \frac{1}{L^k(t)} \quad (8)$$

Where $L^k(t)$ is the length of the kth ant's tour (path). When the kth ant cannot reach the destination in a certain number of steps, let the $L^k(t)$ = Infinity.

V. RESULTS DISCUSSION

The experiments are done with different number of obstacles, different population sizes, and a different number of iterations. Furthermore, these studies include the effects of elitism. The circular objects represent the obstacles. The obstacles can be located at any place on the map except at the starting point and at the destinatiSon point. The experiments are conducted using MATLAB.

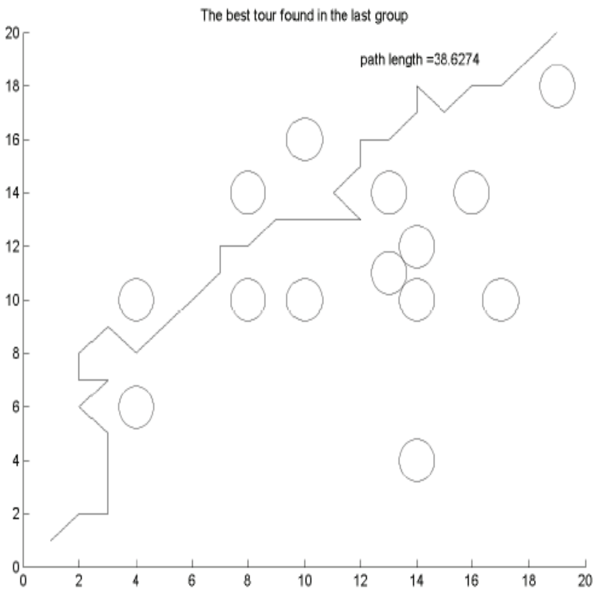


FIGURE 3: Path followed by 100 robots avoiding multiple obstacles

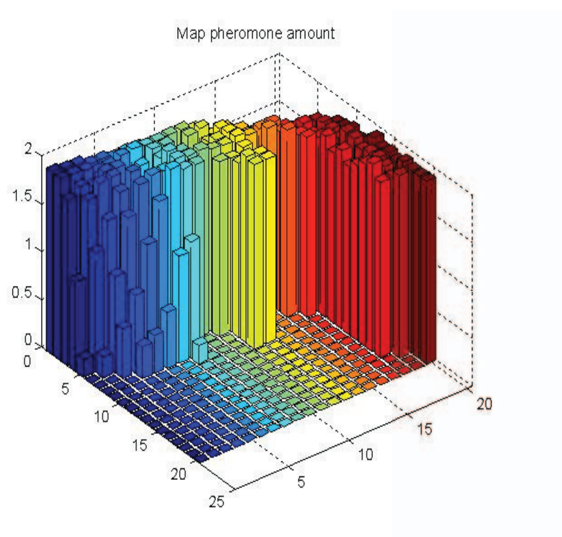


FIGURE 4: Pheromone accumulation in 3D

The simulated robot path planning algorithm generated and set the obstacles randomly. FIGURE 3 shows the result obtained from simulation, (0, 0) is start node and (20, 20) is target node. The optimum path is shown with line. From FIGURE 3 it is clear that with the help of proposed approach robots are able to find the optimal path by avoiding the obstacles.

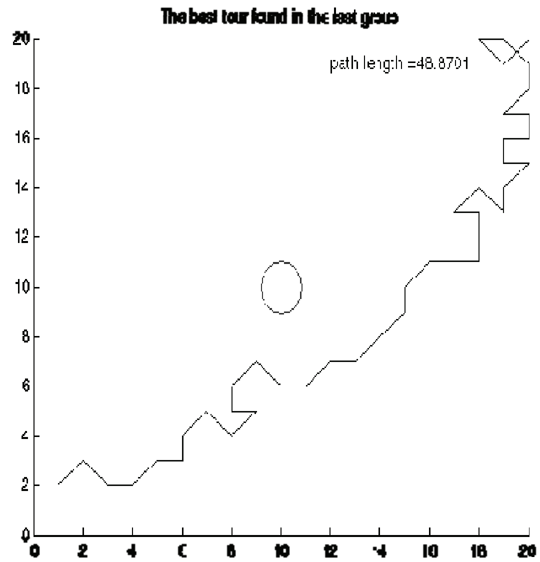
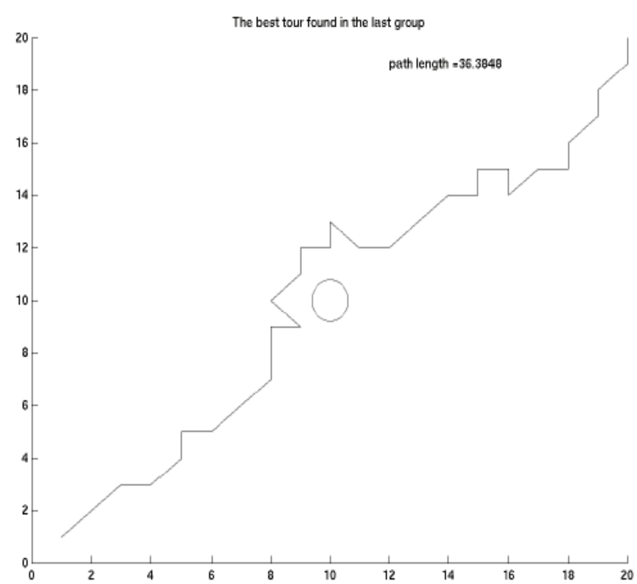


FIGURE 5: Path followed by 10 numbers of ants

FIGURE 6: Path followed by 100 numbers of ants



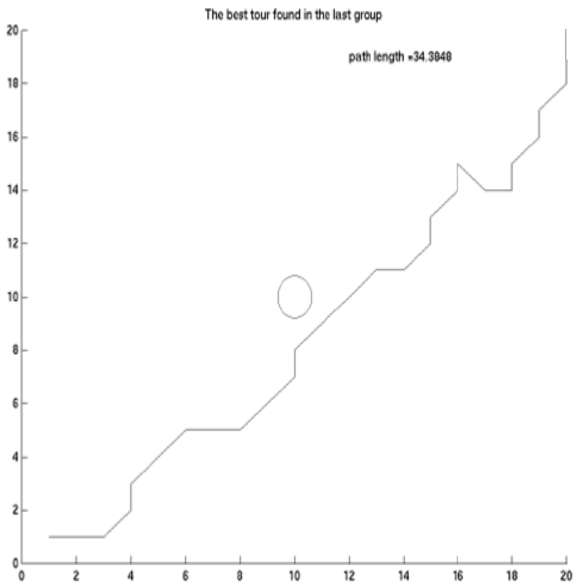


FIGURE 7: Path followed by 500 numbers of ants

TABLE 2: Effect of number of agents

The number of ants	10	100	500
Average path length of 50 trials	48.8701	36.3848	34.3848

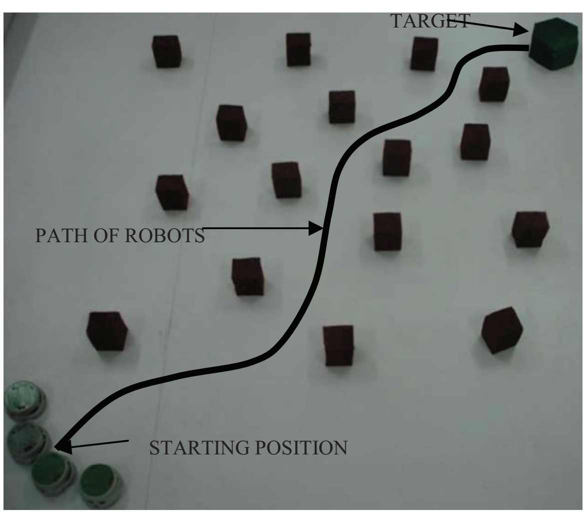


FIGURE 8: Path followed by Mobile robots during Experiment

From the results (FIGURE 5, FIGURE 6 and FIGURE 7) it is observed that a larger size group is better than a smaller size group. With the increase of numbers of Robots, a better optimized path is obtained (resulting less distance travel to reach the goal). If the group size is too small, not only does the number of ants that can reach the destination get smaller but also the amount of pheromone accumulation decreases. FIGURE 8 shows the Path of four Khepera II Mobile Robots during experiment. The Simulation results and Experimental results are in good agreement.

VI. Conclusion

From the above results and discussion it is concluded that with the help of developed ACO algorithm robots are able to find the target in optimal path by avoiding the obstacles. During comparison of the results it is concluded that a larger size group of robots is better than a smaller size group of robots. In this paper, an algorithm based on ant colony is proposed to solve robot path planning problems. The proposed ant algorithm has the following characteristics: (1) it hurdles the abuse of local optimization and expedite searching speed, with the experiments, the method is proved effective. (2) The algorithm has good scalability. This is because the ant algorithm leaves information on the path for the sake of next routing during the search process. The information can be kept when more nodes are added. In the future, multi-robot path planning problem in dynamic obstacle environment will be studied. At that time the foregoing knowledge methodology will be used.

References

[1] E. Bonabeau, M. Dorigo, and G. Theraulaz, “Swarm Intelligence: From Natural to Artificial Systems,” Oxford University Press, New York, 1999.

[2] R. Brooks, “A robust layered control system for a mobile robot,” IEEE Journal of Robotics and Automation, 1997, Vol. 2(1), pp. 137-153.

[3] L.E. Parker, “ALLIANCE: An architecture for fault tolerant multirobot cooperation,” IEEE Transactions on Robotics and Automation, vol 14(2), pp. 220-240, 1998.

[4] R.C. Arkin, “Behavior-Based Robotics,” the MIT Press, Cambridge, 1998.

[5] Ying-Tung Hsiao, Cheng-Long Chuang, Cheng-Chih Chien, “Ant colony optimization for best path planning,” Communications and Information Technology, Proceedings of the IEEE Int. Symposium on Communications and Information Technology, 2004, vol.1, pp109- 113.

[6] M.M. Mohamad, M.W. Dunnigan, N.K. Taylor, “Ant Colony Robot Motion Planning,” Proceedings of the IEEE Int. conference on Computer as a Tool, 2005, vol.1, pp 213-216.

[7] Yu Bin, Yang Zhong-Zhen, Yao Baozhen,“An improvedant colony optimization for vehicle routing problem,” European Journal of Operational Research,2009, vol.196, pp171–176.