

Communication Enhanced Navigation Strategies for Teams of Mobile Agents

Justin Hayes, Martha McJunkin and Jana Košecká
Department of Computer Science, George Mason University
4400 University Drive, MS 4A5, Fairfax, VA 22030

Abstract—In multi-agent systems engaged in cooperative activities there is an apparent trade-off between the complexity of the individual agents, their sensing capabilities and communication required for accomplishment of particular tasks. One of the main computationally intensive components which affects the complexity of the overall system is the acquisition and maintenance of the environment model where the agents reside. In this paper, in the context of foraging and coordinated traversal task, we will examine control strategies that in the absence of the global model of the environment can substantially improve the performance of the team using additional sensing and communication capabilities. In one case the coordinated strategy is motivated by an ant trail following behavior while in another case the line of sight information is used to constrain the movement of individual agents guaranteeing shorter total traversal times.

I. INTRODUCTION

The area of multi-agent robotic systems has been considered in recent years as a suitable platform for studying design and analysis issues of distributed hybrid systems. The applications varied in the type of environment considered, level of coordination between the agents as well as complexity of the overall system. At the one end of the spectrum, the researchers considered complex, typically hierarchical, architectures, where the higher level planner maintained the consistent model of the environment and was responsible for determining higher level goals and coordination strategies of individual agents [10]. On the other hand, it has been demonstrated that a variety of simple robotic tasks can be accomplished in a purely distributed manner by using only local communication and sensing and invoking appropriate feedback strategies in different stages of the task completion. Within these behavior based approaches, the (emergent) global qualitative behavior was achieved through interactions between individual agents and the environment. In the context of multi-agent systems this approach has been applied successfully to tasks such as dispersion, flocking, foraging, and coverage [11], [2]. More quantitative statements about properties and correctness of global strategies emerging from local interactions have been given in [6], [3]. The control laws of individual agents were designed to follow a particular possibly time-varying vector field, computed as a gradient of an appropriate potential function [2]. The use of potential functions for either global or local

navigation in dynamic and complex environments often results in the failure to reach the desired destination, due to the presence of local minima. In such cases either random walk methods were used to escape from local minima or global planner was used to select an alternative waypoint [13]. Global replanning however requires a model of the particular environment and hence additional computational overhead. In multi-agent systems maintaining the consistent model of the environment is typically accomplished in a centralized or distributed manner. In both cases it requires additional communication overhead either between individual agents and higher level planner or among agents themselves. From the perspective of environment modelling, we will demonstrate two strategies that in the presence of limited local information about the environment substantially improve the performance of multi-agent teams without a need for centralized coordination unit or additional communication overhead. With the advent of new technologies, such as mobile sensor networks, the design of coordination strategies which are of low computational/communication cost is of importance [14].

The first strategy is examined in the context of the foraging task and is motivated by an ant trail following behavior. In the second, coordinated traversal task, we will propose an alternative strategy where the lack of the global view of the environment is complemented by coordination between individual members of the team exploiting the line of sight information. In the rest of the paper we will describe the two tasks considered, and report the results of simulations, which demonstrate the improvements in the performance of the team.

II. FORAGING TASK

In the foraging task the team of mobile agents is tasked to search the environment for resources and deliver them to the home base. In the absence of the environment model and the computational resources to build one and maintain it, individual agents typically resort to random walk strategies. Our approach is motivated by strategies used in biological systems whereby the collective behavior of multiple unsophisticated agents interacting with their local environment exhibits problem-solving capabilities. Using this approach, certain types of problems can be solved

without the need for centralized control or a global model. In the natural world, an ant colony must employ some intelligent way to gather the resources so it can survive. Simply wandering around looking for the resources is not likely to lead to the long-term survival of the colony. One of the methods that ant colonies use to improve on this approach is to use the pheromone trail [1]. After an ant locates a resource, it takes a piece of the resource and delivers it back to the nest, emitting pheromones as it goes. By doing so, it creates a pheromone trail between the resource location and the nest. Other ants can sense the pheromones and therefore be attracted to the trails, and hence can discover the resources more efficiently.

The ant trail analogy as well as pheromones properties have been exploited previously in the context traversal problems in the discrete setting, where the environment is represented as a graph. In [9] the authors demonstrated that multi-robot exploration formulated as a graph traversal can be successfully accomplished by a team of agents despite the global knowledge of the graph. The ant-inspired problem solving strategies have also been analyzed theoretically using real-time heuristic search methods and implemented on actual robots engaged in traversal and coverage tasks [7]. Using the same computational paradigm, the so-called virtual pheromones were introduced, where the actual sensing was replaced by communication in the limited range [15]. Employing the trail-following strategies in the foraging task requires to resolve additional issues to make them effective.

We show how to incorporate the pheromones trails in distributed foraging task and demonstrate that the trail following behavior substantially improves the performance and the survival of the ant colony. The Java Teambots [8] simulator has been used to implement the simulation environment (see Figure 3). We are currently extending the simulator with an interface for visualizing the pheromone trails. The Teambots simulator adopts the schema based approach [2] for representing tasks of the individual agents modelled in terms of finite state machines (FSM). Within each state of FSM particular reactive feedback strategy is applied and the transition between the states are triggered by events, which typically correspond to a completion of a strategy or an externally observed event.

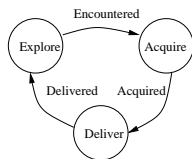


Fig. 1. Finite State Machine representing the Forage task.

The FSM for the foraging task as used in the Teambots Simulator is in Figure 1. The agent in the Explore state

is engaged in random exploration strategy while avoiding the obstacles and other agents. Upon encountering the resource the transition to the Acquire state is made and the resource is taken. The resource acquisition is followed by the delivery (state Deliver) of the resource to the home base and continued search for the resources. In Explore and Deliver states of the FSM, the navigation strategy of each agent is governed by a commonly used control law

$$\begin{aligned}
 U(\mathbf{x}, t) &= U_a(\mathbf{x}, t) + U_r(\mathbf{x}, t) \\
 \dot{\mathbf{x}}_g &= -k_g \frac{\nabla U(\mathbf{x}, t)}{\|\nabla U(\mathbf{x}, t)\|},
 \end{aligned} \tag{1}$$

where U_a and U_r are the attractive and repulsive components of the vector field, $k_g \in \mathbb{R}$ and $\mathbf{x} = [x, y]^T$ is the current pose of the robot. Various choices of $U(\mathbf{x}, t)$ have been discussed previously in [16], [17], [18]. The main modification we propose here is to augment the exploration strategy Explore with a partial local model of the environment, which can be sensed by the individual agents. When the agent acquires the resource it lays down a trail of pheromones as the food is delivered to the goal. The pheromone trail can be sensed by other agents and will affect their behavior in the exploration stage. The presence of pheromone trail will cause an adjustment of the robots heading to account for the pheromone attraction.

A. Representation of the Pheromone Trails

For the purpose of simulation a global pheromone map is maintained in terms of a discrete grid. Each location in the pheromone map contains a value representing the strength of the pheromone trail at that location in the real world. An ant robot is only affected by the pheromone trails if it is in the Explore state. The pheromone trail attraction vector \mathbf{u}_p is used to adjust the heading and the speed of the agent

$$\dot{\mathbf{x}}_p = k_p \mathbf{u}_p.$$

The computation of the vector \mathbf{u}_p depends on the type of function used to update the pheromone level. The direction and magnitude of the attraction vector \mathbf{u}_p is computed either as a linear combination of the directions of nearest pheromones lying within the certain radius of agent, scaled by the magnitude of the pheromone level or the direction of the maximal pheromone level. Since the pheromone map is represented as a grid, the nearest neighbors are searched in the 8-neighborhood of agent's location (in directions N, NE, E etc.) depicted in Figure 2. Furthermore we assume that the agent at each instance knows what is the heading direction to the home base with location $\mathbf{x}_n = [x_n, y_n]^T$, denoted by unit vector \mathbf{u}_n . The trail attraction is taken into effect only when the attraction vector \mathbf{u}_p is pointing away from the nest. In such case the

inner product between the \mathbf{u}_n and the attraction vector \mathbf{u}_p satisfies following relationship

$$\left(\frac{\mathbf{x}_p}{\|\mathbf{x}_p\|} \right)^T \left(\frac{\mathbf{x} - \mathbf{x}_n}{\|\mathbf{x} - \mathbf{x}_n\|} \right) < 0,$$

in another words the angle θ between the two vectors $\theta \in (\frac{\pi}{2}, \frac{3\pi}{2})$. This is especially important once several pheromone trails have been laid down because the strongest concentration of pheromones will be right around the nest where all of the trails converge. If this provision is not made, robots will congregate around the nest. This condition also prevents the agent from following the old pheromone trails, which lead towards the depleted resource. The pheromone level is increased

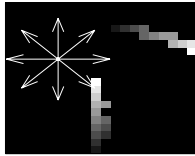


Fig. 2. Pheromone Trail Attraction Computation.

only when the agent is laying down the trail while going towards the home base and carrying the resource. We have investigated two different methods for depositing the pheromones, which yield similar performance. In one case the grid location corresponding to the present discretized position (i, j) of the agent $\text{map}[i, j] = t_k$ is assigned the current value of the simulation clock, which represents the pheromone level; in another case the pheromone level is simply incremented by a constant amount c , $\text{map}[i, j] = \text{map}[i, j] + c$. Another key decisions are that the pheromone trails must evaporate over time. In the current simulation the pheromone level decreases every 100 time steps by the amount $0.1c$; proportional to the constant depositing factor. The world used in simulation subtends a rectangle with dimensions 5000×5000 units. The size of the actual map is reduced by factor of 10. The trail evaporation assures that the robots over time are not attracted to pheromone trails that head to locations in the world where resources used to be. We are currently investigating alternative choices for the pheromone evaporation function as well as different strategies for following the pheromone trails.

B. Simulations of the foraging task

To determine the effect of pheromones on the ability of the ant colony to forage for resources several experiments were carried out. A sample world is created, 10 simulations are run without any pheromone trails, and 10 are run with pheromone trails. The colony starts with a set number of resources and each robot expends a constant amount of resources every second that the simulation is running. If the colony runs out of resources, it dies. Figure 4 shows

the number of additional resources the colony collects in each of the trials with and without pheromones. Figure 5 illustrates the survival time of the colony in each of the trials, again, both with and without pheromones. The colony collects, on average, over 14 more resources and survives over 50 seconds longer when pheromones are in use.

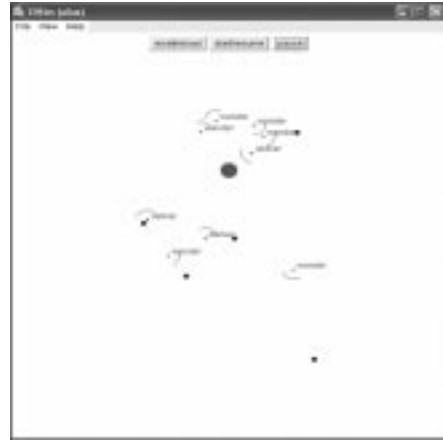


Fig. 3. Simulation environment provided by Teambots simulator. The nest (largest circle) is in the middle, while the five resources are scattered around in the environment. The agents are denoted by the smallest circles, with an arc representing the field of view of the agent.

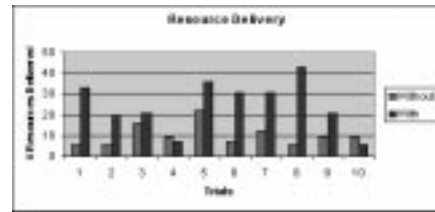


Fig. 4. Number of resources collected by the colony with and without the pheromone trails.

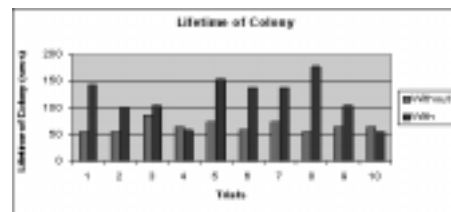


Fig. 5. Survival time of the colony with and without the pheromone trails.

Not surprisingly the results of the initial simulations clearly indicate that the use of pheromones presents us with a better solution to the ant colony resource foraging problem than the random wandering approach. The effectiveness of the pheromone trail foraging strategy also depends on the number of agents (see Figure 7). With

the small number of agents the collection of resources proceeds more slowly, as it takes more time until the trails are laid down. On the other hand when the number of agents increases the total number of collected resources does not increase proportionally. In this scenario the trail following becomes less effective due to the fact that the density of the agents in the trail vicinity increases and individual agents spend more time avoiding collisions, then progressing towards the goal. In the current simulation the ant-robots are only attracted to trail while searching for the resource. In case there is no trail in the vicinity of the agent, the agent is exploring the environment randomly. The use of the trails left behind during the exploration phase, combined with trail avoidance has been used extensively in the coverage tasks [7].



Fig. 6. Pheromone trails for the environment in Figure 3; three out four trails leading to resources have been laid out.

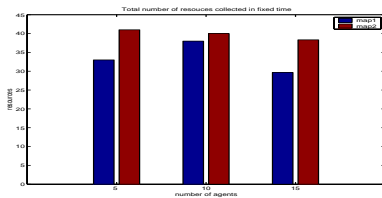


Fig. 7. Total number of resources collected during fixed period of time with two different mapping strategies and varying number of agents.

We are currently investigating the incorporation of both types of trails as well as interference effects between the two in order to be able to accomplish a bigger variety of tasks. The realization of the trail-following strategy on the physical robots requires additional trail sensing and trail leaving capabilities, whose performance in real scenarios differs significantly.

III. COORDINATED TRAVERSAL TASK

The second coordination strategy between multiple agents will be described in the context of the coordinated traversal task. Initially the agents are dispersed randomly in the complex environment and are tasked to reach a specified location. The navigation capabilities of the individual

agents are comprised of simple reactive strategies for heading towards the goal and avoiding the obstacles given by equation (1). Due to the complexity of the environment and the absence of the global model, the local strategies are clearly ineffective. In case the agent encounters a local minimum random walk strategy is employed to escape from the minima. In this scenario we will demonstrate that using the line of sight information the agents can coordinate and accomplish the task more effectively. The premise of the approach is, that the at each instance of time the agents locally compute so called *visibility graph*. The nodes of an undirected graph correspond to the robotic agents $a_i, i = 1, \dots, n$ and a_0 the goal. The edges represent the visibility information; if a_i and a_j are visible an edge (a_i, a_j) is present. The graph in the context of our simulation is represented by an adjacency matrix A . At every time step, each agent determines the visibility relationship with respect to the other agents and the goal. Once the graph is constructed the shortest path to the goal is computed for each agent using the standard Dijkstra's algorithm. The shortest path for each agent represented as a sequence of vertices a_k, \dots, a_0 is then used to determine leader-follower relationship between the team members. In particular, if an agent a_i is a predecessor of agent a_j on the shortest path towards the goal $a_i \prec a_j$, the motion of the agent a_i is suspended until agent a_j approaches its vicinity. This successor/predecessor relationship changes over time as the team progresses towards the goal and the visibility graph and shortest path are being recomputed. In spite of this sequential chaining strategy for the team, the overall performance of the team improves.

Agents that were initially not connected to the graph, use the basic navigation strategy determined by vector field $\dot{\mathbf{x}}_g$ given by equation (1).

A. Simulations of the coordinated traversal

The simulations were performed using the Teambots simulator in two different environments shown in Figure 8 and 9. The goal was located in the middle of the right boundary of the environment. The line of sight between the agents, or the goal is depicted by a line segment connecting the two. This information as well as the shortest path computation is changing dynamically as the agents negotiate the environment.

For the simpler environment the results of 20 runs with 12 mobile agents being randomly located at the start of each run are reported in Table 1. The average distance agents travelled in a run and the total time for all agents to reach the goal has been computed for the uncoordinated case (without using the line of sight information) and the coordinated case. For each run, the agents began at the same location for each of the control strategies. As was expected, the use of the coordination strategies improves the performance dramatically. In a more complex envi-

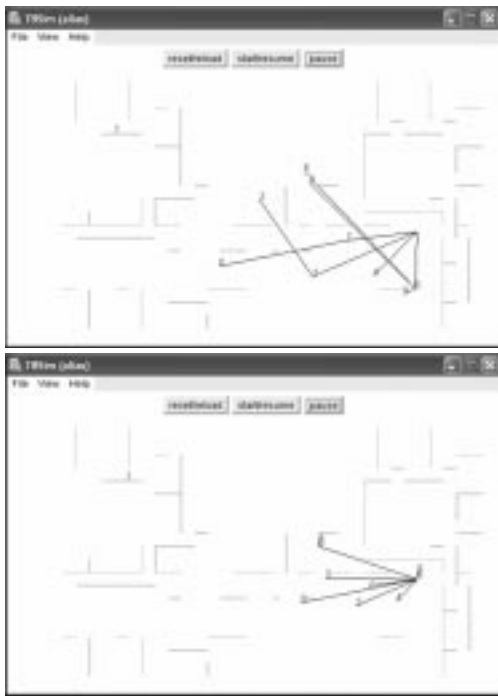


Fig. 8. Stages of the coordinated traversal task using the line of sight information in the simple environment.

	Uncoordinated	Coordinated
mean	71.38	194.08
std	2.18	29.94

TABLE I

COMPARISON OF COORDINATED AND UNCOORDINATED STRATEGIES FOR THE ENVIRONMENT IN FIGURE 8.

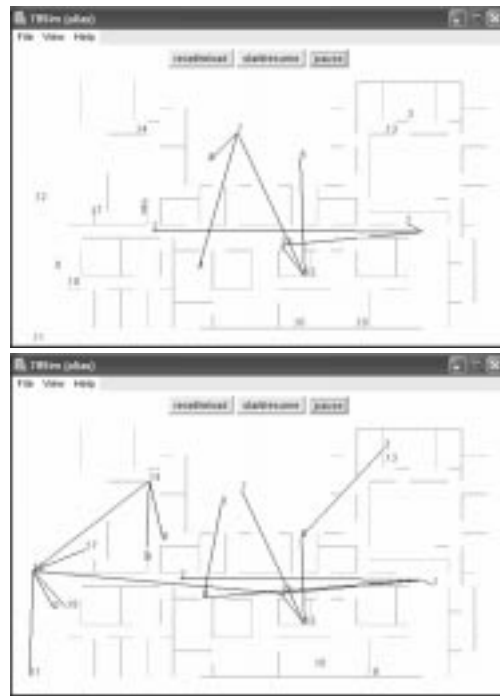


Fig. 9. Stages of the coordinated traversal task using line of sight information in more complex environment.

	# of agents	time	distance
mean	19.2	990.4	38.10
std	2.98	210.27	9.23

TABLE II

PERFORMANCE STATISTICS OF THE TRAVERSAL TASK FOR THE ENVIRONMENT IN FIGURE 9.

ronment as in Figure 9 reaching the final destination in the absence of the global environment model using purely reactive strategies is more time consuming. Given a set of random initial positions of the agents, larger percentage of them is initially not connected to the goal. Once the goal is encountered by at least one agent, the initial visibility graph is formed and the coordinated traversal becomes effective. The simulations are averaged over 20 runs, with 25 agents. The Table 2 shows the the number of agents which completed the task (arrived within the largest connected component), average task completion time and the distance travelled. The traversal task was considered completed once all the agents in the largest connected component of the graph reached the goal. The comparison with an uncoordinated strategy was not carried out due to the fact, that in the absence of coordination the task completion using purely local would be very ineffective.

The effect of number of agents on the completion time was also examined in the context of this environment. The dependency of the task completion and total distance

travelled was tested in simulation for varying number of agents and different environments. The results are reported in Figures 10 and 11. As we can see, not surprisingly, the total completion time increases with the complexity of the environment, but the average completion time per agent decreases as the number of agents grows and the area/distance traversed by the agents also grows.

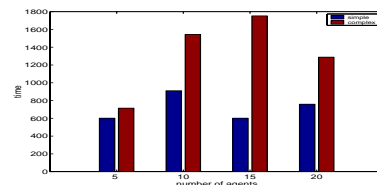


Fig. 10. Traversal completion time averaged over 10 trials for two types of environments and varying number of agents.

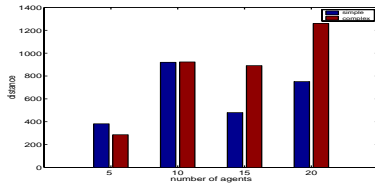


Fig. 11. Total distance travelled for two types of environments and varying number of agents.

IV. CONCLUSION

In this article we have presented two different coordination strategies for teams of mobile agents engaged in navigation and foraging tasks. The strategies are purely local, do not assume any model of the environment and either utilize additional sensing or line of sight communication between the individual team members. In the initial experiments have shown that the tasks can be accomplished more effectively in spite of the lack of global model of the environment. We are currently investigating the scalability properties of the approach, alternative properties of the pheromone trails, as well as extensions to more complex environments. The initial insights offer promising avenue for providing more quantitative statements about communication, sensing and complexity tradeoffs in the multi-agent teams.

Acknowledgements. We would like to thank Liviu Panait and Sean Luke for their help with and extensions to the Teambots simulator.

V. REFERENCES

[1] C. Anderson and J. Bartoldi. Centralized vs. Decentralized Control in manufacturing: lessons from social insects. Complexity and Complex Systems in Industry, Proceedings, University of Warwick, September 2000.

[2] R. Arkin Behavior-Based Robotics. MIT Press, 1998.

[3] J. Cortes, S. Martinez, T. Karatas, and F. Bullo. Coverage Control for Mobile Sensing Networks Submitted to the IEEE Transactions On Robotics and Automation, Nov 2002.

[4] L. M. Gambardella and M. Dorigo. Solving Symmetric and Asymmetric TSPs by Ant Colonies. Proceedings of the IEEE International Conference on Evolutionary Computation. May 20-26, 1996. Nagoya, Japan. 622-627.

[5] A. Coloni, M. Dorigo, V. Maniezzo. Distributed Optimization by Ant Colonies. Proceedings of ECAL 91(European Conference on Artificial Life). Elsevier Publishing. Paris, France. pp. 134-142.

[6] A. Jadbabaie, J. Lin and A. S. Morse. Coordination of Groups of Mobile Autonomous Agents Using Nearest Neighbour Rules. IEEE Conference on Decision and Control, 2002.

[7] J. Svennebring and S. Koenig. Building Terrain-Covering Ant Robots. Technical Report, GIT-COGSCI-2002/10.

[8] T. Balch. Teambots simulator, <http://www.teambots.org>.

[9] I. A. Wagner, M. Lindenbaum and A. Bruckstein. Cooperative Covering by Ant-Robots using Evaporating Traces. Technical Report #CIS9610, CS Dept. Technion, 1996.

[10] S. Sastry, G. Meyer, C. Tomlin, J. Lygeros, D. Godbole and G. Pappas. Hybrid Control in air traffic management systems. IEEE Conference on Decision and Control, p: 1478-1483, 1996.

[11] J. Fredslund and M. Mataric. Robot Formations Using Only Local Sensing and Control. IEEE International Symposium on Computational Intelligence in Robotics and Automation, p: 308-313, 2001.

[12] J. Sweeney, T.J. Brunette, Y. Yang and R. Grupen. Coordinated Teams of Reactive Mobile Platforms. Proceedings of the 2002 IEEE Conference on Robotics and Automation, Washington, D.C. May, 2002 2002 IEEE.

[13] J.-C. Latombe. Robot Motion Planning. Kluwer Academic Publishers. 1991.

[14] D. Estrin, L. Girod, G. Pottie and M. Srivastava. Instrumenting the world with wireless sensor networks. Proceedings of ICASP 2001, Salt Lake City, pp.2675-2678.

[14] A. Howard, M. Mataric and G. Sukhatme. An Incremental Self-Deployment Algorithm for Mobile Sensor Networks. Autonomous Robots, Special Issue on Intelligent Embedded Systems, 13(2), September 2002, pp.113-126.

[15] D. Payton, M. Daily, R. Estowski, M. Howard and C. Lee. Pheromone Robotics. Autonomous Robots, Special Issue on Intelligent Embedded Systems, 11(3):319-324, 2001.

[16] J. Košecká, C. Tomlin, G. Pappas and S. Sastry. Generation of Conflict Resolution Maneuvers for Air-Traffic management. IROS 97, vol. 3:3069-74.

[17] A. Masoud. Using hybrid vector-harmonic potential field for multi-robot, multi-target navigation in stationary environments. Proceedings of the IEEE Conference on Robotics and Automation, 1996, pp. 3564-3571.

[18] E. Rimon and D. Koditschek. Exact robot navigation function using artificial potential functions. IEEE Transactions on Robotics and Automation, vol. 8(5):501-518, 1992.