Social Potentials for Scalable Multi-Robot Formations*

Tucker Balch
The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15217

Maria Hybinette
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332

Abstract

Potential function approaches to robot navigation provide an elegant paradigm for expressing multiple constraints and goals in mobile robot navigation problems [9]. As an example, a simple reactive navigation strategy can be generated by combining repulsion from obstacles with attraction to a goal. Advantages of this approach can also be extended to multi-robot teams. In this paper we present a new class of potential functions for multiple robots that enables homogeneous large-scale robot teams to arrange themselves in geometric formations while navigating to a goal location through an obstacle field. The approach is inspired by the way molecules "snap" into place as they form crystals; the robots are drawn to particular "attachment sites" positioned with respect to other robots. We refer to these potential functions as "social potentials" because they are constructed with respect to other agents. Initial results, generated in simulation, illustrate the viability of the approach.

1 Introduction

The flocking, schooling and herding behaviors we see in nature benefit the animals that use them in various ways. Each animal in a herd, for instance, benefits by minimizing its encounters with predators [15]. Formation is also useful in group tasks where sensor assets are limited. Formations allow individual team members to concentrate their sensors across a portion of the environment, while their partners cover the rest. Air Force fighter pilots for instance, direct their visual and radar search responsibilities depending on their position in a formation [7]. Formation maintenance is applicable in many other domains such as search and rescue, agricultural coverage tasks and security patrols. To address a wide range of multi-robot tasks we seek a formation strategy that provides:

- **scalability:** the approach should easily scale to any number of agents,
- **locality:** the behaviors should depend only on the local sensors of each agent,
- **flexibility:** the behaviors should be flexible so as to support many formation shapes.

To provide these features we introduce a new behavior-based approach to robot formation-keeping. The new strategy is based loosely on the way molecules form crystals. From the point of view of each robot in the group, every other robot has several local "attachment sites" other robots may be attracted to. This type of attachment site geometry roughly corresponds to molecular covalent bonding [1]. Just as different crystal shapes result from different covalent bond geometries, robot formation shapes are influenced by the attachment site geometries employed. Figure 1 illustrates the four attachment site geometries examined in this work. An example of two robots using the technique to move into formation is provided in Figure 2.

The overall behavior of the robots is determined through superposition of several potential functions coded as motor schemas [9, 2] (e.g. **avoid_obstacle**, **move_to_goal** and so on). The formation component of behavior depends on the locations of other nearby robots. We refer to this type of potential function, constructed with regard to other robots, as a "social potential" to distinguish it from other types of functions for robot navigation (e.g. repulsion from obstacles).

1.1 Related work

An early application of artificial formation behavior was the behavioral simulation of flocks of birds and schools of fish for computer graphics. Results in this area originated in Craig Reynolds work [14]. He developed a simple egocentric behavioral model for flocking which is instantiated in each member of the simulated group of birds (or "boids"). A contribution of

*ICRA-2000, to appear

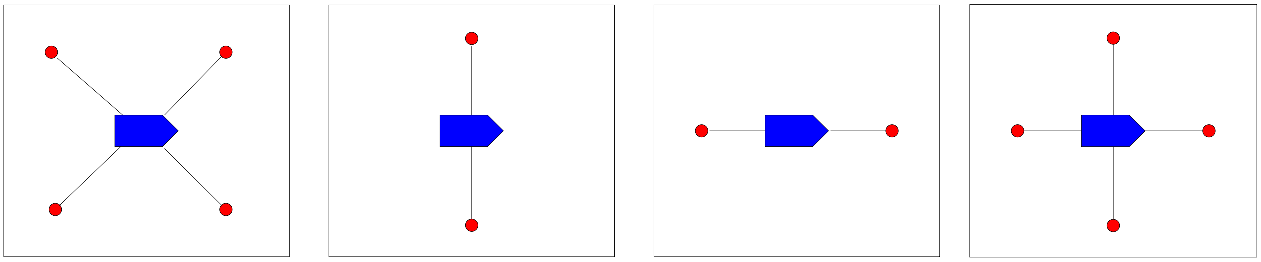

Figure 1: From the point of view of each robot in the group, every other robot has several local "attachment sites" other robots may be attracted to. Attachment site geometries for different formations are illustrated above. From left to right: *diamond*, *line*, *column* and *square*. Robots are represented as five-sided polygons moving from left to right; attachment sites are shown with small circles.

Reynold's work is the generation of successful overall group behavior while individual agents only sense their local environment and close neighbors.

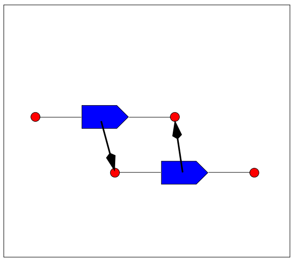

Figure 2: Example of how agents are attracted to the "attachment sites" of other robots. In this example, two robots using a *column* attachment site geometry move into position.

The components of Reynolds' flocking behaviors are similar in philosophy to the motor schema paradigm used here, but his approach is concerned with the generation of visually realistic flocks and herds for large numbers of simulated animals, a different problem domain than the one this research addresses. In contrast, our research studies the problem of organizing robots in specific geometric arrangements.

The dynamics and stability of multi-robot formations have drawn recent attention [16, 6]. Their research centers on the analysis of group dynamics and stability, and does not provide for obstacle avoidance. In the approach forwarded in this article however, geometric formations are specified in a similar manner, but formation behaviors are fully integrated with obstacle avoidance and other navigation behaviors.

Other recent related papers on formation control for robot teams include [10, 8, 13, 18, 17]. Mataric's work shows that simple behaviors like avoidance, aggregation and dispersion can be combined to create an emergent flocking behavior in groups of wheeled robots [10]. Parker's thesis [13] concerns the coordination of multiple heterogeneous robots. Of particular interest is Parker's work in implementing "bounding overwatch," a military movement technique for teams of agents; one group moves (bounds) a short distance, while the other group overwatches for danger. Yoshida [18], and separately, Yamaguchi [17], investigate how robots can use only local communication to generate a global grouping behavior. Similarly, Gage [8] examines how robots can use local sensing to achieve group objectives like coverage and formation maintenance.

In the work most closely related to this research, Parker simulates robots in a line-abreast formation navigating past waypoints to a final destination [12]. The approach includes a provision for obstacle avoidance, but performance in the presence of obstacles is not reported. Parker's results suggest that performance is improved when agents combine local control with information about the leader's path and the team's goal. This research is distinguished from Parker's in that we are concerned with supporting many types of formation geometries.

In earlier work we presented a formation strategy for teams of up to four unmanned ground vehicles (UGVs) intended to be fielded as a scout unit by the U.S. Army [5]. Contributions of this earlier work include behaviors for four-robot *diamond*, *line*, *column*, and *wedge* formations and a performance analysis of each formation type in turns and across obstacle-strewn terrain. The approach has been demonstrated on laboratory robots and on three UGVs in the Army's UGV Demo II program. The technique is still incorporated in the ongoing UGV Demo III program [11].

The earlier technique works well, but has several limitations. First, the approach only supports formations for two to four robots. Extending it to larger groups of agents is possible, but requires the generation of a template for each number of robots and each formation geometry. Second, each robot has a specifically designated position in the formation. In some situations robots must cross each others' path to position themselves correctly in the formation. This is appropriate for some applications, but in general it is probably more efficient for the closest robot to fill any given position. Both of these problems are addressed by the new technique presented here.

2 Behaviors for formation

2.1 Overview

The formation behaviors are implemented as *motor schemas* using the Clay library [2, 3]. Each component of the task (e.g. move to the goal, avoid obstacles) is coded as a separate process (schema) that outputs a vector indicating which direction the robot should travel. Each vector's magnitude indicates the relative importance the associated schema (this may vary over time). The resultant vectors are summed for final output to command the robot's movement. The approach is similar to potential field navigation initially proposed by Khatib [9].

The formation component of the robots' behavior is accomplished in two steps: first, a perceptual process, **detect_formation_position**, determines the robot's proper position in formation based on current sensor data; second, the motor process **maintain_formation**, generates motor commands to direct the robot toward the correct location. The motor schema paradigm enables the formation behavior to be simultaneously active in combination with other navigation behaviors.

The overall navigational strategy integrates the formation behavior with other navigational schemas in a manner similar to the approach developed in earlier research [5]. The motor schemas **move_to_goal**, **avoid_static_obstacles**, **avoid_robots** and **maintain_formation** implement the overall behavior for a robot to move to a goal location while avoiding obstacles, collisions with other robots and remaining in formation. An additional background schema, **noise**, serves as a form of reactive "grease", dealing with some of the problems endemic to purely reactive navigational methods such as local maxima, minima and cyclic behavior [2].

The key extension that distinguishes the new formation behavior from previous work is the perceptual technique used to determine the proper formation position for each robot. Instead of having each agent assigned to a particular position as in the previous approach there are a number of available locations for each robot in the formation.

Each attachment site geometry is characterized by three parameters:

- r, the distance from the center of the robot to each attachment site,
- N, the number of sites available, and
- θ, the offset in degrees with respect to the front of the robot (straight ahead) where the first site is positioned.

We assume the N sites are positioned uniformly around each robot. In the example geometries presented here, $r = 1.5$ meters in all cases; $N = 2$ for *column* and *line* geometries and $N = 4$ for *diamond* and *square*; $\theta = 0^o$ for *column* and *square*, $\theta = 45^o$ for *diamond* and $\theta = 90^o$ for *line* formations.

To determine a formation position each robot builds a list of all potential attachment sites for all of the robots within sensor range based on the formation type it is using. An attractive vector is generated towards the closest site.

In addition to the motor schemas mentioned earlier, a low-gain attractive force, **move_to_unit_center**, is added to draw all of the robots together. As the team converges, the robots "snap" into position, and a regular geometric shape emerges (Figure 2). Example formations resulting from the integration of these behaviors are illustrated in Figure 6.

Note that for the *diamond* and *square* attachment site geometries there are many possible robot team arrangements. It is also possible that interaction with obstacles will "unsnap" the formation into smaller sub-formations. In most cases however, the formations re-group after splitting around obstacles.

2.2 Computational details

At each movement step, each motor schema computes a vector. Each vector is multiplied by a gain value indicating the relative importance of the associated schema. The resulting vectors are then summed to compute the overall movement direction. The parameters and gains for the motor schemas used in this work are summarized in Table 1. Output vectors for the motor schemas are computed as follows:

motor schema	gain values
avoid_static_obstacles $S = 2.0\text{m}$ $M = 0.1$	1.1
avoid_robots $S = 2.0\text{m}$ $M = 0.1\text{m}$	1.1
move_to_goal $C = 0.0\text{m}$, $D = 0.0\text{m}$	0.7
maintain_formation $C = 1.0\text{m}$ $D = 0.0\text{m}$	1.3
move_to_unit_center $C = 3.0\text{m}$, $D = 2.0\text{m}$	0.6
noise $P = 5.0\text{sec}$	0.1

Table 1: Motor schema parameter values and gains used in formation experiments.

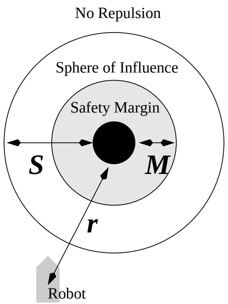

Figure 3: Parameters used in the calculation of **avoid** motor schema vectors. The repulsive potential increases as the robot moves closer to the obstacle. The object to be avoided is represented as a black circle at the center of the illustration.

• **avoid_static_obstacles**: repulsion from detected obstacles. The magnitude of repulsion varies with distance from each obstacle (Figure 3). When beyond the sphere of influence (S), no repulsion is generated. Within the sphere of influence, repulsion increases linearly until the robot reaches the safety margin. When the robot is within the safety margin, the magnitude of repulsion is ∞. The behavior is parameterized by S, the sphere of influence beyond which detected obstacles have no effect and M, the safety margin. A separate vector is computed for each detected obstacle as follows, where r is the distance from the center of the robot to closest point on the obstacle:

$$V_{\text{direction}} = \begin{array}{l}\text{along a line from the center of the}\\ \text{obstacle to the robot, moving}\\ \text{moving away from obstacle}\end{array}$$

$$V_{\text{magnitude}} = \begin{cases} 0 & \text{for } r > S \\ \frac{S-r}{S-M} & \text{for } M < r \leq S \\ \infty & \text{for } r \leq M \end{cases}$$

The overall **avoid_static_obstacles** vector is computed by summing the individual vectors calculated for each obstacle.

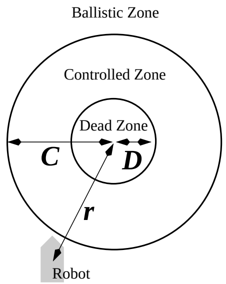

Figure 4: Parameters used in the calculation of **maintain_formation** and **move_to_goal** motor schema vectors. The attractive potential decreases as the robot moves into position.

• **maintain_formation**: an attractive force to draw the robot into the proper formation position. The magnitude of the vector varies with distance from the formation position. Figure 4 illustrates three zones, defined by distance from the position, used for magnitude computation. The radii of these zones are parameters of the schema. Outside the *controlled zone* attraction is set at a fixed maximum (1.0). Within the controlled zone attraction decreases linearly from 1.0 to 0.0 at the boundary of the *dead zone*. Inside the dead zone the magnitude is 0.0. The schema is parameterized by C, the radius of the controlled zone; and D, the radius of the dead zone. The vector is computed as follows, where r is the distance from the center of the robot to the goal location:

$$V_{\text{direction}} = \begin{array}{l}\text{along a line from the robot}\\ \text{to the goal, moving to the goal}\end{array}$$

$$V_{\text{magnitude}} = \begin{cases} 1 & \text{for } r > C \\ \frac{r-D}{C-D} & \text{for } D < r \leq C \\ 0 & \text{for } r \leq D \end{cases}$$

• **move_to_goal**: attractive force to draw the robot to a goal location. The goal is positioned 1000 meters beyond the finish line. The robots never actually reach the goal in experimental trials because each trial

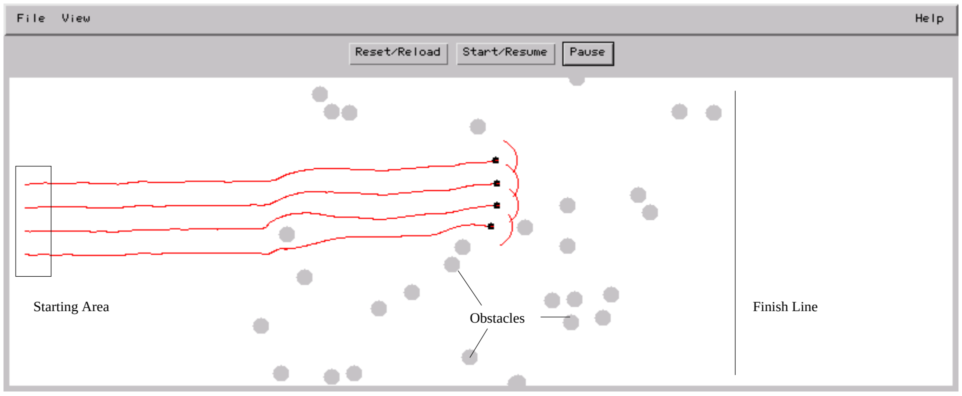

Figure 5: The simulation environment used in experiments. Robots (simulated Nomadic Technologies' Nomad-150 robots) are initialized on the left. They navigate from left to right through the obstacles. Performance is measured as time to cross the finish line.

terminates when the robots cross the finish line. This schema is computed in the same manner as **maintain_formation** but with respect to the goal location rather than a formation position.

• **move_to_unit_center**: a low-gain attractive force, added to draw all of the robots together. Computed in the same manner as **move_to_goal** and **maintain_formation** but with regard to the averaged locations of all other robots in sensor range.

• **avoid_robots**: repulsion from detected robots. This schema's output is computed in exactly the same manner as **avoid_obstacle** except with respect to robots instead of fixed obstacles.

• **noise:** generates movement in a pseudo-random direction. Parameterized by P, persistence, the time in seconds between each change in direction. The vector is computed as follows:

$$\begin{aligned} V_{\text{direction}} &= \text{pseudo-random direction} \\ &\quad\ \text{between } 0 \text{ and } 2\pi \\ V_{\text{magnitude}} &= 1 \end{aligned}$$

3 Simulation environment and performance measurement

The task examined in these experiments is for a team of robots to move across a field as quickly as possible while maintaining a geometric formation and avoiding collisions with obstacles and other robots. To enable comparative evaluation of the various formation strategies presented above, we specify performance in terms of the time for the *entire* team of robots to move across the field. This is equivalent to the performance of the last agent to cross the field. This metric was chosen because it indicates, to some degree, the extent of cooperation between the robots. Other measures might show improved performance when individual agents "abandon" their partners in an effort to cross the finish line more rapidly.

Figure 5 illustrates the *TeamBots* simulation environment used in the experiments. The simulated field measures 20m by 60m. 30 obstacles, each 1m^2 in area, are distributed randomly about a 20 by 30 meter zone in the middle of the field (5% obstacle coverage). The robots are initialized on the left side of the field. They then navigate to the right side, through the obstacles to the finish line on the right. Timing stops when the last robot crosses the line. The agents are initialized line abreast on the left side of the field. This initial configuration was chosen because it ensures all robots are equidistant from the finish line. The first 20m of the field are clear of obstacles to enable the robots to settle into formation positions before encountering the obstacle field. After crossing the obstacle-free section the robots encounter a 30m long zone cluttered with hazards.

Two aspects of the experimental setup should be considered when reviewing performance results. First, the arrangement of agents at the beginning of each run may bias the shape of the formation towards line abreast. Second, the measured time to complete the task includes the time taken for the agents to cross the initial, obstacle-free area. Thus overall performance is a combination of performance in obstacle-free and cluttered terrain.

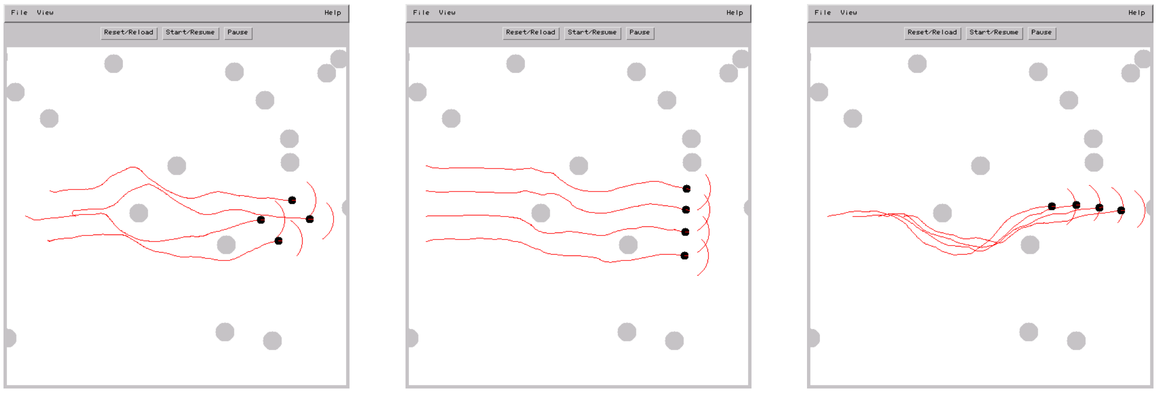

Figure 6: Example four-robot formations resulting from the use of different attachment site geometries. From left to right: *column*, *line* and *diamond*. In each of these short demonstration runs the robots were initialized in proper formation positions, experimental runs are over a longer course.

4 Results

4.1 Geometries and scalability

As illustrated in Figure 6 the formation behaviors enable robot teams to maintain formation while navigating through an obstacle field. As expected, different attachment site geometries lead to different team formation geometries.

Additionally these behaviors are easily scalable. As an example, consider the team of 32 robots illustrated in Figure 7.

4.2 Performance

To evaluate the relative performance of the various strategies, experiments were run in simulation with one to eight robots using *diamond*, *line* and *column* formation geometries.

In addition to three formation strategies, we also compared the performance of a robot team using no formation. This provides a benchmark to evaluate whether robot teams benefit from the formation behaviors. The *no_formation* strategy utilizes the same navigational behaviors as in the other strategies except **maintain_formation** is not activated. The group of robots are still attracted to one another because the **move_to_unit_center** motor schema is activated.

Performance was evaluated by running each simulated robot team through each of five different randomly generated worlds 50 times. A total of 250 simulations were run for each number of robots for each formation geometry, or a total of 8000 trials overall. The average time for robots to complete the traverse is plotted for each strategy in Figure 8.

The relative performance of teams using *diamond*, *line* and *column* geometries mirrors similar results reported earlier [4]. As was the case in the earlier experiments in team navigation across an obstacle field we find that the *column* formation strategy provides the best performance. In the column formation the team as a whole presents a smaller cross section to the obstacles as it moves across the field. The *line* formation performs worst because it presents the broadest cross section.

It is interesting to note that for 1 to 7 robots not only does the *column* strategy offer better performance than other formation strategies, it also yields better performance than *no_formation*. This indicates that for small teams of robots, this strategy provides a cooperative benefit to the team.

The performance of teams using *no_formation* improves consistently as the number of robots increases. Eventually, with groups of 8 robots, performance is slightly better for the teams using *no_formation*. This is probably because in all strategies except *no_formation*, when a robot is slowed as it encounters an obstacle, other robots are likely to remain near it and slow down also. In the *no_formation* strategy, the low-gain **move_to_unit_center** behavior slows progress of the other agents, but it will not stop them. In addition the **move_to_unit_center** behavior provides the side-effect of pulling "stragglers" out of the areas they may be stuck in.

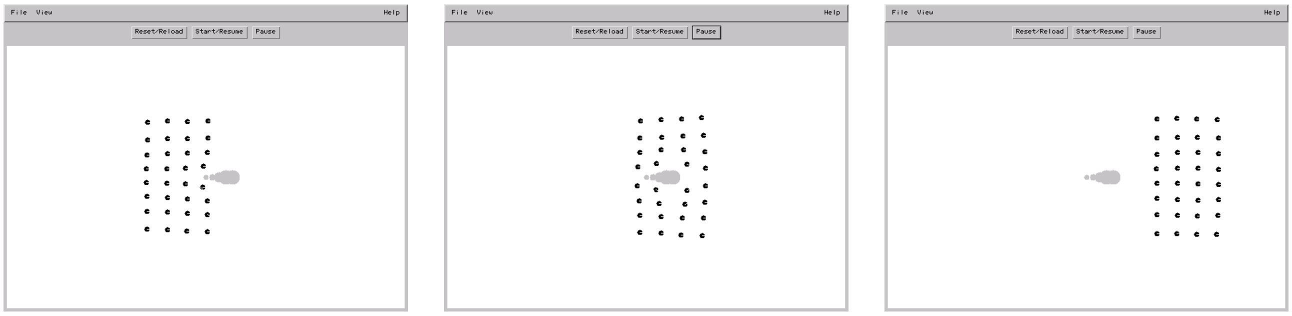

Figure 7: Example of a large-scale formation of 32 robots using the *square* attachment site geometry. The robots (black circles) start on the left side of the field and navigate to the right around a group of obstacles in the middle of the field. Note how the formation splits around the obstacle, but rejoins once past it.

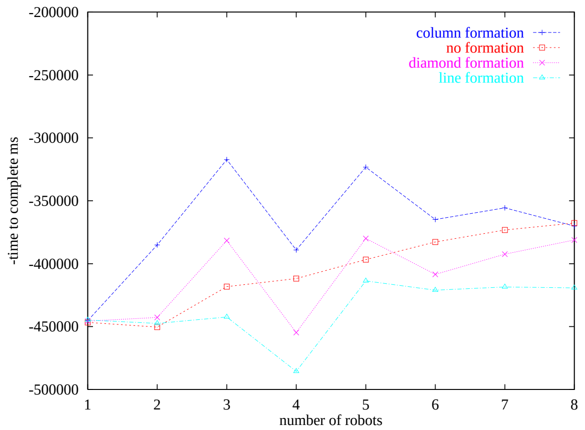

Figure 8: Average performance for navigating teams using different formation strategies.

5 Conclusion

A new behavior-based approach for scalable multi-robot formations was presented. The key extension that distinguishes the new approach from previous work is the perceptual technique used to determine the proper formation position for each robot. Individual robots are not assigned to particular locations but are instead attracted to the closest position in the formation. The approach is based loosely on molecular crystal formation: each robot is drawn to "attachment sites" arranged with respect to its teammates. The resulting robot team geometry is determined by the arrangement of the attachment sites.

The design goals for the new strategy are met; specifically:

- **scalability:** the approach easily scales to any number of agents,
- **locality:** the behaviors depend only on the local sensors of each agent,
- **flexibility:** the behaviors are flexible so as to support many formation shapes.

Simulation experiments illustrate the approach and demonstrate the relative performance of several formation geometries. Performance was evaluated for groups of 1 to 8 robots using each of three different formation geometries. The results confirm earlier work that indicates column formations are best for traversing an obstacle field [5].

The approach is scalable because each agent only relies on locally available information; namely, the locations of nearby robots. Global communication of robot position is not required, instead, local sensors (perhaps vision) can be utilized to generate effective formation behavior in large robot teams. Scalability of the approach is demonstrated in a large team composed of 32 simulated robots.

5.1 Future work

It is important to realize that for some attachment site geometries there are multiple arrangements of the robot team. For instance, using the *square* geometry it is possible to arrange four robots in a stable column, line or square. This was observed in simulation experiments where a team formation would sometimes "snap" apart to move around obstacles and then rejoin into a different overall shape. We plan to extend the work to account for this and to enable the user to specify an overall desired formation shape.

The *TeamBots* simulation environment that was used to generate the results in this paper has also been used extensively in earlier work to prototype behaviors for Nomadic Technologies Nomad-150 robots [3]. The *TeamBots* environment enables the same behaviors to run in simulation and on mobile robots. We have found good correspondence between the performance of control systems running in the *TeamBots* simulator and their behavior on real robots. Even so, it is important to verify performance on mobile robots. To support experiments on a large numbers of robots, we are in the process of building a team of 10-20 Cye robots (manufactured by Probotics, Inc.). The *TeamBots* environment is also being modified to support this robot platform (Figure 9). More information on this project is available online at `http://www.cs.cmu.edu/~coral/minnow`.

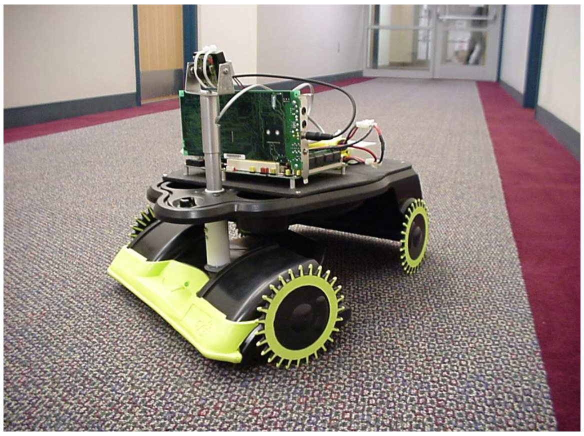

Figure 9: We are planning future experiments on a team of 10-20 Cye robots.

Acknowledgments

Thanks to Jonathan Cameron, Ron Arkin, Manuela Veloso, Matt Mason and Reid Simmons for helpful comments on the ideas presented in this paper.

References

[1] Chemical bonding. In *Encyclopaedia Britannica*. 1999.

[2] R. Arkin and T. Balch. AuRA: principles and practice in review. *Journal of Experimental and Theoretical Artificial Intelligence*, 9(2), 1997.

[3] T. Balch. *Behavioral Diversity in Learning Robot Teams*. PhD thesis, College of Computing, Georgia Institute of Technology, 1998.

[4] T. Balch and R.C. Arkin. Motor schema-based formation control for multiagent robot teams. In *First International Conference on Multiagent Systems*, pages 10–16, June 1995. San Francisco.

[5] T. Balch and R.C. Arkin. Behavior-based formation control for multiagent robot teams. *IEEE Transactions on Robots and Automation*, 1999.

[6] Q. Chen and J. Y. S. Luh. Coordination and control of a group of small mobile robots. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 2315–2320, San Diego, CA, USA, 1994.

[7] U.S. Air Force. *Air Combat Command Manual 3-3*. Department of the Air Force, Washington, D.C., 1992.

[8] D.W. Gage. Command control for many-robot systems. *Unmanned Systems Magazine*, 10(4):28–34, 1992.

[9] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. In *Proc. IEEE Int. Conf. Robotics and Automation*, page 500, 1985.

[10] M. Mataric. Designing emergent behaviors: From local interactions to collective intelligence. In *Proceedings of the International Conference on Simulation of Adaptive Behavior: From Animals to Animats 2*, pages 432–441, 1992.

[11] M Morganthaler. Personal communication. SAIC, April 1999.

[12] L. Parker. Designing control laws for cooperative agent teams. In *Proceedings of the 1993 IEEE International Conference on Robotics and Automation*, pages 582–587. IEEE, 1993.

[13] Lynne E. Parker. *Heterogeneous Multi-Robot Cooperation*. PhD thesis, M.I.T. Department of Electrical Engineering and Computer Science, 1994.

[14] C Reynolds. Flocks, herds and schools: A distributed behavioral model. *Computer Graphics*, 21(4):25–34, 1987.

[15] S. L. Veherencamp. Individual, kin, and group selection. In P. Marler and J.G. Vandenbergh, editors, *Handbook of Behavioral Neurobiology, Volume 3: Social Behavior and Communication*, pages 354–382. Plenum Press, New York, 1987.

[16] P.K.C. Wang. Navigation strategies for multiple autonomous robots moving in formation. *Journal of Robotic Systems*, 8(2):177:195, 1991.

[17] H. Yamaguchi. Adaptive formation control for distributed autonomous mobile robot groups. In *Proceedings of the 1997 IEEE Conference on Robotics and Automation*, April 1997. Albuquerque, NM.

[18] E. Yoshida, T. Arai, J. Ota, and T. Miki. Effect of grouping in local communication system of multiple mobile robots. In *Proceedings of the 1994 IEEE International Conference on Intelligent Robots and Systems*, pages 808–815, Munich, Germany, 1994.