Motor Schema-based Formation Control for Multiagent Robot Teams *

Tucker Balch and **Ronald C. Arkin**
Mobile Robot Laboratory
College of Computing
Georgia Institute of Technology
Atlanta, Georgia 30332-0280

Abstract

New reactive behaviors that implement formations in multi-robot teams are presented and evaluated. These motor schemas, or primitive behaviors, for relative positional maintenance are integrated with existing navigational behaviors to help robots complete navigational tasks while in formation. Four formations, based on existing military doctrine (Army 1986), and three methods for determining correct vehicle position are investigated. The performance of a group of four simulated robots using this technique is evaluated quantitatively for both turning and for navigation across an obstacle field. These team behaviors will ultimately be fielded on four military vehicles as part of ARPA's UGV Demo II program.

Introduction

This research concerns the development of behaviors for formation maintenance in heterogeneous societies of mobile robots. The target platform is a set of four robotic vehicles to be employed as a scout unit by the U.S. Army as part of ARPA's UGV Demo II program. Formations are important in this application as they allow the unit to utilize its sensor assets more efficiently than if the robots are arranged randomly.

The robots in this work are heterogeneous in that each is assigned a unique identification number (ID). A robot's designated position in a given formation depends upon its ID. There are no other behavioral differences between the robots.

Formation control is one part of a more complex behavioral assemblage which includes other components for high-level task achievement. In addition to maintaining their position in formation, robots must also move to a specified goal location while avoiding collisions with obstacles.

The formation behaviors presented here are implemented as *motor schemas*, a type of reactive navigational strategy (Arkin 1989). Motor schemas operate as concurrent asynchronous processes each of which instantiates a high-level behavioral intention. Perception is translated into a response vector for each active behavior. The resulting responses are summed and normalized and then output for execution by the mobile vehicle. Readers desiring additional information on schema-based reactive control are referred to (Arkin 1989).

Background

Formation behaviors in nature, like flocking and schooling, benefit the animals that use them in various ways. Each animal in a herd, for instance, benefits by minimizing its encounters with predators (Veherencamp 1987). By grouping, animals also combine their sensors to maximize the chance of detecting predators and to more efficiently forage for food. Studies of flocking and schooling show that these behaviors emerge as a combination of a desire to stay in the group and yet simultaneously keep a separation distance from other members of the group (Cullen, Shaw, & Baldwin 1965). Since groups of artificial agents could similarly benefit from formation tactics, robotics researchers and those in the artificial life community have drawn from these biological studies to develop formation behaviors for both simulated agents and robots. A brief review of a few of these efforts follows. Approaches to formation generation in robots may be distinguished by their sensing requirements, their method of behavioral integration, and their commitment to pre-planning.

An early application of artificial formation behavior was the behavioral simulation of flocks of birds and schools of fish for computer graphics. Important results in this area originated in Craig Reynolds pioneering work (Reynolds 1987). He developed a simple egocentric behavioral model for flocking which is instantiated in each member of the simulated group of birds (or "boids"). The behavior consists of several separate components, including: inter-agent collision avoidance, velocity matching and flock centering. Each of the components is computed separately, then combined for movement output. An important contribution of Reynold's work is the generation of successful overall group behavior while individual agents

*To appear in Proceedings of the First International Conference on Multiagent Systems, San Francisco.

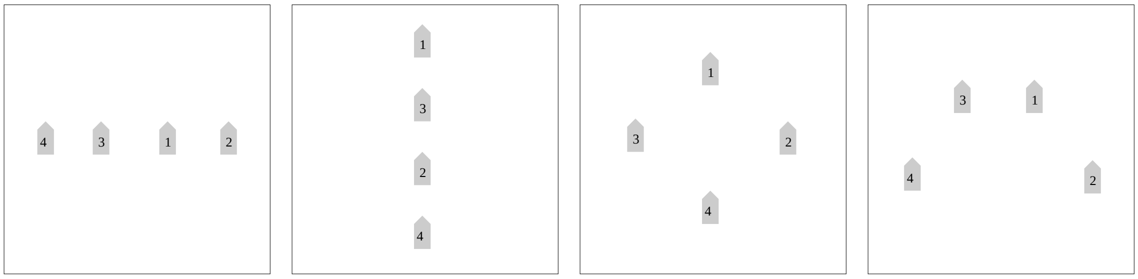

Figure 1: Formations for four robots (from left to right: line, column, diamond, wedge)

only sense their local environment and close neighbors. Hodgins and Brogan (Hodgins & Brogan 1994) have recently extended this work for artificial herds of one-legged animals. Their results are even more visually realistic than Reynolds' because they also simulate the mechanics of motion; Reynolds' approach utilized particle models only.

The individual components of Reynolds' flocking and Hodgins' herding behaviors are similar in philosophy to the motor schema paradigm used here, but their approaches are concerned with the generation of visually realistic flocks and herds for large numbers of simulated animals, a different problem domain than the one this paper addresses. In contrast, our research studies behaviors for a small group (four) of mobile robots, striving to maintain a specific geometric formation.

The dynamics and stability of multi-robot formations has drawn recent attention (Wang 1991; Chen & Luh 1994). Wang (Wang 1991) developed a strategy for robot formations where individual robots are given specific positions to maintain relative to a leader or neighbor. Sensory requirements for these robots are reduced since they only need to know about a few other robots. His analysis centered on feedback control for formation maintenance and stability of the resulting system. It did not include integrative strategies for obstacle avoidance and navigation. In work by Chen and Luh (Chen & Luh 1994) formation generation by distributed control is demonstrated. Large groups of robots are shown to cooperatively move in various geometric formations. This research also centered on the analysis of group dynamics and stability, and does not provide for obstacle avoidance. In the approach forwarded in this paper, geometric formations are specified in a similar manner, but formation behaviors are fully integrated with obstacle avoidance and other navigation behaviors.

Mataric has also investigated emergent group behavior (Mataric 1992b; 1992a). Her work shows that simple behaviors like avoidance, aggregation and dispersion can be combined to create an emergent flocking behavior on wheeled robots. Her research is in the vein of Reynolds' work in that a specific agent's geometric position is not designated. The behaviors described in this paper differ in that positions relative to the group are specified and maintained for each active robot.

Other recent related papers on formation control for robot teams include (Parker 1994; Yoshida, Arai, & Yomoyoshi 1994; Gage 1992). Parker's thesis (Parker 1994) concerns the coordination of multiple heterogeneous robots. Of particular interest is her 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 (Yoshida, Arai, & Yomoyoshi 1994) investigates how robots can use only local communication to generate a global grouping behavior. Similarly, Gage (Gage 1992) examines how robots can use local sensing to achieve group objectives like coverage and formation maintenance.

Approach

Several formations for a team of four robots are considered (Fig. 1):

- *line* - where the robots travel line-abreast.
- *column* - where the robots travel one after the other.
- *diamond* - where the robots travel in a diamond.
- *wedge* - where the robots travel in a "V".

These particular formations were selected because they are used by mechanized scout platoons on the battlefield (Army 1986). For each formation, each robot has a specific position (based on its ID). Figure 1 shows the formations and robots' positions within them. Active behaviors for each of the four robots are identical, except in the case of Robot 1 in leader-referenced formations (see below).

In this research, the task for each robot is to simultaneously move to a goal location, avoid obstacles, avoid colliding with other robots and maintain a formation position (typically in the context of a higher-level mission scenario). The motor schemas, **move-to-goal**, **avoid-static-obstacle**, **avoid-robot**, and **maintain-formation**, respectively, implement these behaviors. An additional background schema, **noise**, serves as a form of reactive "grease", dealing with some of the problems endemic to purely reactive navigational methods (Arkin 1989). Each schema generates

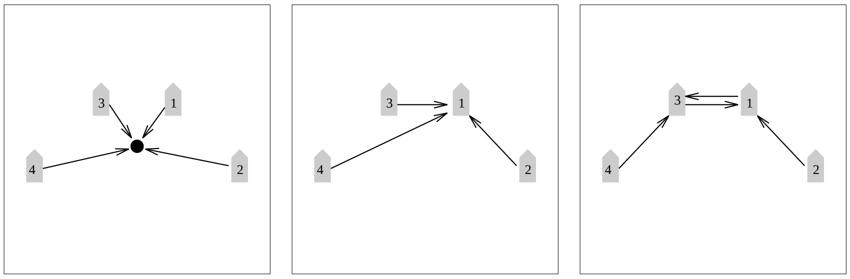

Figure 2: Formation position determined by various reference techniques (from left to right: unit-center, leader, neighbor)

a vector representing the desired behavioral response (direction and magnitude of movement) for the robot given the current sensory stimuli provided by the environment. A gain value is used to indicate the relative importance of the individual behaviors. The high-level combined behavior is generated by multiplying the outputs of each primitive behavior by its gain, then summing and normalizing the results. The gains and other schema parameters used for the results presented in this paper are listed in Table 1. See (Arkin 1989) for a complete discussion of the computational basis and parameter definitions for the **avoid-static-obstacle**, **move-to-goal** and **noise** motor schemas.

Parameter	Value	Units
avoid-static-obstacle		
gain	1.5	
sphere of influence	50	meters
minimum range	5	meters
avoid-robot		
gain	2.0	
sphere of influence	20	meters
minimum range	5	meters
move-to-goal		
gain	0.8	
noise		
gain	0.1	
persistence	6	steps
maintain-formation		
gain	1.0	
desired spacing	50	meters
controlled zone radius	25	meters
dead zone radius	0	meters

Table 1: Motor schema parameters for navigation and formation.

Formation maintenance is accomplished in two steps: first, the perceptual process, **detect-formation-position**, determines the robot's proper position in formation based on current environmental data; second, the motor schema **maintain-formation** generates a movement vector toward the correct location. Each robot must compute its proper position in the formation for each movement step. Three techniques for formation position determination have been identified:

- **Unit-center-referenced**: a unit-center is computed by averaging the x and y positions of all the robots involved in the formation. Each robot determines its own formation position relative to that center.
- **Leader-referenced**: each robot determines its formation position in relation to the lead robot (Robot 1). The leader does not attempt to maintain formation; the other robots are responsible for formation maintenance.
- **Neighbor-referenced**: each robot maintains a position relative to one other predetermined robot.

These relationships are depicted in Figure 2. Arrows show how the formation positions are determined. Each arrow points *from* a robot *to* the associated reference. The perceptual schema **detect-formation-position** uses one of these references to determine the proper position for the robot. The spacing between robots is determined by the *desired spacing* parameter of **detect-formation-position**.

Once the desired formation position is known, the **maintain-formation** motor schema generates a movement vector towards it. The vector is always in the direction of the desired formation position, but the magnitude depends on how far the robot is from the desired position. Figure 3 illustrates three zones, defined by distance from the desired position, used for magnitude computation. The radii of these zones are parameters of the **maintain-formation** schema. In the example, Robot 3 attempts to maintain a position to the left of and abeam Robot 1. Robot 3 is in the controlled zone, so a moderate force towards the desired position (forward and right) is generated by **maintain-formation**. The magnitude of the vector is computed as follows:

- **Ballistic zone**: the magnitude is set at its maximum, which equates to the schema's gain value.
- **Controlled zone**: the magnitude varies linearly from a maximum at the farthest edge of the zone to zero at the inner edge.

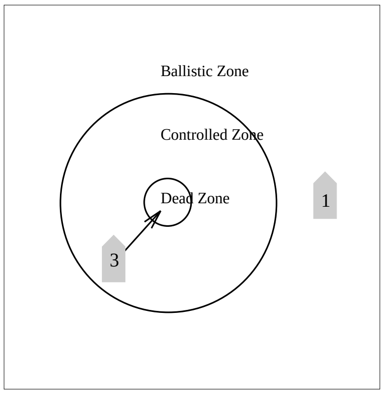

Figure 3: Zones for the computation of **maintain-formation** magnitude

- **Dead zone**: in the dead zone vector magnitude is always zero.

Results

Simulation Environment

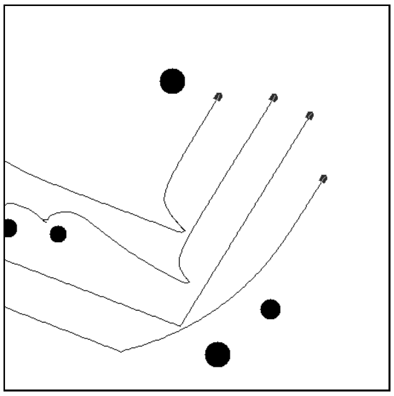

Figure 4: Typical simulation run showing four robots in a leader-referenced wedge formation executing a 90 degree left turn.

Results were generated using Georgia Tech's MissionLab robot simulation environment (MacKenzie, Cameron, & Arkin 1995). MissionLab runs on Sun SPARCs under SunOS and the X11 graphical windowing system. The simulation environment is a 1000 by 1000 meter two dimensional "playing field" upon which various sizes and distributions of circular obstacles can be scattered. Each simulated robot is a separately running C program that interacts with the simulation environment via a Unix socket. The simulation displays the environment graphically and maintains world state information which it transmits to the robots as they request it. Figure 4 shows a typical simulation run. The robots are displayed as five-sided polygons, while the obstacles are black circles. The robots' paths are depicted with solid lines.

Sensors allow a robot to distinguish between three perceptual classes: robots, obstacles and goals. The robots' sensors and actuators are implemented in the main simulation. When one of the robot's perceptual processes requires obstacle information for example, a request for that data is sent via the socket to the simulation. A list comprised of angle and range data for each obstacle in sensor range is returned. Robot and goal sensor information is provided similarly. A robot moves by transmitting its desired velocity to the simulation process. The simulation process automatically maintains the position and heading of each robot.

Qualitative Results

The *line*, *column*, *wedge* and *diamond* formations have been implemented using the unit-center-referenced and leader-referenced approaches. Neighbor-referenced formations are under development.

Figure 5 illustrates robots moving in each of the basic formations using the leader-referenced approach. In each of these simulation runs the robots were first initialized on the left side of the simulation environment, then directed to proceed to the lower center of the frame. After the formation was established, a 90 degree turn to the left was initiated. Results were similarly obtained for the unit-center-referenced formations.

Qualitative differences between the two approaches can be seen as a formation of robots moves around obstacles and through turns (see Figure 6). For leader-referenced formations any turn by the leader causes the entire formation to shift accordingly, but when a "follower" robot turns, the others in formation are not affected. In unit-center-referenced formations any robot move or turn impacts the entire formation. In turns for leader-referenced formations, the leader simply heads in the new direction; other robots must adjust to move into position. In unit-center-referenced turns, the entire formation initially appears to spin about a central point, as the robots align with a new heading.

Quantitative Results

To investigate quantitative differences in performance between the various formation types and references, two experiments were conducted in simulation: the first experiment evaluates performance in turns, and the second evaluates performance across an obstacle field.

To evaluate performance in turns, the robots are commanded to travel 250 meters, turn right, then travel another 250 meters. The robots attempt to maintain formation throughout the test. A turn of 90 degrees was selected for this initial study, but performance likely varies for different angles. In this evaluation, no obstacles are present. For statistical significance, 10 simulations were run for each formation type and reference. To ensure the robots are in correct formation at the start of the evaluation, they travel 100

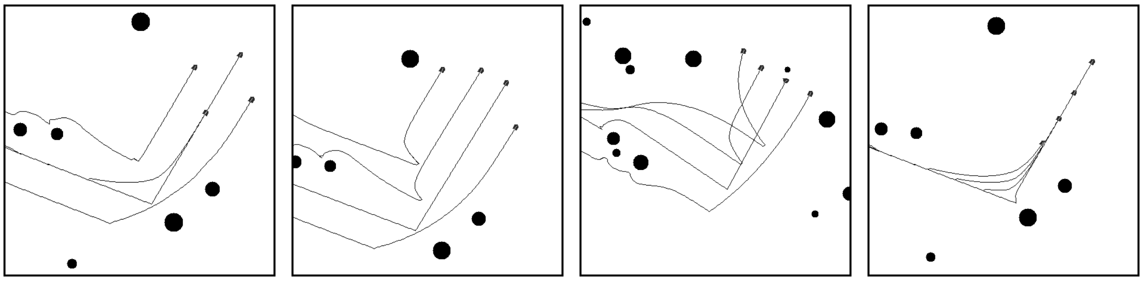

Figure 5: Four robots in leader-referenced *diamond*, *wedge*, *line* and *column* formations.

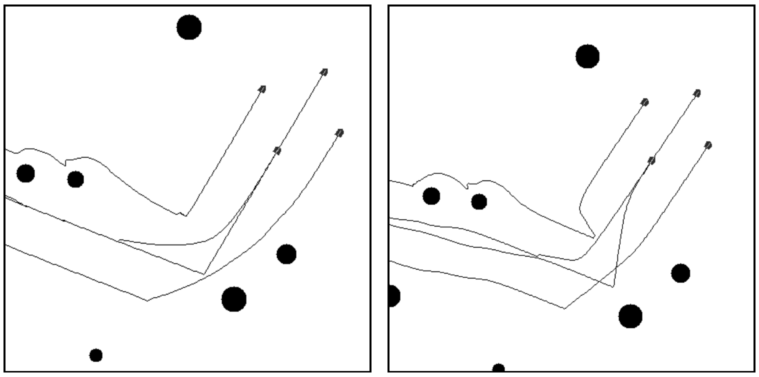

Figure 6: Comparison of leader-referenced (left) and unit-center-referenced (right) diamond formations.

meters to align themselves before the evaluation starts. This initial 100 meters is not included in the 500 meter course. An evaluation run is complete when the unit-center of the formation is within 10 meters of the goal location. Even though a unit-center computation is used to determine task completion, it is not required for leader-referenced formation maintenance.

Three performance metrics are employed: path length ratio, average position error, and percent of time out of formation. Path length ratio is the average distance traveled by the four robots divided by the straight-line distance of the course. A lower value for this ratio indicates better performance. A ratio of 1.02, for example, means the robots had to travel an average of 2% further because they were in formation. Position error is the average displacement from the correct formation position throughout the run. Robots occasionally fall out of position due to turns, etc.; this is reflected in the percent of time out of formation data. To be "in position" a robot must be within 5 meters of its correct position. 5 meters was selected arbitrarily, but amounts to 10% of the overall formation spacing. Results for the turn experiment are summarized in Table 2; the standard deviation for each quantity is listed in parentheses.

Performance was also measured for four robots navigating across a field of obstacles in formation. In this evaluation, the robots are commanded to travel between two points 500 meters apart. Obstacles are placed randomly so that 2% of the total area is covered with obstacles 10 to 15 meters in diameter. Future work will investigate how various percentages of obstacle coverage impacts formation performance. As in the turn evaluation above, path length ratio, average position error, and percent out of formation is reported for each run. Data from runs on 10 random scenarios were averaged for each datapoint, the standard deviation of each factor is also recorded. Results for this experiment are summarized in Table 3.

Analysis of Results

For turns in a unit-center-referenced formation, diamond formations perform best. The diamond formation minimizes path ratio (1.03), position error (6.8 meters) and time out of formation (20.1 %). Unit-center-referenced formations appear to turn by rotating about their unit-center, so robots on the outside edge of the formation have to travel further in turns. The improved performance in diamond formations may reflect the smaller "moment of inertia" as compared to other formations. In a diamond formation, no robot is further than 50 meters from the unit-center. In contrast, the flanking robots in wedge, line, and column formations are 75 meters from the unit center.

For turns in a leader-referenced formation, wedge and line formations perform about equally. The line formation minimizes position error (8.2 meters), while the wedge formation minimizes time out of formation

Formation Type	Path Ratio Unit	Path Ratio Leader	Position Error Unit	Position Error Leader	Time out of Formation Unit	Time out of Formation Leader
Diamond	1.03 (0.08)	1.06 (0.08)	6.8 (0.2) m	11.4 (5.9) m	20.8 (0.3) %	21.6 (10.8) %
Wedge	1.04 (0.09)	1.06 (0.09)	9.4 (4.5) m	9.1 (6.2) m	25.6 (6.0) %	17.3 (9.6) %
Column	1.04 (0.06)	1.16 (0.02)	8.4 (5.6) m	21.1 (17.3) m	22.4 (8.1) %	32.4 (22.8) %
Line	1.04 (0.10)	1.05 (0.06)	8.5 (5.5) m	8.2 (5.1) m	25.7 (7.4) %	18.9 (10.8) %

Table 2: Performance for a 90 degree turn for both unit-center and leader-referenced formations. The standard deviation is indicated within parentheses.

Formation Type	Path Ratio Unit	Path Ratio Leader	Position Error Unit	Position Error Leader	Time out of Formation Unit	Time out of Formation Leader
Diamond	1.05 (0.04)	1.08 (0.05)	5.2 (1.9) m	7.1 (5.0) m	38.9 (15.0) %	34.8 (21.8) %
Wedge	1.04 (0.04)	1.08 (0.05)	5.2 (1.4) m	9.5 (8.4) m	37.9 (9.4) %	37.2 (24.3) %
Column	1.05 (0.04)	1.08 (0.04)	3.4 (1.6) m	6.4 (5.2) m	23.2 (11.8) %	28.5 (20.2) %
Line	1.05 (0.05)	1.05 (0.04)	5.3 (1.5) m	9.4 (8.5) m	36.1 (10.5) %	35.6 (23.8) %

Table 3: Performance for navigation across an obstacle field.

(17.3 %). Leader-referenced formations pivot about the leader in sharp turns. Robots significantly behind the leader will be pushed through a large arc during the turn. Line and wedge formations work well because fore and aft differences between the lead robot and other robots (0 and 50 meters respectively) is less than that for diamond and column formations (100 and 150 meters). Performance for column formations is significantly worse than that for line, wedge and diamond formations because the trail robot is the farthest back of all (150 meters).

For travel across an obstacle field, the best performance is found using column formations. Column formations minimize position error and percent time out of formation for unit-center- and leader-referenced formations. This result reflects the fact that column formations present the smallest cross-section as they traverse the field. Once the lead robot offsets to avoid an obstacle, the others can follow in its "footsteps."

In most instances, unit-center-referenced formations fare better than leader-referenced formations. A possible explanation is an apparent emergent property of unit-center-referenced formations; the robots appear to work together to minimize formation error. For instance, if one robot gets stuck behind an obstacle the others "wait" for it. The unit-center is anchored by the stuck robot so the **maintain-formation** schema instantiated in the other robots holds them back until the stuck robot navigates around the obstacle. This does not occur in leader-referenced formations.

Overall path length for robots in a leader-referenced formation is generally longer than in unit-center-referenced formations. This may be because any turn or detour by the lead robot is followed by all four robots, even if their path is not obstructed by the obstacle the leader is avoiding. A detour by the lead robot in a unit-center-referenced formation affects the entire formation, but the impact is 75% less than that found in leader-referenced formations since in the unit-center case an individual robot must shift 4 meters to move the unit-center 1 meter.

Summary and Conclusions

Reactive behaviors for four formations and two formation types were presented and evaluated. Experiments in simulation show that for 90 degree turns, the diamond formation performs best when the unit-center-reference for formation position is used, while wedge and line formations work best when the leader-reference is used. For travel across an obstacle field, the column formation works best for both unit-center- and leader-referenced formations. In most cases, unit-center-referenced formations perform better than leader-referenced formations. Even so, some applications probably rule out the use of unit-center-referenced formations:

- **Human Leader:** A human cannot be expected to reasonably compute a formation's unit-center on the fly, especially while simultaneously avoiding obstacles. A leader-referenced formation is most appropriate for this application.
- **Communications Restricted Applications:** The unit-center approach requires a transmitter and receiver for each robot and a protocol for exchanging position information. Conversely, the leader-referenced approach only requires one transmitter for the leader, and one receiver for each following robot. Bandwidth requirements are cut by 75% in a four robot formation.
- **Passive Sensors for Formation Maintenance:** Unit-center-referenced formations place a great demand on passive sensor systems (e.g. vision). In a four robot visual formation for instance, each robot would have to track three other robots which may spread across a 180 degree field of view. Leader- and neighbor-referenced formations only call for tracking one other robot.

This research has not yet addressed various modes of robot failure. Communications, sensor, and motor failures can significantly impact a formation. Mechanisms to deal with these failures might include automatic reconfiguration of the formation (renumbering) and application of fault-tolerant communications strategies. These behaviors are being ported to mobile robots both in our lab, and to Martin Marietta's UGV Demo II vehicles with more comprehensive results to be available in the Summer of 1995.

Acknowledgments

This research is funded under ONR/ARPA Grant # N00014-94-1-0215. The authors thank Doug MacKenzie and Jonathan Cameron who wrote the simulation software and helped debug the formation behaviors. Doug MacKenzie also developed the CNL language and compiler in which the formation behaviors are implemented.

References

Arkin, R. 1989. Motor schema based mobile robot navigation. *International Journal of Robotics Research* 8(4):92–112.

Army. 1986. *Field Manual No 7-7J.* Department of the Army, Washington, D.C.

Chen, Q., and Luh, J. Y. S. 1994. Coordination and control of a group of small mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2315–2320.

Cullen, J.; Shaw, E.; and Baldwin, H. 1965. Methods for measuring the three-dimensional structure of fish schools. *Animal Behavior* 13:534–543.

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

Hodgins, J., and Brogan, D. 1994. Robot herds: Group behaviors for systems with significant dynamics. In *Proceedings of Artificial Life IV*, 319–324. IEEE.

MacKenzie, D.; Cameron, J.; and Arkin, R. 1995. Specification and execution of multiagent missions. To appear IROS-1995.

Mataric, M. 1992a. 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*, 432–441.

Mataric, M. 1992b. Minimizing complexity in controlling a mobile robot population. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 830–835. IEEE.

Parker, L. E. 1994. *Heterogeneous Multi-Robot Cooperation.* Ph.D. Dissertation, M.I.T.

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

Veherencamp, S. L. 1987. Individual, kin, and group selection. In Marler, P., and Vandenbergh, J., eds., *Handbook of Behavioral Neurobiology, Volume 3: Social Behavior and Communication.* Plenum Press. 354–382.

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

Yoshida, E.; Arai, T.; and Yomoyoshi, M. 1994. Effect of grouping in local communication system of multiple mobile robots. In *Proc. 1994 IEEE International Conference on Intelligent Robots and Systems*, 808–815.