Heterogeneous Mobile Sensor Net Deployment Using Robot Herding and Line-of-Sight Formations

Lynne E. Parker, Balajee Kannan, Xiaoquan Fu and Yifan Tang
Department of Computer Science, The University of Tennessee, Knoxville, TN 37996–3450
Email: {parker, balajee, fu, ytang}@cs.utk.edu

***Abstract*— This paper presents an approach for deploying a team of mobile sensor nodes to form a sensor network in indoor environments. The challenge in this work is that the mobile sensor nodes have no ability for localization or obstacle avoidance. Thus, our approach entails the use of more capable "helper" robots that "herd" the mobile sensor nodes into their deployment positions. To extensively explore the issues of heterogeneity in multi-robot teams, we employ the use of two types of helper robots – one that acts as a leader and a second that: 1) acts as a follower and 2) autonomously teleoperates the mobile sensor nodes. Due to limited sensing capabilities, neither of these helper robots can herd the mobile sensor nodes alone; instead, our approach enables the team as a whole to successfully accomplish the sensor deployment task. Our approach involves the use of line-of-sight formation keeping, which enables the follower robot to use visual markers to move the group along the path executed by the leader robot. We present results of the implementation of this approach in simulation, as well as results to date in the implementation on physical robot systems. To our knowledge, this is the first implementation of robot herding using such highly heterogeneous robots, in which no single type of robot could accomplish the sensor network deployment task, even if multiple copies of that robot type were available.**

I. Introduction

In this paper, we address the issue of robot team heterogeneity in the context of mobile sensor net deployment in an indoor environment. In general, if all mobile sensor nodes have the ability to locomote and to sense other robots and obstacles in the environment, then a distributed dispersion algorithm based on potential fields (e.g., [1]) would be an appropriate solution strategy for deploying the mobile sensor network. However, if some of the robots do not have the sensing capability to detect obstacles or other robots (but they do have locomotion capabilities and special-purpose sensors needed in the sensor network, such as acoustic or chemical sensors), then such a solution strategy would no longer work. On the other hand, if some of the robot team members were highly capable robots that could help navigate the less capable robots, then a workable solution strategy would be for the more capable robots to guide the less capable robots to a deployment position. This is the approach we present in this paper.

Section II provides an overview to our approach and the behaviors of the various robots. In Section III, we discuss our approach to vision-based detection of robot ID and relative pose using visual markers. Section IV discusses our approach to maintaining line-of-sight formations. Our approach for planning for sensor net deployment is briefly discussed in Section V. We present the results of our integrated approach in Section VI. Related work is described in Section VII. We offer concluding remarks in Section VIII.

II. Overview of Approach

Our approach to mobile sensor net deployment involves the collaboration of three types of robots. The first type is the mobile *Sensor Node*, which has the ability to move and perform acoustic sensing, but cannot localize or avoid obstacles. The second type is a more capable *Leader Helper* robot. The *Leader Helper* robot has a laser scanner that allows it to perform localization using a method such as [2]. The third type is also a more capable *Follower Helper* robot, except this robot can only perform relative localization to other robots detected using a vision system. All these robots can communicate with each other to share information and control commands as needed in order to successfully deploy the sensor network. Eventually, our project will involve 70 *Sensor Node* robots, plus several *Helper* robots.

Figure 1 shows an illustration of this herding process. In this figure, the robot nearest to the camera is the *Leader Helper*, which is responsible for planning and following the path to the sensor net deployment positions, using its localization capabilities to accurately locate itself in the environmental map. The robot in the back is the *Follower Helper*, which is using a camera to detect the current position and orientation of the robots in front of it (i.e., the *Leader Helper* and the *Sensor Nodes*). The *Follower Helper*'s goal is to follow the path taken by the *Leader Helper*, and to autonomously navigate the *Sensor Nodes* so that they follow the same path. Here, by *autonomously navigate*, we mean that the *Follower Helper* communicates velocity and steering commands to each *Sensor Node* robot to enable them to follow the path taken by the *Leader Helper*. Since motion commands will be unique to each robot, the *Follower Helper* must be able to detect the unique identification and pose of each of the robots in its herd. To make this detection of robot pose easier, we make use of color markers as described in Section III. Clearly, in order to use visual markers for this purpose, the markers must always be within view of the *Follower Helper*, and marker occlusion should be minimized. To accomplish this, we generate allowable formations that maintain line-of-sight between the *Follower Helper* and all other robots, as described in Section IV. The *Follower Helper* then generates motion commands for the *Sensor Nodes* to ensure that the line-of-sight

Fig. 1. Photo of an example “herd” of robots that will use autonomous teleoperation and line-of-sight formations to enable the complete team (i.e., “herd”) to move to deployment positions. The robot nearest the camera is the *Leader Helper*, the robot in the rear is the *Follower Helper*, and the two robots in the middle are the *Sensor Nodes* being herded.

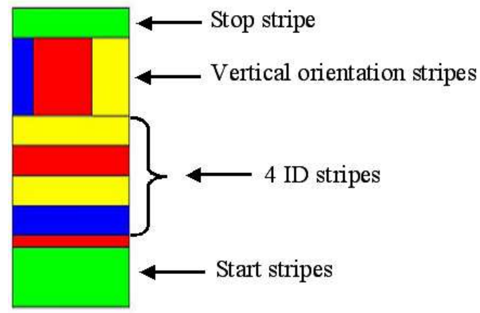

Fig. 2. Color marker design for unique robot identification.

formation is maintained while the team follows the *Leader Helper*.

Initially, a plan for how to deploy the entire set of sensor nodes must be developed (as briefly introduced in Section V). From this planning process, the *Leader Helper* robot is provided a deployment path that it should follow for deploying the *Sensor Nodes*. As the deployment path is generated, all robots within a herd move into their starting formation. The formation is generated in order to maintain visibility from the *Follower Helper* robot to all other robots in the team. The *Follower Helper* robot determines when all robots are in the starting formation and signals this event to the *Leader Helper* robot. The *Leader Helper* robot then begins moving to the next deployment position, based upon its derived deployment path. As it moves, the *Follower Helper* robot follows the *Leader Helper* and autonomously teleoperates the *Sensor Nodes* along the way. As the group moves, the *Follower Helper* also ensures that the *Lead Helper* does not get too far ahead of the group. If the *Leader Helper* begins getting too far ahead (based upon a predefined preferred distance), the *Follower Helper* signals the *Leader Helper* to adjust its velocity appropriately. Once the group has reached the next deployment position, the *Leader Helper* signals this fact to the *Follower Helper*. At that point, the group stops while the *Follower Helper* teleoperates one of the *Sensor Nodes* into its sensor net deployment position. Once the *Sensor Node* is in position, it converts to its primary sensor network detection role. The rest of the group then proceeds to the next deployment position, and the process continues until all *Sensor Nodes* in the group have been deployed.

III. Robot Detection

For the *Follower Helper* robot to be able to autonomously teleoperate the *Sensor Nodes*, it needs to be able to detect the unique ID and pose of each *Sensor Node*. The *Follower Helper* robot also needs to detect the current position of the *Leader Helper* robot, in order to follow its path. Our approach to providing this capability is to use a color cylindrical marker on each robot that provides information about the unique identity and orientation of the robot. Based upon the location and orientation of the marker in the image, the ID and relative pose information of the robot can be extracted.

Figure 2 shows the design of the color cylindrical marker we are currently using, which is about 22 centimeters high. At the bottom of the marker, a combined green and red stripe together form a START block. Two color stripes are used in this way for the START block to decrease the likelihood of false positives. Another green stripe at the top of the marker is the STOP stripe. The region between the START block and the STOP stripe contains additional stripes for ID and pose information.

Above the START block are four horizontal ID stripes that define the ID of the robot. Each stripe can be one of three colors, giving $3^4 = 81$ different robot IDs available, which is sufficient for our purposes. Above the ID stripes are three vertical orientation stripes around the circumference of the marker. The orientation of the marker can be calculated from the color and width ratio of the orientation stripes that are visible in the image. The distance of the marker from the camera can be calculated from the size of the marker in the image.

To reduce the sensitivity of this approach to lighting conditions, we are implementing an autonomous calibration capability that enables the system to autonomously vary the color values that represent the marker. Figure 1 shows this marker design mounted on our robots.

IV. Line-of-Sight Formation Keeping

Our approach to heterogeneous sensor net deployment is dependent upon the ability of the *Follower Helper* robot to see the markers that identify each robot in the group and provide relative pose information. Thus, the group must move so as to maintain the line-of-sight from the *Follower Helper* to all other robots. In our application, only the *Follower Helper* has vision capabilities; the other robot team members are not able to detect the relative positions of their teammates.

We divide this line-of-sight formation problem into two parts. The first is generating a formation that satisfies the line-of-sight constraints. The second is the control technique that

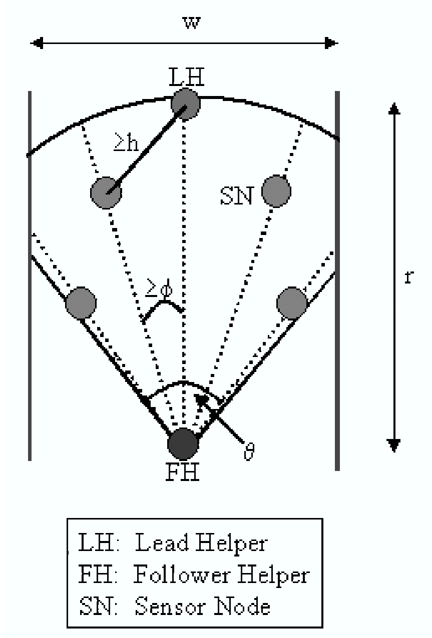

Fig. 3. Representation of constraints that operate upon line-of-sight formation generation.

enables the robot team to maintain the selected formation as they move through the environment.

We formulate the formation generation problem by specifying the constraints that must be met in a satisfactory solution. In this discussion, let n be the number of *Sensor Node* robots plus 1 (for the *Leader Helper* robot). As illustrated in Figure 3, the constraints are as follows:

1) The field of view of the *Follower Helper* is limited to the angle θ.
2) In the *Follower Helper* robot's image of its teammates, robot markers must be separated by a minimum angle of separation, ϕ.
3) The maximum effective sensing range of the *Follower Helper*'s camera is r.
4) All robot team members must be separated in physical space by a minimum distance, h.
5) The maximum width of the formation is w.

The field of view and range restrictions are a result of the physical limitations of the camera mounted on the *Follower Helper*. The maximum width of the formation is dependent upon the environmental constraints. We plan for this width to be calculated by the *Leader Helper* using a laser range scanner.

Our approach to generating a formation that satisfies these constraints is to first ensure that $\theta \geq (n-1) \times \phi$. If this is not true, then no solution is possible for n robots that satisfies the minimum angular distance of separation constraint. If this is true, then we divide θ into $n-1$ equal angles[1], as shown by the dotted lines in Figure 3. Our approach then is reduced to finding *Sensor Node* location assignments along each of the dotted lines such that the distances between all robots is at least h (to satisfy our fourth constraint above). We note that this problem can be solved using linear programming techniques. However, these techniques are computationally intensive, and cannot easily be expected to run in real time onboard the robots. Therefore, we address this problem by first computing an off-line solution using linear programming, such that for any n, a formation that maintains the line-of-sight constraints is pre-computed. We define this table of formations for a coarse discretization of values of w. We assume that, for a given *Follower Helper* camera system, the values of r and θ are fixed. Then, at run-time, the *Follower Helper* robot performs a table lookup to determine which formation is preferred for the current n and w.

To maintain the formation, the *Follower Helper* must send control commands to the *Sensor Nodes* based upon the current relative position of the *Leader Helper*. At present, we use a potential fields approach (similar to [3]) to drive the *Sensor Node* robots to their positions in the formation, using remote commands from the *Follower Helper*.

In practice, the environment width will change during the robot motions, requiring the robot team to be able to change formations when needed. We accomplish this by an additional table lookup based upon the current width of the environment, w. Additionally, the formation can automatically change during the application if the number of robot team members, n, changes, again through another table lookup. One possible method for changing formations would be to take advantage of the work of Desai [4], which defines a framework for transitioning between various formations.

V. Sensor Net Deployment Planning

In a large application, multiple herds of robots will be deployed simultaneously in different parts of the building. Thus, a plan is needed to coordinate these multi-team activities. Our ongoing work is aimed at coordinating the deployment of a large sensor network through pre-planning the routes that different herds should take and the positions in which the *Sensor Nodes* should be placed. This deployment planning is based upon the use of the *Sensor Nodes* not only for distributed sensing, but also for maintenance of a mobile communications network. Space does not allow a discussion of the complete pre-planning process. However, a first step is to plan the deployment positions for all of the sensor nodes. Our approach includes both a static component and a dynamic component for deployment planning. The static component involves an analysis of the environmental map (which we assume has been

[1] In practice, we actually reduce the field of view slightly in these calculations, to give extra room at the extreme edges of the field of view. This helps ensure that a robot is not placed at the edge of visibility, where slight errors in position will place the robot out of the field of view of the *Follower Helper*. This is shown in the figure by the slight offset of the dotted lines to the interior region of the *Follower Helper*'s field of view.

previously generated by mapping robots using a technique such as [2]) to determine placement of *Sensor Nodes* to maximize sensor coverage. In this step, we use a method similar to [5] to incrementally find deployment positions that maximize the additional visibility coverage.

In brief, our approach for planning the deployment positions works as follows. We choose a starting position for the first sensor node (which is referred to as an anchor position in [5]). We use a ray sweep algorithm based on the occupancy grid map to calculate the potential deployment positions and the line-of-sight coverage of those positions. A tree structure is provided to hold important information about all potential deployment positions. The tree is traversed according to the maximum additional coverage of each potential deployment position. The ray sweep algorithm is applied twice – first with the constraint that successive deployment positions must be within line of sight of each other, and then without this constraint, in the case that a sufficient number of deployment positions are not found. The static component also involves adjusting the deployment positions to be next to the obstacleswalls instead of in the middle of hallways or doorways, so as to minimize the obstruction of the movement of other robots. Furthermore, a minimum distance between the *Sensor Nodes* is enforced to optimize sensor net performance.

The dynamic component of the *Sensor Node* deployment process operates as the robots are being deployed, rather than in the pre-planning step. This component involves placement of the *Sensor Nodes* to maintain the communications network when the strength of the radio signal back to the nearest network node falls below a threshold before the next pre-calculated sensor position can be reached. This step involves placing additional sensor nodes between pre-calculated sensor positions during deployment. Implementation of this approach to sensor net deployment planning is underway.

VI. Results

Our approach to heterogeneous mobile sensor net deployment using robot herding and line-of-sight formations has been implemented in the Player/Stage simulator [6]. In these simulation studies, we have used an indoor environment that represents a decommissioned hospital at Fort Sam Houston in San Antonio, Texas. Figures 4 and 5 show an example run of this implementation with several “herds” of size five. Figure 4 shows the early stages of sensor net deployment that uses three groups of *Helper* and *Sensor Node* robots. In this sample run, there are three additional *Sensor Nodes* and a *Follower Helper* robot waiting for deployment near the center of the building. Figure 5 shows the robots as the three groups are reaching the deployment positions further in this sample run. Once the *Sensor Nodes* have been deployed, their positions are noted in the environmental map and the *Leader Helper* robots return for another set of *Sensor Nodes* to deploy. A video is available of a sample run of this shepherding deployment.

Experimentation is ongoing to fine-tune our shepherding approach and to write the “wrapper” code that enables multiple iterations of the shepherding behavior by the same *Helper* robots. These techniques will allow a large number of *Sensor Nodes* (70, for our experiments) to be shepherded through the environment through multiple iterations of the behaviors. We are also fine-tuning our approach to ensure high robustness and fault tolerance of this group behavior, enabling robot team members to recover from a variety of failure modes during this shepherding process. Our aim is to develop fault tolerance such that any single robot failure does not cause the failure of the entire group.

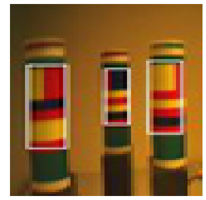

Fig. 6. Results of our marker detection algorithm. The markers detected by our software are marked with white bounding boxes. Refer to Table I for the robot ID and pose information calculated by our detection software for this example.

TABLE I

Results from Marker Detection Software (for markers in Figure 10, read from left to right).

Marker ID	**Distance from Camera**	**Orientation to Camera**	**Orientation of Marker**
1233	21.9 inches	117°	349°
3231	35.0 inches	84°	185°
2321	25.0 inches	62°	31°

Our marker detection algorithm has been implemented and is being evaluated and fine-tuned for this application. Figure 6 shows example results of our marker detection code. This figure shows the location of the markers detected by our algorithm, as indicated by white bounding boxes. Table I shows the ID and pose information determined by our software for this example. In this table, the “Orientation to Camera” value is the position of the marker in the image relative to the camera, with 0° being directly to the right of the image plane. The “Orientation of the Marker” value is the orientation of the marker about the marker’s vertical axis.

At present, our marker detection code is able to provide complete ID and pose information for several markers when the markers are unobstructed in the image. When markers are only partially visible, our approach can also give partial pose information for those incomplete markers, depending on which parts of the marker are occluded. Our ongoing tests are tuning the system to achieve a high success rate of marker detection and interpretation. Some factors that affect the detection accuracy are the lighting conditions, background colors, relative mounting positions of the camera and markers, and marker color selections.

We are also implementing our approach on a team of physical robots, as shown in Figure 1. The *Helper* robot capability is being implemented on ATRV-mini robots, one of which has a SICK laser range scanner (which will be the

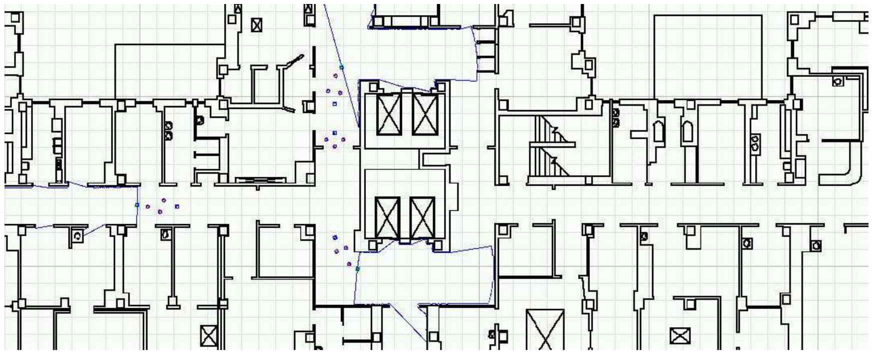

Fig. 4. Early stages of sensor net deployment with multiple groups of *Helper* and *Sensor Node* robots.

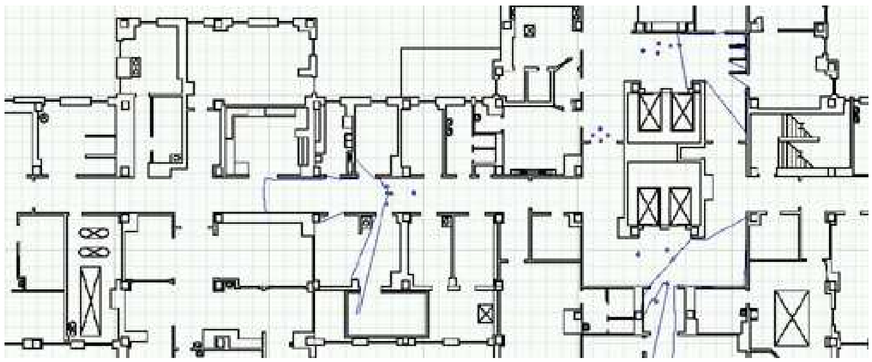

Fig. 5. Robot positions as the deployment positions are being reached. Note the *Sensor Nodes* in the lower right quadrant deploying to distributed positions. The other two groups are nearing their deployment positions.

Leader Helper), and the other of which has a Sony pan-tilt-zoom camera (which will be the *Follower Helper*). The *Sensor Nodes* consist of AmigoBots with iPAQ computers for computation and a low-fidelity microphone. The AmigoBots have no other sensors other than wheel encoders. All robots can communicate using wireless ad hoc networking. In addition, implementation of the off-line line-of-sight formation generation is underway, along with extensive experimentation to determine whether practical considerations should further constrain the solution for line-of-sight formation generation.

VII. Related Work

Several areas of related work are relevant to this project, including sensor net deployment, formation generation, formation keeping, and vision-based robot detection. In the area of sensor net deployment, Chakrabarty et al. [7] have developed approaches for deployment in a grid field. However, this method requires a very large number of sensors and thus is not suitable for our application in indoor environments. Howard, et al. [5] have developed an approach for the incremental deployment of sensor nodes. This algorithm enables deployment in an unknown area based on the cumulative information from each deployed sensor, while satisfying the line-of-sight constraint. Our approach is different in that we know the environment *a priori* and can perform static pre-calculation to attempt to optimize the deployment positions. An additional approach to deployment has been developed by Howard, et al. [1], involving the use of a potential field deployment strategy that enables a maximal coverage in an unknown area. However, this approach requires range-sensing capabilities from the sensor nodes, which is not present for our application. Furthermore, in applying the algorithm of [1], the sensor nodes will be deployed equally distanced from obstacles, e.g. in the middle of the hallway, and thus will hamper the movement of other robots in the area.

Payton, et al. [8] implement attraction/repulsion behaviors to enable robot swarms to be distributed into an unknown area. The robots must maintain line-of-sight for the purposes of communication and virtual pheromones are used to signal a discovery. In this approach, the robots act and communicate locally without the need for a centralized map. Clouqueur, et al. [9] introduce path exposure (“the probability of detecting the target or an intruder”) as a metric for sensor net coverage. They employ a random distribution of sensor nodes and examine the cost of sensor deployment. In order to reduce the deployment cost, they propose a solution to deploy part of the maximal available sensors first, then use the information collected from the deployed sensors to determine whether the desired path exposure is achieved.

A significant amount of work has dealt with formations in multi-robot teams; space does not allow an extensive discussion of this prior work. For example, Balch [3] highlights the advantages and disadvantages of different formations in dynamic environments as well as the usefulness of various approaches under certain environmental constraints. However, his work did not address the issue of formation generation dealing with line of sight constraints. Once a formation is generated, various approaches exist for maintaining that formation. For example, control algorithms for vision-based formation control have been successfully implemented by Das, et al. [10]. Their approach builds a control-theoretic framework for formation control using omnidirectional vision. By maintaining certain control heuristics, the follower can maintain its position in the formation with respect to the leader. The choice of formation strategy used in [10] is based on the ability of each individual robot to change formation to avoid obstacles. In our approach, only the *Follower Helper* robot has vision capabilities.

In the area of vision-based robot detection, several previous approaches have used color markers to simplify the detection problem. For example, the approach in [11] uses solid colored 2D circles and regular triangles in six colors as fiducials. These solid colored fiducials are more robustly detectable than multi-colored fiducials of the same size. However, this approach can not provide enough combinations to make the required number of unique fiducials (in our case, more than 70). Additionally, it cannot provide orientation information of the fiducial. The approach in [12] uses concentric black and white circular fiducials to measure distance. Similarly, Cho and Neumann [13] use concentric multi-ring, multi-size color circular fiducials. However, these approaches do not provide the combined information of robot identification and pose. Malassis and Okutomi [14] use a simple three-color fiducial to apply colored surface projection to obtain pose information, which provided inspiration to our marker design.

VIII. Conclusions

In this paper, we have described an approach for heterogeneous mobile sensor network deployment using robot herding and line-of-sight formations. In this approach, no single type of robot is able to accomplish the sensor net deployment task. Instead, three types of robots work collaboratively to enable the deployment to be accomplished. This approach involves the use of two types of *Helper* robots that assist in moving *Sensor Node* robots through the environment. The *Leader Helper* robot is able to plan paths and localize in the environment, while the *Follower Helper* robot can use a vision system to detect the relative pose of other robot team members. *Sensor Nodes* are autonomously teleoperated by the *Follower Helper* robot to maintain a line-of-sight formation with the *Leader Helper* robot, which is moving along a planned deployment path.

We have successfully implemented this approach in simulation and presented example results of these implementations in the Player/Stage simulation environment. The implementation of this approach on our team of physical robots is also underway. We presented results of our implementation of the color marker detection and interpretation algorithms that are critical for the success of this approach. Our ongoing experiments are aimed at 1) completing the linear programming calculations of optimal line-of-sight formations for a variety of robot team sizes and environmental constraints, 2) completing the implementation of our deployment planning strategy, and 3) implementation of the formation control on the physical robot team. To our knowledge this is the first implementation of robot herding using such highly heterogeneous robots, in which no single type of robot could accomplish the sensor network deployment task, even if multiple copies of that robot type were available. From a broader perspective, this research illustrates how highly heterogeneous teams can work together to share sensor capabilities to accomplish challenging tasks.

Acknowledgments

The authors thank Chris Reardon and Ben Birch for their valuable discussions regarding this research. This research was sponsored in part by DARPA/IPTO's Software for Intelligent Robotics program, through Science Applications International Corporation, and in part by The University of Tennessee's Center for Information Technology Research. This paper does not reflect the position or policy of the U. S. Government and no official endorsement should be inferred.

References

[1] A. Howard, M. J. Mataric, and G. S. Sukhatme, "Mobile sensor network deployment using potential fields: A distributed scalable solution to the area coverage problem," in *Distributed Autonomous Robotic Systems 5: Proceedings of the Sixth International Symposium on Distributed Autonomous Robotic Systems (DARS 2002)*. Springer-Verlag, 2002, pp. 299–308.

[2] S. Thrun, D. Fox, and W. Burgard, "A probabilistic approach to concurrent mapping and localization for mobile robots," *Autonomous Robots*, vol. 5, pp. 253–271, 1998.

[3] T. Balch and R. Arkin, "Behavior-based formation control for multi-robot teams," *IEEE Transactions on Robotics and Automation*, December 1998.

[4] J. P. Desai, "Modeling multiple teams of mobile robots: A graph theoretic approach," in *Proceedings of 1993 IEEE International Conference on Intelligent Robots and Systems (IROS '01)*, 2001, pp. 381–386.

[5] A. Howard, M. J. Mataric, and G. S. Sukhatme, "An incremental self-deployment algorithm for mobile sensor networks," *Autonomous Robots*, vol. 13, no. 2, pp. 113–126, 2002.

[6] B. Gerkey, R. Vaughan, K. Stoy, and A. Howard, "Most valuable player: A robot device server for distributed control," in *Proc. of 2001 IEEE/RSJ International Conference on Intelligent Robotics and Systems*, 2001, pp. 1226–1231.

[7] K. Chakrabarty, S. S. Iyengar, H. Qui, and E. Cho, "Coding theory framework for target location in distributed sensor networks," in *Proceedings of IEEE International Conference on Information Technology: Coding and Computing*, 2001.

[8] D. Payton, M. Daily, R. Estowski, M. Howard, and C. Lee, "Pheromone robotics," *Autonomous Robots*, vol. 11, no. 3, pp. 319–324, 2001.

[9] T. Clouqueur, V. Phipatanasuphorn, P. Ramanathan, and K. K. Saluja, "Sensor deployment strategy for target detection," in *Proceedings of First ACM International workshop on wireless sensor networks and applications*, 2002.

[10] A. Das, R. Fierro, V. Kumar, J. Ostrowski, J. Spletzer, and C. Taylor, "A vision based formation control framework," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 813–825, October 2002.

[11] Y. Cho, J. Parker, and U. Neumann, "Fast color fiducial detection and dynamic workspace extension in video see-through self-tracking augmented reality," in *Proceedings of the Fifth Pacific Conference on Computer Graphics and Applications*, 1997, pp. 168–177.

[12] A. Walthelm and R. Kluthe, "Active distance measurement based on robust artificial markers as a building block for a service robot architecture," in *Proceedings of the Fifth IFAC Symposium*, 2001, pp. 325–330.

[13] Y. Cho and U. Neumann, "Multi-ring color fiducial systems for scalable fiducial tracking augmented reality," in *Proceedings of the Virtual Reality Annual International Symposium*, 1998, p. 212.

[14] L. Malassis and M. Okutomi, "Three-color fiducial for pose estimation," in *Proceedings of the Asian Conference on Computer Vision*, 2000.