Proc. of IEEE International Conferences on Robotics
and Automation, New Orleans, LA, April 2004.

Planning Mobile Sensor Net Deployment for Navigationally-Challenged Sensor Nodes

Yifan Tang, Ben Birch, Lynne E. Parker
Distributed Intelligence Laboratory, Department of Computer Science
The University of Tennessee, Knoxville, Tennessee 37996-3450
Email: {ytang, birch, parker}@cs.utk.edu

***Abstract*— This article describes novel algorithms for planning the deployment of a large number of navigationally-challenged mobile sensor nodes in known indoor environments. Due to cost and power constraints, our mobile sensor nodes have the ability to move and communicate, but they cannot detect or avoid obstacles or localize to the environment. Additionally, they have only minimal capabilities for detecting other robot team members, through the use of a crude camera. Our deployment process, therefore, uses an *assistive navigation* technique that enables a more capable *Leader* robot, equipped with a laser rangefinder and a camera, to guide several mobile sensor nodes to their deployment positions. To ensure the successful deployment of the mobile sensor nodes, we have developed an autonomous planning process that plans the positions of the sensor nodes based upon a number of constraints, including maintaining line of sight, maximizing visibility coverage, avoiding placement in doorways, minimizing obstruction of corridors, and so forth. Additionally, because of the navigational constraints of simple robots following a *Leader* to these deployment positions, our algorithm also derives two *Leader* waypoints for each sensor position, which constrain the motion of the *Leader* path to the deployment position. These *Leader* waypoints ensure that the sensor robots following behind are properly positioned to be guided into their deployment positions. The final part of our planning process involves grouping and ordering sensor positions into smaller teams that are assigned for deployment in a single pass by a single *Leader*. To maximize the likelihood of the successful deployment of each deployment team, our planning process groups and orders sensor positions to ensure certain turning constraints of the *Leader* path are maintained. We have successfully implemented and integrated these planning algorithms, and present the results of the implementation using three different environmental maps. Additionally, our sensor deployment plans have been successfully executed by teams of physical mobile robots, further validating our approach. To our knowledge, this is the first set of algorithms developed for planning the assistive deployment of navigationally-challenged mobile sensor nodes.**

I. INTRODUCTION

A growing body of research is addressing the topic of sensor networks and the development of techniques that allow them to perform distributed sensing and information gathering in a variety of environments. Sensor network research includes the development of power-efficient hardware to create useful sensor devices, the development of algorithms that allow information to be efficiently merged and interpreted from distributed sources, and techniques for deploying the sensor nodes into their target environment. This paper addresses this latter issue of sensor network deployment, specifically for the case of navigationally-challenged mobile sensor nodes.

Most commonly, sensor nodes are considered to be immobile, meaning that they are incapable of generating motion of their own volition. Of course, these sensor nodes may in fact move in their environment when they are caught in wind or water currents, or due to gravity, but they cannot typically move against these natural forces. Often, these sensor devices are deployed through means such as air drop, ballistic propulsion, release in bodies of water, manual placement, etc. Other types of sensor networks are mobile, and can actively move in their environment in order to achieve a desired sensor node density or distribution, or to follow a gradient, and so forth.

Our research involves a team of up to 70 mobile sensor nodes that must achieve a desired spatial distribution that meets several criteria in geometric indoor environments. However, due to cost and power constraints, the mobile sensor nodes in our research individually have no capability for obstacle detection, obstacle avoidance, localization, and only minimal recognition of their "kin" (i.e., other sensor nodes) through a simple color blob tracking capability. These sensor nodes do have mission-relevant sensors that allow them to detect important environmental features where they are located (acoustic signals, in the case of our research). However, in other respects they are sensor-limited, and cannot navigate in their environment to achieve a desired dispersion in geometric indoor environments.

Our approach to achieving deployment of these simple sensor robots is an *assistive navigation* technique that uses more intelligent *Leader* robots to guide the sensor nodes to planned deployment positions in the environment[1]. These *Leader* robots, which use a laser rangefinder to localize themselves to the environment, are followed by a small group of sensor nodes (on the order of 1-5), which form a chain behind the *Leader* and use a crude camera to perform simple color blob tracking to follow the robot ahead of it. Once the robot team reaches a deployment position, the *Leader* then uses a camera to detect a fiducial on the first sensor robot and autonomously teleoperate that robot to its exact planned deployment position. This process repeats with the *Leader* and the remaining chain proceeding to the next deployment position until all the sensor nodes of that team have been

---

[1] Refer to [6] for a discussion of various alternative deployment strategies in heterogeneous multi-robot teams.

Fig. 1. Physical robots in the process of executing the assistive navigation technique to deploy navigationally-challenged robots. Here, a *Leader* robot is guiding four sensor node robots to the deployment positions that have been planned using the algorithms described in this paper.

deployed. The *Leader* then returns to its Home position to pick up more sensor node robots for deployment. Figure 1 shows a team of 4 sensor node robots following a *Leader* robot in the process of executing this deployment strategy. These robots are en route to a subset of the sensor positions planned using the algorithms described in this paper.

With this requirement for providing navigational assistance, it is important to develop a thoughtful deployment strategy to ensure that the sensor nodes are positioned in a desired distribution, and to ensure that the *Leader* robots can efficiently guide the sensor nodes to their desired positions while leading a follow-the-leader formation. Obviously, the navigational challenges grow if the *Leader* robot moves to random sensor node deployment positions without taking into account the turning constraints of the formation of robots that is following behind. Additionally, sensor robots should not be placed in positions that may block doorways or create difficult obstacle fields in which other robots must operate.

Ad hoc methods of distributing the sensor nodes in their environment are not guaranteed to achieve the desired distribution of the sensor nodes in the environment. Additionally, it is unclear how to develop efficient ad hoc techniques for deployment when each *Leader* robot can only assist a few sensor nodes at a time, when they must use chaining for deployment, and when several deployment teams are operating in parallel to speed the deployment. Clearly, the stage is set for wildly uncoordinated clusters of robots if advanced planning is insufficient.

The remainder of this paper describes our approach to solving this problem, beginning with an overview of our solution in Section II. Section III describes the sensor deployment position planning algorithm, while Section IV describes a team assignment algorithm that divides the sensor positions into subgroups for deployment. In Section V, we present results of the planning algorithm applied to three maps generated by physical robots, along with brief results of physical robots executing a deployment plan. We discuss related work in Section VI and conclude in Section VII.

II. OVERVIEW OF APPROACH

Our approach to the deployment planning of navigationally-challenged sensor nodes involves several steps. Since we assume a known environment, the process begins with an automatically-generated map of the environment. In our research, this map is developed by one or more mobile robots that generate an occupancy map using a laser scan matching technique such as [10]. This map is converted to an occupancy grid of resolution approximately 25cm x 25cm, which is then used throughout our planning phases.

In the first planning step, sensor node deployment positions are derived that meet several constraints, including maintaining line of sight, maximizing visibility coverage, avoiding placement in doorways, minimizing obstruction of corridors, achieving a minimum distance between sensor nodes, and so forth. A second component of our planning calculates two *Leader* waypoints for each sensor position, in order to constrain the motion of the *Leader* path to the deployment position. These *Leader* waypoints ensure that the sensor robots following behind are properly positioned to be guided into their deployment positions. These two components of the planning process are tightly coupled in that, after each new possible sensor deployment position is generated, the algorithm tests whether or not there is sufficient operating space around that sensor deployment position for the *Leader* robot. If there is enough space, this position is considered as a candidate sensor deployment position by the algorithm. If insufficient space is available for the *Leader* robot, this position is rejected.

Since each *Leader* robot can only deploy a few sensor nodes at a time (typically, 1-5 in our physical robot experiments), and since several *Leader* robots are available to operate in parallel, the final part of our planning process involves grouping and ordering sensor positions into smaller teams that are assigned for deployment in a single pass by a single *Leader*. To maximize the likelihood of the successful deployment of each deployment team, our planning process must group and order sensor positions to ensure certain turning constraints of the *Leader* path are maintained. Thus, these team assignments are generated to achieve the objectives of minimizing travel by the deployment teams and minimizing the amount of turning a team must perform as it travels to all of its assigned deployment positions. Achieving the second objective can reduce the complexity of deployment as the sensor node robots travel in chains behind the *Leader* robot during the deployment.

The following sections describe the further details of this deployment planning process.

III. PLANNING SENSOR DEPLOYMENT AND LEADER ROBOT POSITIONS

Our approach to planning sensor deployment positions is related to the work of Howard, et al. [3], which develops an approach for the incremental deployment of sensor nodes. This prior work introduces a tree-like structure of potential sensor deployment positions to assist in the planning process, which we also use in our planning. However, we have introduced several new capabilities to this prior work, based upon the sensing limitations of our mobile sensor nodes and the deployment constraints of *Leader*-driven navigational assistance.

Using our algorithm, the first candidate sensor deployment position is generated at a particular location of interest in the map, whose position is supplied to the planning process. The algorithm then generates candidate sensor positions by ray sweeping (at 5-degree increments) from the last generated sensor deployment position. Each candidate position must satisfy the **WITHIN_SENSING_RANGE** criterion, which ensures that candidate positions are at least within sensing range of a prior deployment position. Additionally, candidate sensor positions that meet the **WITHIN_LOS** criterion, which ensures that the candidate position is also within line-of-sight of a prior deployment position, are preferred. Next, the **VISIBILITY_COVERAGE** criterion is applied, preferring new deployment positions that maximize the visibility coverage. Candidate deployment positions are then considered in descending order of the visibility coverage, with preference first going to candidate solutions that meet the **WITHIN_LOS** criterion.

Each candidate position must meet several additional constraints. First, the **ADJ_NEARBY_OBSTACLE** and **AVOID_DOORWAY** criteria are applied to minimize pathway occlusion by avoiding sensor deployment positions in the middle of a narrow hallway or in the middle of a doorway. To achieve these constraints, the algorithm measures the distance between the sensor deployment position and the next nearest obstacle to ensure that it satisfies pre-defined distance constraints. The algorithm also searches for nearby doorways to ensure that the candidate position is sufficiently far to prevent blockage.

Next, the **MIN_SENSOR_DIST** criterion is applied to maintain a minimum distance between all sensor deployment positions. This constraint is necessary to ensure that the resulting sensor positions are adequately distributed for mission-relevant sensing once the sensor network is deployed. The **ROOM_FOR_LEADER** criterion is applied that couples the calculations of the sensor deployment positions and the *Leader* robot waypoint positions to ensure sufficient operating space for the *Leader* robot in the actual deployment process. Figure 2 illustrates the relationship between the planned sensor position and the two leader waypoint positions.

If the candidate sensor position meets these criteria, it is then added to the list of selected deployment positions. The new deployment position is also added as an obstacle to the map to prevent *Leader* robot positions from being generated too near to a deployed sensor. This process then iterates, with the next set of candidate deployment positions being generated from this new deployment position by repeating the ray-tracing process and recursively exploring the tree of candidates until no more new sensor deployment positions can be found.

After the sensor deployment positions are planned, the waypoints for the *Leader* robot to travel to during the deployment process must be generated, as shown in Figure 2. During the actual deployment process, the *Leader* robot passes through the first planned waypoint position and then stops at the second planned waypoint position. The *Leader* robot positions are planned in such way that the sensor node robot immediately following the *Leader* robot will be adjacent to the planned sensor deployment position when the *Leader* robot stops at the second waypoint. In this manner, the sensor node will be properly positioned for deployment by the *Leader* robot using autonomous teleoperation. Because the *Leader* robot can autonomously localize itself (using laser scan matching to the known map), it employs visual marker detection to detect a fiducial on the sensor node robot and autonomously teleoperate the sensor node robot to its desired deployment position.

The order in which the two *Leader* waypoint positions are visited is dependent upon the path the *Leader* takes through its environment, and the specific set of deployment positions that the *Leader* has been assigned. The order of visitation of these *Leader* waypoints is determined by our team assignment algorithm described in the following section. At execution time, once a sensor node has been guided into position, the *Leader* continues on with the rest of its robot chain to the next assigned deployment position, until all of its assigned robots are deployed.

The pseudocode for our sensor and *Leader* robot position planning is shown in Table I. In these algorithms, several parameters are provided that implement the described constraints. The settings of these parameters can be easily varied according to the environment and mission requirements, depending upon the needs of the specific application.

IV. TEAM ASSIGNMENT

After determining the positions of the nodes of the sensor network and the positions to which the *Leader* robots are to travel during deployment, the next phase in the planning process is dividing the N sensor positions into smaller groups of maximum size n. Each of these groups of positions, along with an equal number of sensor nodes, is assigned to a *Leader* robot for deployment. The maximum team size, n, can be varied; larger team sizes will reduce the number of *Leader* robots or the number of trips from the Home area required, but may increase the difficulty of executing the deployment. (In our physical robot experiments, the practical range of n is 1-5.)

A naive approach to team assignment would simply group proximate sensor positions and assign them to a *Leader*. However, the physical robot limitations of the simple sensor nodes following the *Leader* in a chain prevents this approach from working well in practice. Because the navigationally

TABLE I

SENSOR DEPLOYMENT AND LEADER POSITION PLANNING

Sensor Deployment and Leader Position Planning $(map : M)$

1) Convert map M to occupancy grid and expand obstacles. Beginning at the location of a specified object of interest, perform a tree-like search of potential deployment positions as follows:
2) For the current candidate sensor position (curr_sensor_position):
   - Check to ensure that (curr_sensor_position) satisfies the following criteria:
     - **ADJ_NEARBY_OBSTACLE**: Position is adjacent to nearby obstacle;
     - **AVOID_DOORWAY**: Position is not in a doorway;
     - **MIN_SENSOR_DIST**: Position is at least a minimum distance from previous sensor positions;
     - **ROOM_FOR_LEADER**: Position allows enough space to place two *Leader* robot waypoints (see Figure 2).
   - If curr_sensor_position satisfies these criteria, do following:
     - Add curr_sensor_position to list of deployment positions;
     - Use 5 degree ray-tracing from curr_sensor_position to generate the next set of candidate sensor deployment positions (called "children" of curr_sensor_position) satisfying either the first or both of the following criteria:
       - **WITHIN_SENSING_RANGE**: Position is within sensing range of curr_sensor_position;
       - **WITHIN_LOS**: Position is within line-of-sight of curr_sensor_position;
     - Compute **VISIBILITY_COVERAGE** of candidate positions;
     - Recursively consider candidate deployment positions in descending order of **VISIBILITY_COVERAGE**, favoring deployment positions that meet both **WITHIN_LOS** and **WITHIN_SENSING_RANGE** by returning to step 2).
   - If curr_sensor_position does not satisfy either of these criteria, do following:
     - Consider the next child from the same parent (returning to step 2);
     - If none of the children from the same parent satisfies the criteria, backtrack the tree-like structure to previous parent with remaining potential sensor deployment positions and recursively repeat (returning to step 2) until the entire tree is expanded and no more new sensor deployment positions can be found.

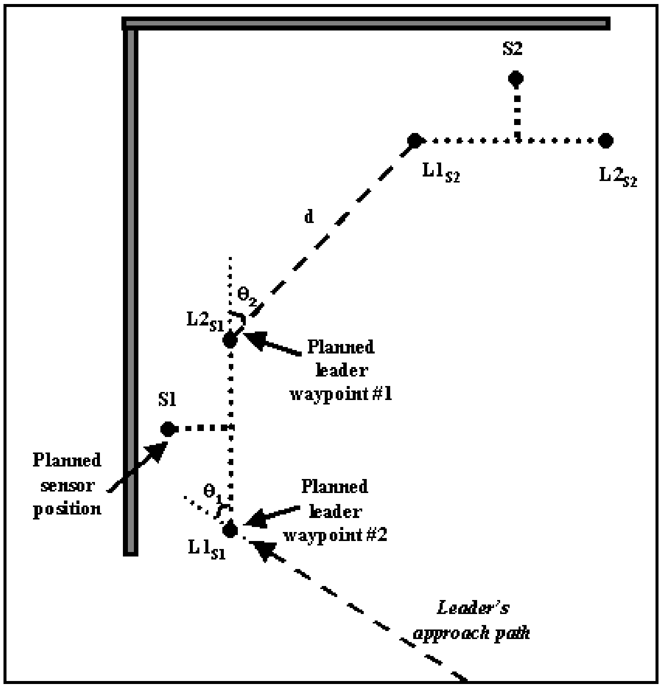

Fig. 2. Relationship between a planned sensor position and the two leader positions, and successive sensor positions in a group. Because the simple sensor nodes are following in a chain behind the *Leader* robot, the *Leader* must first travel through one of the *Leader* waypoint positions, and then the second. This ensures that the following sensor node is in the proper position relative to the planned sensor position and the *Leader*, so that the *Leader* robot can successfully teleoperate the following sensor robot into its planned deployment position.

challenged sensor nodes cannot successfully follow in a chain through paths that take many twists and turns, it is imperative to group deployment positions so that the path taken to visit each position in the group is as smooth as possible. (In practice, the simple sensor robots in a chain have a strong tendency to get caught on doorways or furniture if the *Leader*'s path makes many sharp turns.)

Therefore, we have developed a team assignment algorithm that incorporates a number of turning constraints, toward the objective of ensuring that the path the *Leader* takes to visit the grouped deployment positions is as smooth as possible. Outlined in Table II, this algorithm begins by selecting the sensor position furthest (in terms of robot travel distance) from the Home area, which is the area from which the sensor node robots will be deployed. The algorithm then considers candidate deployment positions to group with the selected sensor on a trial basis. The algorithm searches for the order of visitation of the deployment positions of the candidate team that produces the least amount of turning. Using this possible ordering, the algorithm then determines if this candidate solution satisfies additional turning constraints, as follows. As illustrated in Figure 2, let θ_1 be the turning angle that would be taken by the *Leader* when approaching the first waypoint, $L1_{S1}$, of sensor position $S1$ and turning to the second waypoint, $L2_{S1}$. Let θ_2 be the turning angle that would be taken by the *Leader* leaving the second waypoint of one deployment position, $L2_{S1}$ and turning toward the first waypoint of the following deployment position, $L1_{S2}$. Finally, let d be the distance from the second waypoint of

TABLE II

THE TEAM ASSIGNMENT ALGORITHM

```
The Team Assignment Algorithm $(map : M,\ number\ of\ sensor\ nodes : N,\ maximum\ team\ size : n)$
1) For each sensor node, plan a path from the Home area, $s$, to the node, $g$:
    • $p$ = Path from Dual_Wavefront_Path_Planner$(M, s, g)$.
    • $PL1$ = Path Length($p$).
2) While (assignments $< N$):
    • From the sensor nodes that are not yet assigned to a team, select the node with the greatest $PL1$. Assign this sensor position to a team.
    • For each of the 2 orderings in which the planned leader waypoint positions could be visited, determine if the following criterion is violated (see Figure 2):
        – $\theta_1 <$ MAX_TURN_1
    • If this criterion is violated for both orderings, this sensor node will be deployed individually ($teamsize$ = 1).
    • Else, While $(teamsize < n)$:
        – For all sensor nodes not yet assigned to any team or rejected from this team, plan a path from the most recently assigned sensor position, $s$, to a sensor node not yet assigned or rejected, $g$:
            ∗ $p$ = Path from Dual_Wavefront_Path_Planner$(M, s, g)$.
            ∗ $PL2$ = Path Length($p$).
        – While (assigned nodes + nodes rejected from this team $< N$):
            ∗ Select the node with the smallest $PL2$ that has not been rejected. Assign this node to this team on a trial basis, and increment $teamsize$.
            ∗ Reorganize the team assignment so that the total amount of turning required during deployment, $T$, is minimized:
                · For each of the $teamsize$! possible orderings in which nodes assigned to this team could be deployed, plan a path from the Home area to the first node, then through every other node in order.
                · Select the ordering with the smallest $T$. This is the order in which the sensor nodes assigned to a team are to be deployed.
            ∗ For each of the $2^{teamsize}$ possible orderings in which planned leader positions could be visited, plan a path from the Home area to the first leader position and then through every other leader position in turn, in order to determine if any of the following criteria are violated (see Figure 2):
                · $\theta_1 <$ MAX_TURN_1
                · If $d <$ MIN_DIST, then $\theta_2 <$ MAX_TURN_2A
                · If $d \geq$ MIN_DIST, then $\theta_2 <$ MAX_TURN_2B
            ∗ If any of these criteria are violated, reject this node for assignment to this team, and decrement $teamsize$.
            ∗ Else, permanently assign this node to this team. The planned leader positions will be visited in an order which satisfies the above criteria.
```

one deployment position ($L2_{S1}$) to the first waypoint of the next deployment position ($L1_{S2}$). Then, the enforced turning constraints require the following:

- $\theta_1 <$ MAX_TURN_1
- If $d <$ MIN_DIST, then $\theta_2 <$ MAX_TURN_2A
- If $d \geq$ MIN_DIST, then $\theta_2 <$ MAX_TURN_2B

These constraints were derived through a process of testing the physical sensor robot capabilities for following the *Leader* through various turns. A particular difficulty of the chain formation occurs when the *Leader* turns and stops after only a short distance. In this case, the following sensor node robots tend to cluster and not remain in a chain formation. Thus, these turning constraints ensure that when the *Leader* turns, it continues to move for some distance before stopping.

In examining candidate positions in this manner, all possible orders in which planned leader positions could be visited are considered. Team assignment continues until the maximum team size, n, is reached or there are no more candidate positions. The process then repeats for the furthest unassigned sensor position from the Home area, until all positions have been considered. Throughout this process, the paths and distances between Home and the planned sensor node positions are calculated using a dual wavefront path planning process. Refer to [7] for more details on this path planning approach.

V. RESULTS

Figures 3, 4, and 5 show the results of our planning process for the deployment of navigationally-challenged sensor nodes. All of these maps were developed by physical robots using laser rangefinders. In each of these figures, the starting position for the planning is indicated by the "Object of Interest" label. The Home position, from which the robots are deployed, is indicated by the marking for "Home". The gray squares represent the sensor deployment positions generated by our algorithms. Notice that these positions provide a fairly uniform distribution throughout the environment, and that the sensor positions do not obstruct hallways or doorways. Additionally, these figures show two small dots corresponding to each sensor position, which represent the two planned *Leader* waypoint positions. These waypoints ensure that the *Leader* robot and the following sensor nodes are lined up properly as they reach the deployment position to reduce navigational complexity. These deployment positions are generally arranged on either side of the sensor deployment position along a line perpendicular to the deployment position's contact with its adjacent obstacle (except for wide open areas).

In the environments in Figures 3 and 4, the maximum group size, n, is 3. In Figure 5, n is 2. In all of these figures, the deployment positions that are grouped together are labeled with the same starting character (e.g., 'A', 'B', etc.). The number after this alphabetic character indicates the order of visitation of the deployment positions, starting from Home. Sensor positions with no labels are those positions that are deployed individually. For example, in Figure 4, four groups of three robots are teamed together ((A1, A2, A3), (B1, B2, B3), (C1, C2, C3), (D1, D2, D3)), one group of two robots is teamed together ((E1, E2)), and one sensor position is to be deployed individually (the unlabeled sensor position in the lower left corner). At execution time, from the Home position, a *Leader* robot takes one deployment group of 3 sensor robots and deploys them at position A1, followed by position A2, and then A3, in order. The next *Leader* robot takes a single sensor robot and deploys it to the unlabeled position in the lower left corner. The next *Leader* robot takes another deployment group of 3 sensor robots and deploys them to positions B1, B2, and B3, in order, and so forth. Note that our turning constraints in the grouping of the sensor positions significantly restricts the average number of positions generated per team. However, these constraints are a limitation of the physical robots as reflected in the parameter settings, not a limitation on the grouping algorithm itself.

We have tested our deployment position generation algorithms with various parameter settings. For the experiments reported here, typical parameter settings are as follows: **WITHIN_SENSING_RANGE** less than 7 meters; **ADJ_NEARBY_OBSTACLE** less than 0.25 meters or greater than 3 meters; doorway width ranges from 0.05 meters to 4 meters; distance from doorway ranges from 2 meters down to 0 meters (longer distance required for smaller openings); **MIN_SENSOR_DIST** is 3 meters; **ROOM_FOR_LEADER** requires 1 meter to side of sensor position, 0.85 meters to the front and back, and 0.5 meters from a *Leader* waypoint to an obstacle; **MAX_TURN_1** is 60 degrees, d is 1.5 meters, **MAX_TURN_2A** is 45 degrees, and **MAX_TURN_2B** is 90 degrees.

We have also varied other aspects of these algorithms, such as reducing the ray tracing increment to one degree increment instead of five degrees. However, we found that the generated sensor deployment positions only varied around 5%, while computational requirements jumped an order of magnitude. Based on these results, we judged that the results of the the 5-degree ray tracing increment were sufficient. In these examples, the average running time for sensor and leader position planning is about one minute. The most computationally complex process is the team assignment, which takes from 10 minutes to 45 minutes running on a Dell Linux 2.0 GHz Pentium 4 laptop, depending upon the size of the environment and the grid cell size.

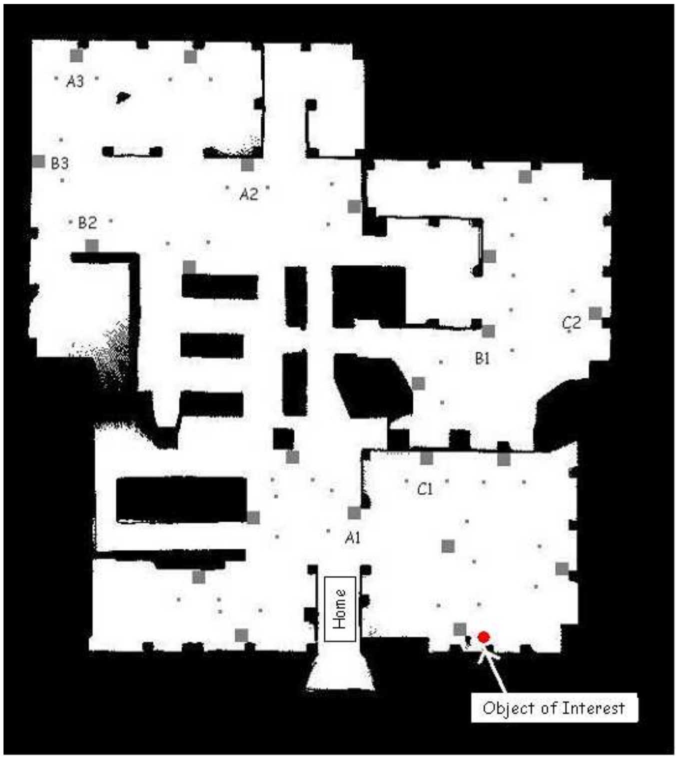

Fig. 3. Results from Environment #1, showing 22 planned sensor positions illustrated as gray squares. The *Leader* waypoint positions are shown as two small gray dots with each sensor positions.

We have successfully executed the deployment plan outlined in this paper on a team of heterogeneous physical robots. Refer to [8] for more details on the deployment process. Figure 6 shows some of the sensor robots deployed according to the plan of Figure 5. Once the robots are deployed, they activate themselves as a distributed acoustic sensor net; refer to [7] for more details on this sensing phase.

VI. RELATED WORK

Besides the previously mentioned work of Howard, et al, [3], other work relevant to this research includes the work of Payton, et al. [9], who 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; virtual pheromones are used to signal a discovery. In their approach, the robots act and communicate locally without the need for a centralized map. A similar approach of swarm robot deployment using potential fields is presented by Howard, et al. [4]. This approach enables a maximal coverage in an unknown area. However, a range-sensing capability is required for the individual sensor nodes, which is not present in our application. Furthermore, the resulting sensor node positions will be equally distanced from obstacles, e.g. in the middle of a hallway, thus hampering the movement of other robots in that area.

Chakrabarty, et al. [1] have developed an approach for deployment in a grid field. However, their method requires

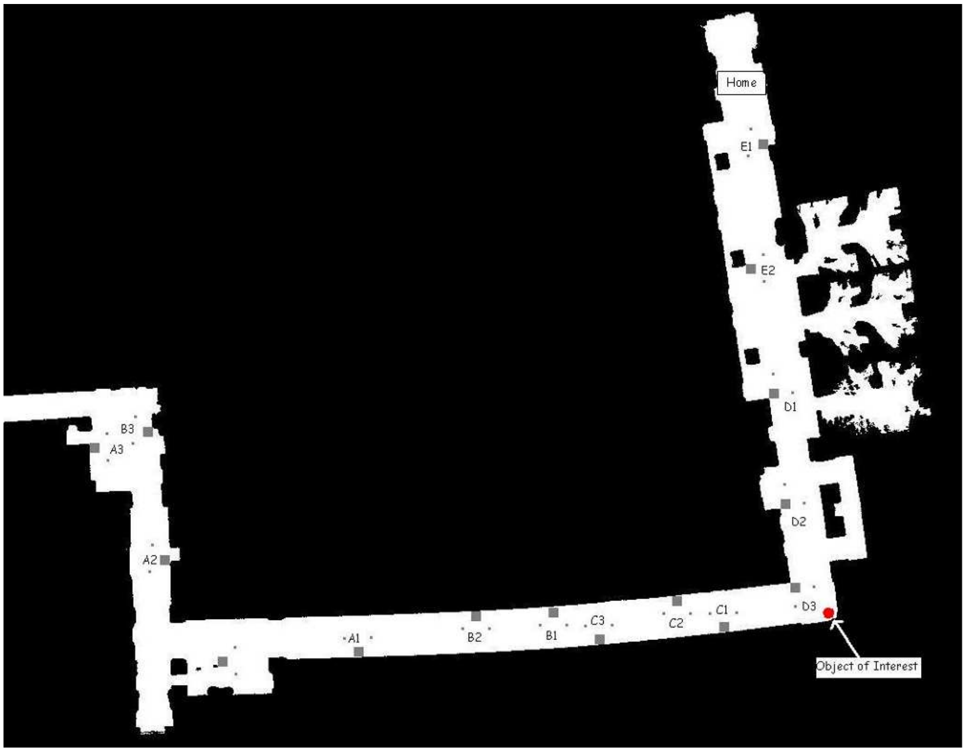

Fig. 4. Results from Environment #2, showing 15 planned sensor positions.

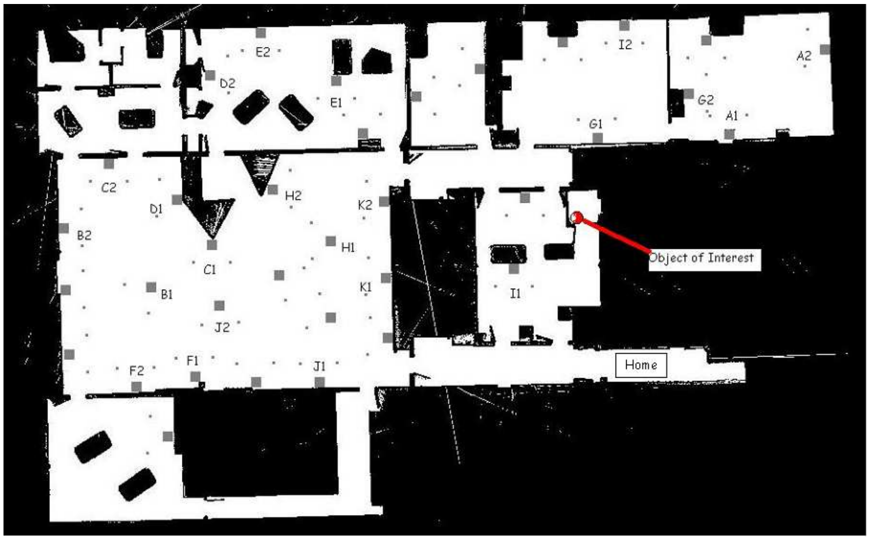

Fig. 5. Results from Environment #3, showing 35 planned sensor positions.

a very large number of sensors and is not suitable for our implementation in indoor environments. Clouqueur, et al. [2], also using grid map, investigate a deployment strategy for target detection over areas of interest. The authors 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 similar algorithm is developed by Meguerdichian, et al. [5]. For our project, we choose to plan all sensor deployment positions at the beginning and to deploy all the sensors at once in order to better coordinate the deployment process.

Fig. 6. Physical robots deployed according to the results of our autonomous planning process for the environment shown in Figure 5. The robot positions shown correspond to the large room in the left-center of the environment in Figure 5.

VII. CONCLUSIONS AND FUTURE WORK

This paper has presented algorithms for planning the deployment of mobile sensor nodes with very limited navigational capabilities. These sensor nodes must be guided to planned deployment positions by more intelligent *Leader* robots. Careful planning is necessary before the deployment process to ensure sensor net coverage, to increase on-line performance, and to reduce failure during the actual deployment process. In our approach, we first plan the sensor deployment positions and the *Leader* robot positions that will be used as waypoints during the deployment process. We then group the sensor positions into smaller teams based on defined turning constraints. We have successfully demonstrated our approach using several different environmental maps. We have also implemented the execution of the deployment plan on a team of heterogeneous physical robots. To our knowledge, this is the first set of algorithms developed for planning the assistive deployment of navigationally-challenged mobile sensor nodes.

In future work, we plan to explore the possibility of combining the two planning processes to perhaps improve upon the computational requirements of the grouping process.

ACKNOWLEDGMENTS

The authors thank Michael Bailey, Balajee Kannan, Chris Reardon, Fang Tang, and Yuanyuan Li for their valuable discussions regarding this research. The authors also thank Andrew Howard and his team of robots from the University of Southern California for generating the maps used in these studies. This research was sponsored in part by DARPA/IPTO's Software for Distributed 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] K. Chakrabarty, S. S. Iyengar, H. Qi, and E. Cho. Coding theory framework for target location in distributed sensor networks. In *IEEE International Conference on Information Technology: Coding and Computing*, April 2001.

[2] Thomas Clouqueur, Veradej Phipatanasuphorn, Parmesh Ramanathan, and Kewal K. Saluja. Sensor deployment strategy for target detection. In *First ACM international workshop on wireless sensor networks and applications*, 2002.

[3] Andrew Howard, Maja J Mataric, and Gaurav S. Sukhatme. An incremental self-deployment algorithm for mobile sensor networks. *Autonomous Robots, Special Issue on Intelligent Embedded Systems*, 13(2):113–126, 2002.

[4] Andrew Howard, Maja J Mataric, and Gaurav S. Sukhatme. Mobile sensor network deployment using potential fields: A distributed scalable solution to the area coverage problem. In *Proceedings of the 6th International Conference on Distributed Autonomous Robotic Systems (DARS02)*, pages 299–308, Fukuoka Japan, 2002.

[5] Seapahn Meguerdichian, Farinaz Koushanfar, Gang Qu, Giacomino Veltri, and Miodrag Potkonjak. Exposure in wireless ad-hoc sensor networks: theory and practical solutions. *Wireless Networks*, 8(5):443–454, September 2002.

[6] L. E. Parker. The effect of heterogeneity in teams of 100+ mobile robots. In A. Schultz, L. E. Parker, and F. Schneider, editors, *Multi-Robot Systems Volume II: From Swarms to Intelligent Automata*. Kluwer, 2003.

[7] L. E. Parker, B. Birch, and C. Reardon. Indoor target intercept using an acoustic sensor network and dual wavefront path planning. In *Proceedings of IEEE International Symposium on Intelligent Robots and Systems (IROS '03)*, October 2003.

[8] L. E. Parker, B. Kannan, F. Tang, and M. Bailey. Autonomous cooperative navigation assistance for sensor-limited mobile robots. In *under preparation*, 2004.

[9] D. Payton, M. Daily, R. Estowski, M. Howard, and C. Lee. Pheromone robotics. *Autonomous Robots*, 11(3):319–324, 2001.

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