Signal Strength Coordination for Cooperative Mapping

Bryan J. Thibodeau, Andrew H. Fagg, and Brian N. Levine
Computer Science Department
University of Massachusetts Amherst
{thibodea,fagg,brian}@cs.umass.edu

***Abstract*— This work addresses the problem of coordinating a team of mobile robots such that they form a connected ad-hoc wireless network while addressing task objectives. Many tasks, such as exploration or foraging, can be performed more efficiently when robots are able to communicate with each other. All or parts of these tasks can be performed in parallel, thus multiple robots can complete the task more quickly than a single robot. Communication and coordination among the robots can prevent robots from duplicating the effort of other robots, allowing the team to address the task more efficiently. In non-trivial environments, maintaining communication can be difficult due to the unpredictable nature of wireless signal propagation. We propose a multi-robot coordination method based on perceived wireless signal strength between cooperating robots for exploration in maze-like environments. This new method is tested and compared to an existing method that relies on preserving a clear line of sight between robots to maintain communication.**

I. Introduction

Teams of cooperating robots have the potential to perform many useful tasks for urban search and rescue, military reconnaissance, and planetary exploration. An important component of cooperation is communication between team members. For tasks where different portions can be accomplished in parallel, such as reconnaissance or exploration, a team of robots can complete the task in a shorter amount of time than a single robot. If a team of robots cooperates and shares information among its members, then the task can be addressed even more efficiently since situations where robots duplicate the effort of other robots can be avoided. An example of this is sharing map information so multiple robots do not explore the same area of an environment [1]. But robots can only exchange information when they can communicate with each other.

Maintaining wireless communication among a team of robots moving through an unknown environment can be a particularly vexing problem. The unpredictable and time-varying nature of signal propagation can make it difficult to determine if two robots will be able to communicate in the near future. Previous work often relies on conservative coordination methods that successfully keep robots in communication with each other, but can over constrain the relative movement of the robots [2, 3]. These methods rely on maintaining a clear *line of sight* between communicating robots, which means that a line drawn between the two robots cannot intersect any other object. Since wireless communication often does not require a line of sight, line of sight constraints can overly restrict the robots' movement. Line of sight methods also require that the distance between communicating robots not exceed some maximum distance. As in [3], we assume this distance is equal to the maximum distance at which one robot can directly sense another robot (with vision, laser, or other appropriate sensor). This can also cause the robot team to be over constrained since the range of wireless communication is often much greater than the range of sensors such as cameras or range finders.

In this paper, we develop a coordination method to maintain communication between robots using the perceived wireless signal strength among robots. Our method allows a robot to address task objectives as long as the wireless signal strength among the robots remains above some threshold. When the signal strength drops below the threshold, one or more robots cease to address task objectives while they take action to increase the signal strength. In simulation, we compare a method that relies on maintaining a line of sight between robots to our method. Compared to coordination methods that require a line of sight between robots, our method allows for greater flexibility since robots are not restricted to configurations where there are no obstacles between them. This greater flexibility leads to a large increase in task performance. In some situations the average time to explore an environment using signal strength coordination is less than $\frac{1}{3}$ of the average time achieved using the line of sight based method. Using our proposed coordination method, robots are more prone to temporary loss of communication with their teammates than if they used a coordination method that enforces a line of sight constraint. Depending on task requirements, these temporary losses of connectivity may be acceptable given the increase in task performance.

II. Related Work

The use of robot teams to explore an initially unknown environment has been the subject of much work [4, 2, 5, 6]. One of the challenges faced in this task is to maintain communication between members of a team. Exploration tasks can be completed more efficiently when robots share information with each other [1]. Communicating robots are also better able to address tasks requiring explicit coordination such as cooperative transport [7, 8], where two or more robots must cooperate to move an object that cannot be moved by a single robot. The problem of maintaining wireless communication among a team of robots has been addressed by constraining robots to be within "line of sight" of each other [2, 3, 9, 10].

This method of coordination is very successful at maintaining communication between robots, but does not allow the robots to take advantage of the fact that wireless signals (such as those used by consumer wireless networking products) typically do not require a direct line of sight between the transmitter and receiver. Thus, the robotic team may be overly constrained and not able to address its task as efficiently as possible.

Clearly it is possible for all robots to share information even if every robot cannot communicate directly (i.e. in one hop) with every other robot in the team. In order to coordinate a team in this manner, some method is required to determine which pairs of robots must maintain direct communication with each other. *Leader-follower* relationships between robots [2, 11, 12, 13, 3] are commonly used to coordinate robot teams. In exploration and formation keeping tasks, when two robots are in a leader-follower relationship, typically the leader is free to make progress towards task objectives, such as exploration, while the follower is restricted to move within some area relative to the leader (e.g. the area in which it can communicate with the leader). Previous work on maintaining communication in a team of mobile robots exploits leader-follower relationships to determine which robots need to maintain a clear line of sight to one another so they can communicate [2]. We will use leader-follower relationships to determine which pairs of robots must maintain wireless links to each other.

Wagner and Arkin [14] propose an approach combining planning and reactive behavior to maintain communication in a team of robots performing a reconnaissance task. In this approach, they use plans designed by hand to help maintain communication in the team. Contingency plans are designed that can be used in the event that wireless communication is close to failing or fails due to insufficient signal strength. Results are provided for teams of up to four robots utilizing various configurations and control schemes. Hand designed plans allow for sophisticated strategies, but require *a priori* map knowledge, and thus are not suitable for exploration tasks in unknown environments.

Powers and Balch [9] describe a method called *Value-Based Communication Preservation* for moving a team of robots to a goal location while maintaining communication among the team members. Control decisions are based upon the perceived and predicted signal strength of communication with neighboring robots. It is assumed for the purpose of control that wireless communication is line of sight only. The results presented by Powers and Balch demonstrate that it is feasible to use the signal strength of wireless signals to maintain communication in a team of robots.

III. Task and Environment Model

A. Task

In this work, we address the task of *cooperative mapping*. The objective of the cooperative mapping task is to explore an initially unknown environment and to map all of the reachable obstacles and free space in that environment. The map is represented by a discrete grid in which each square is marked *freespace*, *obstacle*, or *unexplored*. A grid square is marked as *obstacle* if there is an obstacle in any portion of the world represented by that grid square. Each grid square is 4m × 4m. Even though the map is discretized, robots move continuously though the world. The task is complete when the environment has been explored, i.e., when all grid squares in the map that correspond to regions of the environment accessible from the robots' starting location are marked *freespace* or *obstacle*.

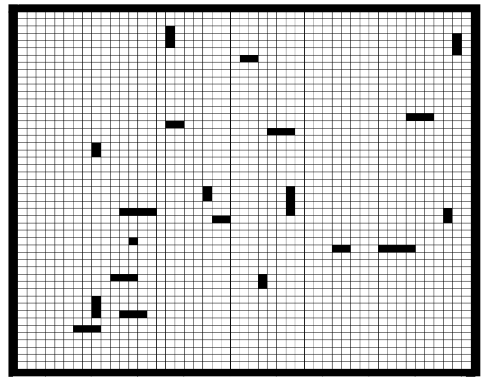

Fig. 1. Example of a sparse environment

In this work, the performance of signal strength coordination is compared to the performance of line of sight coordination in the context of the cooperative mapping task. We observe both the task performance and network connectivity of robot teams of varying sizes addressing the cooperative mapping task in a variety of environments. Since all of the experiments are carried out in simulation, we need to define appropriate models for the environment, robots, and communication between the robots.

B. Environments

All experiments are performed in a $200m$-by-$200m$ square environment. We use *sparse* and *dense* environments for experiments. The sparse environment has 20 randomly placed line-segment obstacles with a randomly chosen length uniformly distributed between $1m$ and $10m$. With equal probability, obstacles are oriented parallel to one of the axes in the environment. The dense environment is similar except it contains 120 line-segment obstacles. Examples of a sparse and a dense environment are shown in Figures 1 and 2. All robots are located in the lower left-hand corner of the environment at the beginning of an experiment.

C. Inter-robot communication

For our experiments conducted in simulation to have significance for real robots communicating wirelessly, we need to take into account the wireless signal propagation characteristics of real signals. It is very difficult to predict how a signal will propagate in an environment, especially if there is no line of sight path between the transmitter and the receiver.

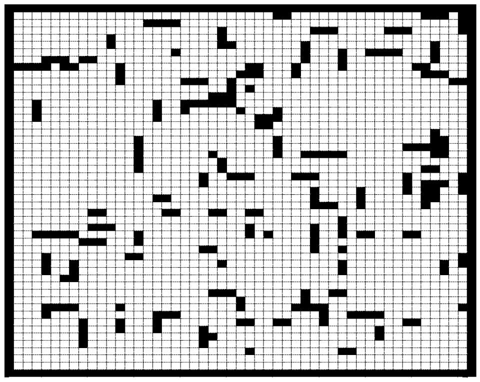

Fig. 2. Example of a dense environment

Thus we must rely on extremely simplified models of signal propagation.

The following assumptions are made about the wireless channel between robots. If the signal between two robots is sufficiently strong, then a link exists. If a link exists, then the robots have enough bandwidth to exchange information regarding their position, and any new map information every simulation time step (once a second). We also assume that the robots can accurately measure the signal strength of any signal they receive and can determine which of their team members they can currently communicate with. Competition for access to the network's physical medium is not modeled.

When referring to the strength of a signal, we do so in terms of the *path loss* that occurs between the transmitter and receiver. Path loss is the amount of power that a signal loses between the transmitter and the receiver measured in decibels (dB). The path loss between a pair of robots depends upon the distance between the robots, the number of obstacles between them, and the properties (such as material or density) of the obstacles between them.

We assume a link exists between two robots when the path loss between them is less than some value R. In order to simulate path loss in an environment with obstacles, we use the following model from Rappaport [15]:

$$\overline{PL}(d) = \overline{PL}(d_0) + 20\log\left(\frac{d}{d_0}\right) + \alpha d + \sum_i PAF_i, \quad (1)$$

where:

- $\overline{PL}(d_0)$ is the path loss in dB at a small distance d_0 from the transmitter,
- $20\log\left(\frac{d}{d_0}\right) + \alpha d$ is the path loss due to the distance the signal must travel in free space to reach the receiver,
- d is the distance between the transmitter and the receiver,
- α is a constant that depends on the type of environment in which the signal is traveling (i.e., office building, warehouse, or outdoors), and
- PAF_i is the *partition attenuation factor* for the i^{th} obstacle between the transmitter and receiver.

The partition attenuation factor of an obstacle is the amount of power a signal loses (dB) by passing through that obstacle and depends on the material and density of the obstacle.

In the model of path loss given in Eq. 1, the path loss smoothly decreases as the receiver moves away from the transmitter, except at the boundaries of shadows cast by obstacles. But even within these shadows, the predicted path loss changes smoothly. In this respect this model does not match the reality of signal propagation. In practice, when there is no line of sight between a transmitter and a receiver, path loss can change radically even for very small physical displacements that do not introduce or remove occlusions between the transmitter and receiver. Also, if no line of sight exists between the transmitter and the receiver, the path loss may vary significantly even for fixed positions of the transmitter and receiver if the environment is not completely static. These behaviors are due to the effects of multi-path propagation, where multiple copies of the transmitted signal reach the receiver at slightly different times and from different directions [15]. Multi-path propagation occurs because objects in the environment reflect and scatter the transmitted signal in ways that can be difficult to predict. When a line of sight exists between the transmitter and the receiver, the signal propagating along the line of sight tends to dominate any multi-path effects, and signal strength is much easier to predict. Multi-path effects are modeled in our simulation by adding Gaussian noise (with mean $\mu \leq 0$) to the model in Eq. 1 only when the transmitter and receiver are not within line of sight of each other. Since multi-path effects are often dominated by propagation along the line of sight, no noise is added to Eq. 1 when the transmitter and receiver are within line of sight of each other. This model will not necessarily predict signal strength fluctuations accurately, rather it is intended to complicate the coordination of a communicating robot team in the same manner that actual multi-path effects would.

D. Communication Parameters

In order to simulate wireless communication in an indoor office environment, the communication parameters in Eq. 1 were set to: $d_0 = 1m, \overline{PL}(d_0) = 30$dB, and $\alpha = 0.35$. These parameters were hand chosen by comparing the path loss predicted by the model at various distances to the path loss predicted at the same distances in the specification for a common 802.11b card [16]. The model does not match the specification in [16] exactly for any environment type, rather it yields path losses that are between those given for *semi-open* and *closed* environments. In most experiments we assume that a signal passing through an obstacle loses 5dB of signal strength. This could be expected from obstacles such as cardboard boxes, storage racks, or other similar objects [15]. These parameters were not empirically verified and may not yield the proper characteristics for any physically realizable signal.

Figure 3 shows the path loss determined by the above model for a transmitter at the center of the environment shown. Lighter shades represent lower path loss and black grid squares contain obstacles. No Gaussian noise was added in this case. In Figure 4, Gaussian noise with $\mu = 0$dB and $\sigma = 5$dB was

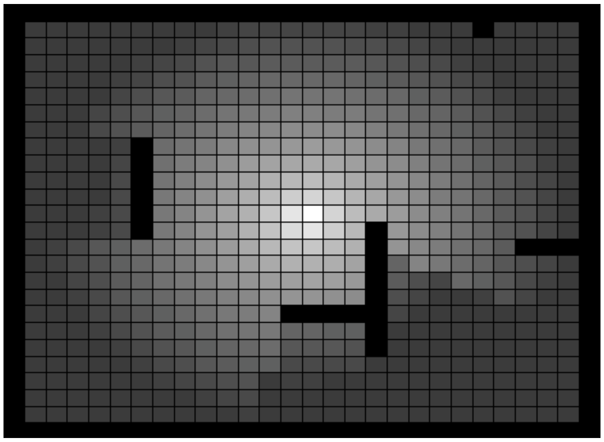

Fig. 3. The path loss (in dB) for a transmitter located at the center of the environment as determined by Eq. 1. The environment shown is 88m $\times$ 96m.

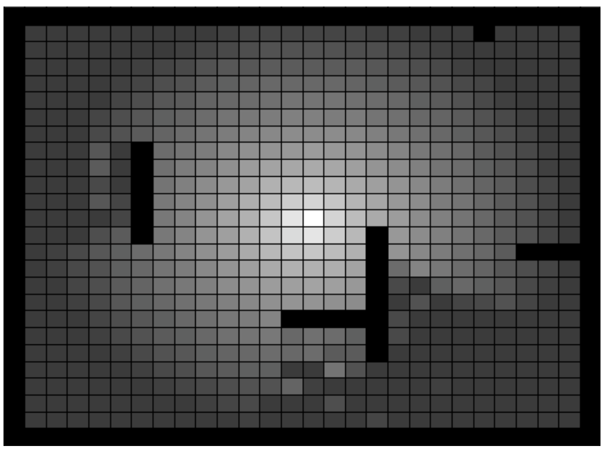

Fig. 4. The path loss (in dB) for a transmitter located at the center of the environment. Here, for grid squares not within line of sight of the center of the environment, Gaussian noise with $\mu = 0$ and $\sigma = 5$dB is added to the path loss determined by Eq. 1. The environment shown is 88m $\times$ 96m.

added to the value given by Eq. 1 for grid squares not within line of sight of the center of the environment.

Two robots can communicate in our simulation when the path loss between them is less than $R = 81.5$dB. Even though actual wireless communication hardware (such as commercial 802.11b equipment) can maintain a link when path loss exceeds 81.5dB, we chose this value as a upper limit on communication range to make communication maintenance sufficiently difficult given the environments that can be reasonably simulated. If there are no obstacles obstructing the signal, a path loss of 81.5dB corresponds to a distance of about $50m$.

E. Ad-hoc network

It is assumed that the robots maintain an ad-hoc network among themselves to the extent that the path loss between them permits. The simulation does not consider the details of such a network, but only determines which subsets of the robot team can currently communicate. To make this determination, a minimum spanning tree is built where the nodes represent robots and the link cost are the path loss between robots. Prim's algorithm [17] is used to build the minimum spanning tree. The number of partitions in the network can be determined by counting the number of links in the minimum spanning tree tree that have a path loss greater than $R = 81.5$dB. It is important to note that the minimum spanning tree does not necessarily represent the complete topology of the ad-hoc network, rather it is used to determine the network connectivity. The minimum spanning tree is also used to determine the leader-follower relationships between robots. The details of constructing the minimum spanning tree will be given when the leader-follower relationships between robots are discussed.

F. Robot Model

The robots are assumed to be holonomic point robots. We assume a vision or range finder sensor that can "see" $S = 8$ meters. If any part of a map grid square is observed, it is assumed that the entire contents of the grid square are observed. Therefore, each robot can detect obstacles and other robots within a range of $8m$. This means that for two robots to be in line of sight they must be no more than $8m$ apart. The robots move at a constant speed of $0.25m$ per time step, where a time step is equal to 1 second. The robots are localized and always know their current position in the world.

IV. Algorithm

We propose a simple coordination method to preserve communication in a team of robots addressing the cooperative mapping task. This coordination method utilizes leader-follower relationships between robots. The leader-follower relationships are determined by a team topology that adapts based upon the path loss between team members. A robot's active controller is determined by the perceived signal strength between that robot and its leader. The purpose of this coordination method is to maintain communication in a team of cooperating robots, while, relative to line of sight coordination, allowing the team more freedom to address task objectives such as mapping the environment.

The team uses the ability to communicate with each other to complete the mapping task more efficiently. When robots can communicate with the team leader, they share map data with the team leader. This allows all team members able to communicate with the team leader to use the same map. When map data is shared, a robot will not try to explore a region that a teammate has already explored.

A. Team Topology

The task of coordinating the robotic team can be simplified by using a team topology with the the following properties. First, the team topology should allow the team to maintain a connected network with each robot using only local information to make control decisions. Specifically, it should be the case that if every follower is in direct communication with its leader, then the network of robots is connected. It is also important that every robot have only one leader.

This simplifies control since multiple leaders could impose conflicting constraints on a follower. Additionally, if a robot has more than one leader, unless those leaders explicitly coordinate their actions, it may not be possible for the follower to maintain communication with all of its leaders, which could cause the network to become partitioned. Another desired property of the team's topology is that there be a robot that is the "team leader." The team leader has no leader (thus it is free to address task objectives) and every other robot should ultimately be a follower of the team leader (i.e. every robot is a follower of the leader, or a follower of a follower of the team leader, etc...). The presence of a team leader helps ensure that the team will always be making some amount of progress toward task objectives.

In order to meet these criteria, the robot team uses an adaptive topology based upon a minimum spanning tree that is updated every k simulation time steps. At the beginning of the task, one member of the team, robot r_0, is chosen as the team leader. r_0 will be the team leader for the remainder of the task. To form the topology for the team, a minimum spanning tree is built where each robot is a node, r_0 is the root node, and the link costs between robots is the path loss of a signal between them. Recall that path loss is the amount of power a signal loses between the transmitter and receiver, and is determined in our simulation environment by the methods described in Section III-C.

We use Prim's algorithm to build the minimum spanning tree [17]. The tree starts as just the team leader, (r_0). At each step of the algorithm, we find the robot, i, not in the tree that has the smallest minimum cost link to any robot already in the tree. Let the robot already in the tree with the minimum cost link to robot i be robot j. Robot i will be added to the tree by adding the link between robot i and robot j to the tree. Robot j will be robot i's leader.

The above algorithm guarantees that every robot, except the team leader r_0, has exactly one leader, and the leader relations propagate such that every robot in the team is ultimately following the team leader. As discussed above, if all of the links in the minimum spanning tree have a cost less than $R = 81.5$dB, the network of robots will be connected. Therefore, if every follower can communicate directly with its leader, then the minimum spanning tree is intact and the ad-hoc network is connected. We have found empirically that updating the topology every $k = 5$ simulation time steps works well since it prevents thrashing when the path loss between various pairs of robots is similar.

B. Harmonic Path Planners

A set of controllers used to realize our coordination method is defined using harmonic path planners. Harmonic path planners generate trajectories using a harmonic function, which is a solution to Laplace's equation. Harmonic functions generate an artificial potential in the robot's configuration space and have a number of properties that make them desirable as path planners. Steepest gradient descent of the artificial potential generated by a harmonic function results in the minimum hitting probability path to a goal location. Harmonic functions are resolution complete and free of local minima [18, 19].

In this work, a robot's configuration consists of its coordinates in a planar world. For the purposes of computing harmonic functions, we will represent configuration space as a discrete grid where every grid square is designated as *freespace*, *goal*, or *obstacle*. Steepest descent of the harmonic potential in this space is guaranteed to result in trajectories that avoid all points designated as *obstacle* and eventually reach one of the grid squares designated as *goal*. Successive over relaxation is used to compute the potentials at each grid square, and bilinear interpolation is then used to compute the gradient at the robot's location. For more details see Connolly and Grupen [18].

Due to issues with numerical precision, in rare cases the gradient of a harmonic function cannot be determined in some portions of the configuration space. This can occur for regions of space that are very far from goals. When a robot cannot determine the local gradient of the harmonic function, it relies on the NF1 navigation function [20] to determine the direction of motion. The NF1 function computes a gradient based on the Manhattan distance from a grid square in configuration space to the nearest goal in configuration space. In very rare cases, the harmonic function can cause dead-lock in a simulation due to the necessity of moving in discrete steps.

C. Controllers

Controllers using harmonic path planners are used to generate the different robot behavior necessary for completing the cooperative mapping task and for maintaining communication. We use two controllers to generate motions for our robots: one that moves a robot into a region where it is in line of sight of another robot; and a second that causes a robot to move toward unexplored areas of the environment. Both of these controllers use a harmonic path planner as described above and differ only in how they define *goals* in configuration space..

We describe controllers using the notation ϕ_i^g, where:

- ϕ is an artificial potential;
- g is sensory information used to determine the shape of ϕ;
- i is a set of effectors used to descend ϕ.

g may refer to sensory information at any level of abstraction. We use sensory abstractions at the level of configuration space maps for specific objectives. We use harmonic functions to generate the artificial potential ϕ, or in cases where the local gradient of the harmonic function cannot be determined, ϕ is determined using the NF1 function. Our effectors always consist of single robots.

The controllers used are similar to those described by Sweeney, *et al.* [2, 21]. Robot r uses the controller $\phi_r^{EXP_r}$ for exploration. The sensory abstraction EXP_r marks all unobserved grid squares as *goal*, all observed grid squares containing obstacles as *obstacle*, and all other observed squares as *freespace*. A grid square is considered observed if robot r directly sensed the grid square itself or was informed about the contents of that grid square by another robot. The boundaries of the configuration space are always designated as *obstacle*. $\phi_r^{EXP_r}$ will generate trajectories that avoid obstacles and move the robot toward unobserved areas of the world.

We use the controller, $\phi_j^{LOS_i}$, to bring robot j to a location where it is in line of sight of robot i ($i \neq j$). The sensory abstraction LOS_i marks all known obstacles (and the boundaries of the configuration space) as *obstacle*. Grid squares that are within some distance S of robot i, do not contain an obstacle, and are within line of sight of robot i are marked as *goal*. Thus, $\phi_j^{LOS_i}$ moves robot j toward the region of space within line of sight of robot i; if robot i is stationary, robot j is guaranteed to reach this region of space.

D. Coordination Methods

To achieve the desired robot behavior using the above controllers, we will combine multiple controllers using a technique inspired by null space control. In systems with excess degrees of freedom, subordinate tasks can be addressed in the null space of superior tasks. Thus, one can guarantee that subordinate tasks will not effect the performance of superior tasks. The Moore-Penrose pseudoinverse is commonly used to invert under-constrained linear systems and provides a natural means of superimposing secondary tasks in the remaining null space. [22]. In general, null space control allows multiple goals to be addressed concurrently.

When tasks are defined using n dimensional artificial potentials, a unique (one dimensional) gradient direction can be computed locally. The $n-1$ orthogonal subset of the potential manifold describes the null space of the potential field - a space in which subordinate actions do not alter the potential underlying the superior controller. Using this notion of a null space we can create compositions of controllers where the actions of subordinate controllers do not effect the progress of superior controllers [23, 24]. An operator accomplishing the null space projection has been constructed called the "subject-to" ($\triangleleft$) operator[23] to compose controllers that descend artificial potentials. For controllers, ϕ_α and ϕ_β, $\phi_\beta \triangleleft \phi_\alpha$ (read "ϕ_β *subject-to* ϕ_α") means that the actions of ϕ_β are projected onto the equipotential manifold of controller ϕ_α's artificial potential. Thus, the actions generated by ϕ_β do not interact destructively with the progress of ϕ_α toward its minimum.

In this work we will use the "subject-to" operator to compose controllers in a way that approximates null space control. In particular, we consider systems that must preserve the equilibrium status of primary controllers while addressing secondary gradients. We use $\phi_\beta \triangleleft \phi_\alpha$ to mean that when the system is in a goal state of ϕ_α, ϕ_β will be used to generate motion commands. When the system is not in a goal state of ϕ_α, then ϕ_α is used to generate motion commands exclusively. This method of control composition requires that superior goals have been met before subordinate goals are addressed and allows the subordinate controller to disturb the superior controller within bounds.

Leader-Follower Relationship for Line of Sight Coordination:
Under line of sight coordination, robot f uses $\phi_f^{EXP_f} \triangleleft \phi_f^{LOS_l}$, where robot l is robot f's leader, for control. This means that robot f will explore the environment as long as it is within line of sight of robot l. When robot f is not within line of sight of robot l, robot f uses $\phi_f^{LOS_l}$ to move toward the region where it would be within line of sight of robot l. For the team leader, $\phi_f^{LOS_l}$ is undefined (because the team leader has no leader), and $\phi_f^{EXP_f}$ is always used for control.

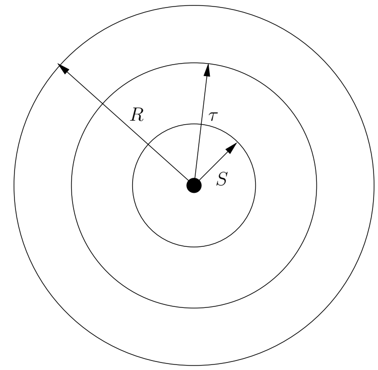

Fig. 5. S is the distance (in meters) in which a robot can "see" obstacles or other robots; R is the range (in dB path loss) of wireless communication between robots; τ (in dB path loss) is the threshold at which a follower robot activates $\phi_f^{LOS_l}$ when using signal strength control. Note that path loss cannot be directly translated to meters without taking into account specific environmental features, but τ is chosen such that in almost all situations τ will correspond to a further distance in meters than S.

Leader-Follower Relationship for Signal Strength Coordination:
For signal strength coordination, we will need to define one more controller. Let $\phi_f^{SIG_l}$ be a controller that is the same as $\phi_f^{LOS_l}$, except the goal region generated by SIG_l is all points in configuration space where the path loss to robot l is less than some threshold τ. τ is always less than R, the maximum path loss at which communication is still possible. For signal strength coordination, robot f uses $\phi_f^{EXP_f} \triangleleft \phi_f^{SIG_l}$ for control. This means that robot f will explore the environment as long as the path loss to robot f's leader, robot l, is less than τ. Otherwise, robot f will move toward the region of space where the path loss to robot l is less than τ.

As discussed above, it is very difficult to predict the path loss for arbitrary locations of a transmitter and receiver. Therefore it is not feasible to compute the configurations of robot f where the path loss from robot l to robot f is less than τ. But, it is still possible for robot f to directly sense (by measuring the strength of the signal from robot l) whether or not it is in a goal state of $\phi_f^{SIG_l}$. Because we cannot compute the goal set of $\phi_f^{SIG_l}$, when robot f is not in a goal set of $\phi_f^{SIG_l}$ (i.e. when the path loss between robots f and l is greater than τ), $\phi_f^{LOS_l}$ will be used for control. $\phi_f^{LOS_l}$ is not guaranteed to monotonically decrease the path loss between robots l and f. But, as long as τ is greater than the path loss for an unobstructed signal traveling distance S (the maximum separation of two robots that are within line of sight), the path loss between robots l and f will be less than τ for all configurations in the goal set of $\phi_f^{LOS_l}$. Thus $\phi_f^{LOS_l}$ is a conservative approximation of $\phi_f^{SIG_l}$, and will in general decrease the path loss between robots l and f.

It is possible for either of the coordination methods to fail to keep every leader-follower pair in contact. When a follower loses contact with its leader, the follower moves toward the position the leader was at when communication was last possible. The only other robot who's behavior changes is the team leader, which immediately stops whenever the network of robots becomes partitioned. By having the team leader remain in place, we can guarantee that communication with the team will eventually be restored. If the robot that lost communication with its leader reaches the last known position of its leader without reestablishing contact with any team member, it will then move to the position of the team leader, which hasn't moved since communication was lost. Since we assume the environment is static, that all of the robots are localized, and that movement is error free, the disconnected robot will eventually establish communication with the team leader if it does not encounter any other member of the team first.

V. RESULTS

In this section, we present results demonstrating the performance of the signal strength coordination method in a variety of conditions and compare the perfomance of the signal strength coordination method to the performance of the line of sight coordination method. The experiments are designed to test the scalability of the coordination methods, the robustness of the coordination methods to various environmental factors, and the sensitivity of signal strength coordination to algorithmic parameters. The adaptive topology is also compared to a number of fixed topologies to empirically verify its performance. Experiments were performed with teams of 2, 4, 8, 16, and sometimes 32 robots in both the *sparse* and *dense* environments defined above. We used the same 25 randomly generated instances of each environment for every experiment. In the following graphs, each bar represents the average of 25 trials, one in each instance of the appropriate environment type. In this work, when the network of robots was partitioned, robots not connected to the team leader were allowed to change their leaders (the topology was still adaptive). We discovered that in rare cases this lead to a live lock situation where the partitions would never merge and all task progress would stop (since the team leader is prevented from moving when the network is partitioned). In cases where this occurred, the experiment timed-out at $20,000$ time steps. This problem will be remedied in future experiments by keeping the topology static in network partitions that do not contain the team leader. A trial that deadlocked due to the problem with discrete motions along a harmonic gradient timed-out at $20,000$ time steps as well.

A. Number of Robots

We first conducted experiments to determine how team size effects the performance in teams using either the line of sight or signal strength coordination methods. This is a crucial metric for algorithms designed for robot teams, since a good coordination method for a team of robots should make efficient use of all members of the team.

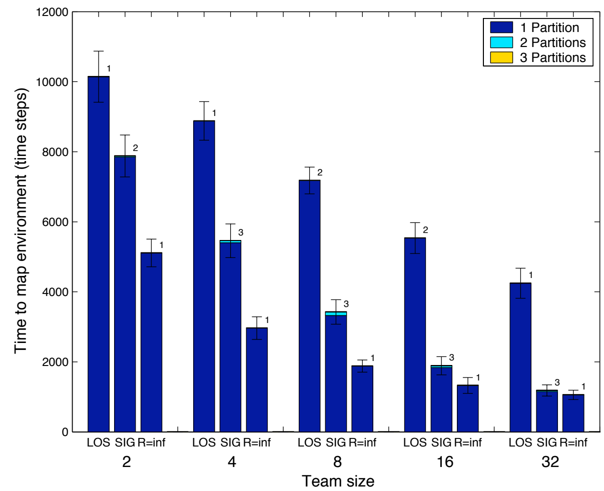

Fig. 6. **Mean number of time steps to map sparse environment versus team size**, broken down by number of partitions in the network. "LOS" indicates the line of sight coordination method, "SIG" denotes the signal strength based coordination method, and "R=inf" denotes the case where communication range is assumed to be infinite and signal strength based coordination is used. The error bars represent one standard deviation, and the numbers to the upper left of each bar indicate the maximum number of network partitions present at one time in any of the 25 trials represented by the bar.

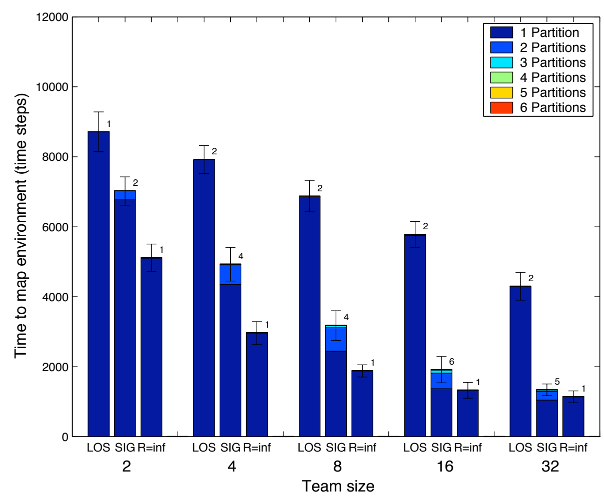

Fig. 7. **Mean number of time steps to map dense environment versus team size**, broken down by number of partitions in the network. "LOS" indicates the line of sight coordination method, "SIG" denotes the signal strength based coordination method, and "R=inf" denotes the case where communication range is assumed to be infinite and signal strength based coordination is used. The error bars represent one standard deviation, and the numbers to the upper left of each bar indicate the maximum number of network partitions present at one time in any of the 25 trials represented by the bar.

The average time to fully search the maze using both the line of sight and signal strength coordination methods is shown in Figures 6 and 7. For signal strength coordination we set $\tau = 73.7$dB. The bars labeled “LOS” correspond to line of sight coordination, and the bars labeled “SIG” correspond to signal strength coordination. The bars labeled “R=inf” correspond to the case where signal strength coordination is used and communication range is unlimited (τ is also set to infinity in this case). This case is included to provide a lower bound on the search time.

The shading of each bar represents the number of partitions in the network. For example, for sixteen robots using signal strength coordination in the dense environment, the network has one partition (i.e., it is fully connected) for about 1400 time steps; for about 450 time steps there were two partitions; for roughly 85 time steps there were three partitions; and for less than 1% of the total number of time steps there were four, five, or six partitions in the network. The total number of time steps in which the group contained more than three partitions is so low that the corresponding regions in the graph are not visible. The maximum number of partitions that occurred during any of the 25 trials is indicated by the number at the upper right of the bar.

The results show that the signal strength coordination method outperforms the line of sight coordination method in every case ($p < 1.5 \times 10^{-12}$)[1], but line of sight coordination is slightly more successful at keeping the network connected. This is to be expected since the signal strength coordination method allows the robots to spread out more than the line of sight coordination method does, increasing the amount of the environment that can be covered in parallel, but also increasing the chance that the network connecting the robots will become partitioned. The search times are lower in general for the dense environment since it has a smaller area to be searched than the sparse environment.

Our results indicate that signal strength coordination benefits much more from additional robots than line of sight coordination does. When line of sight coordination is used, teams of 32 robots complete the search task approximately 2.0-2.4 times quicker than teams of two robots. When signal strength coordination is used, teams of 32 robots complete the search task approximately 5.2-6.7 times quicker than teams of two robots. The signal strength coordination method makes better use of additional robots since it allows robots to disperse further, increasing the likelihood that a robot is observing a part of the environment that has not been observed by another robot. Also, these results demonstrate that it is more difficult to maintain a connected network in the dense environment due to the additional obstacles. The additional obstacles will tend to increase the chance that a motion could cause a large change in path loss (by introducing one or more obstacles between the transmitter and receiver) which makes it more difficult for the network to remain connected.

[1]The paired t-test assumes that the differences between pairs of elements from two different samples are normally distributed. Here, the pairs consist of trials performed in the same environment, but with different coordination methods/parameters. The paired t-test determines the probability p that the mean difference between pairs is not 0.

The cases where the communication range is assumed to be infinite (R=inf) provide an upper bound on the performance of any coordination method given the particular search strategy employed. Figures 6 and 7 show that as the number of robots increases (and particularly for the cases where $n = 16$ or 32) the performance of signal strength coordination approaches the performance achieved when the communication range is assumed to be infinite. The difference in performance between signal strength coordination and the “R=inf” case is always statistically significant in Figures 6 and 7 ($p \leq 0.0042$ for any number of robots in either environment). But the performance of signal strength based coordination is close to optimal in these cases given the chosen search strategy since no method for maintaining communication (that doesn’t involve changing the search strategy) should be able to improve over the case where the communication range is infinite.

B. Signal strength threshold

The signal strength coordination method requires choosing a value for the threshold τ that determines when robots will switch from the $\phi_r^{EXP_r}$ controller to the $\phi_j^{LOS_i}$ controller. In order to evaluate the signal strength coordination method we need to find both the optimum value for τ and how sensitive task performance and network connectivity are to different choices of τ. Finding the optimum value for τ in a given situation demonstrates the performance of the signal strength coordination method. If the method is to be used in practice, we need to know how sensitive performance is to variations in τ. If the sensitivity is too high then it may be difficult for a system designer to pick an appropriate value for τ, especially if the environment is unknown, which would reduce the utility of the method.

Experiments were performed in both the sparse and dense environments, varying the value of τ. Since communication is broken when the path loss exceeds 81.5dB, we only consider values of τ less than or equal to 81.5dB. We found that values of τ around 81.5dB tend to cause teams to become highly partitioned and resulted in search times so high that simulations would not finish in a reasonable amount of time. For this reason we choose 80.5dB as the highest value of τ to test. 59.7dB was chosen as the lowest value of τ to test.

Figures 8 and 9 show the search times in the sparse and dense environment respectively. Each line in these graphs corresponds to a particular team size, and shows the change in search performance as τ is increased. The leftmost point of each line shows the performance of the line of sight method. The graphs indicate that performance is similar for a large range of possible values for τ. The graphs also indicate that values of τ that are too too low can adversely affect the task performance. Figures 8 and 9 seem to suggest that values of τ that are too high can also adversely effect task performance, though in each case there is no statistical significance between the low point of a curve and the right most end of the curve

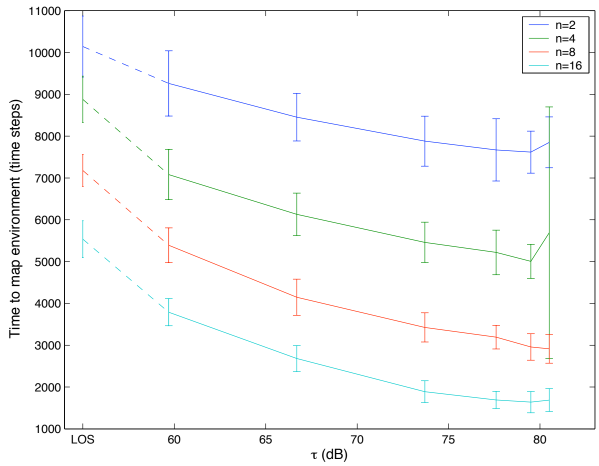

Fig. 8. **Mean number of time steps to map sparse environment versus τ.** τ is the threshold that determines when the controller $\phi_f^{LOS_l}$ is used in the signal strength control method. Each line shows the search time for a specific number of robots (2, 4, 8, or 16) as τ increases. The leftmost point of each line shows the performance of the line of sight method for the corresponding number of robots. For four robots and $\tau = 80.5$dB, one trial timed-out due to the previously discussed problem with the dynamic topology.

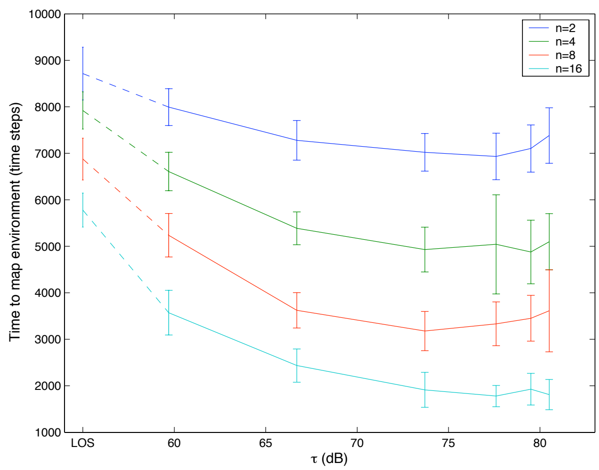

Fig. 9. **Mean number of time steps to map dense environment versus τ.** τ is the threshold that determines when the controller $\phi_f^{LOS_l}$ is used in the signal strength control method. Each line shows the search time for a specific number of robots (2, 4, 8, or 16) as τ increases. The leftmost point of each line shows the performance of the line of sight method for the corresponding number of robots.

($\tau = 80.5$).[2]

In the environments tested, τ can be set around 77dB with essentially no adverse effect on task performance. This removes the need to carefully tune the algorithm for different scenarios. It is possible that for certain environments, performance is more sensitive to the value of τ; this will be the subject of future work.

These experiments also demonstrate how the value of τ effects connectivity in the network of robots. Figures 10 and 11 display the time to map the environment for each team size and value of τ, broken down by the number of partitions in the network. We can see from Figure 10 that τ does not have a large effect on network connectivity in the sparse environment. The network is partitioned only slightly more often as τ increases, which is expected since a higher value of τ allows the robots to disperse more. On the other hand, Figure 11 shows that in the dense environment the network connectivity is effected significantly by the value of τ. This suggests that in the dense environments, network connectivity may be more sensitive to the value of τ than task performance. Thus, in some cases, τ may need to be chosen more carefully if network connectivity is a high priority.

C. Obstacle Composition

The partition absorption factor, or PAF, of an obstacle determines the amount of path loss that occurs due to that obstacle being between the transmitter and receiver. So far, we have only considered obstacles with a PAF of 5dB. In real environments, robot teams are likely to encounter obstacles with a large range of PAFs, since the PAF of an obstacle is dependent upon the obstacle's composition and thickness. The sum of the PAFs for all obstacles between the transmitter and receiver appears as the term $\sum PAF$ in the path loss model given in Eq. 1.

For this set of experiments, the mapping task was performed in the dense environment with a team of 8 robots with the PAF of every obstacle set to 5, 10, 15, 20, or 25dB. This range of PAFs covers many different materials. As stated before, a PAF of 5dB might be expected from obstacles such as empty cardboard boxes or storage racks, whereas PAFs from 20-25dB might be expected from concrete block walls or metal obstacles [15]. The results of these experiments are shown in Figure 12.

The performance of the line of sight coordination method is nearly constant for all of the values of the PAF. When the robot team uses line of sight coordination, there is almost never more than one obstacle between a transmitter and a receiver, and as soon as an obstacle is introduced between a transmitter and receiver (a leader-follower pair), the follower takes action to reestablish line of sight. Therefore, the PAF of the obstacles is not expected to have a large effect on the line of sight method.

In general, the performance of the signal strength based coordination method degrades as the PAF increases. There is a small increase in performance when the PAF goes from 20dB to 25dB, but this difference is not statistically significant. The task performance of the signal strength method degrades because the higher PAF limits how far the group can disperse in the presence of obstacles. Also, a higher PAF implies that motions introducing an obstacle between a transmitter and receiver are more likely to increase the path loss for that link beyond the point where communication is possible. This presents more of a problem for signal strength coordination

[2]To avoid problems arising from multiple comparisons [25], the statistical significance of the difference between the low point on the curve and the right most point on the curve is confirmed in an additional 25 trials that are independent of the 25 trials used to make the graphs shown. This is accomplished by performing the experiments in 25 independently generated environments. For example, comparing the $\tau = 77.6$dB and $\tau = 80.5$dB cases in dense environment yields $p < 0.0009$ in the initial data set, but comparing the same two cases in the independently generated data set yields $p < 0.1077$. Accordingly, we cannot claim the difference is significant.

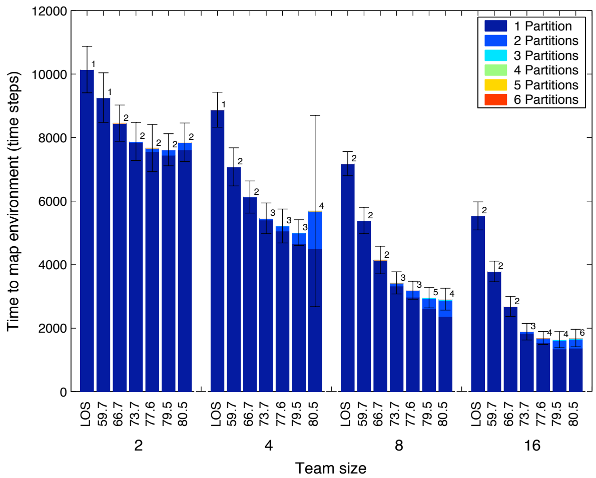

Fig. 10. **Mean number of time steps to map sparse environment versus τ .** τ is the threshold that determines when the $\phi_f^{LOS_l}$ is used in the signal strength control method. The bars are grouped by number of robots, and each bar is labeled with its corresponding threshold value τ, or "LOS" for line of sight control. For four robots and $\tau = 80.5$dB, one trial timed-out due to the previously discussed problem with the dynamic topology. The error bars represent one standard deviation, and the numbers to the upper left of each bar indicate the maximum number of network partitions present at one time in any of the 25 trials represented by the bar. The data used to produce this graph is the same as that used to produce Figure 8.

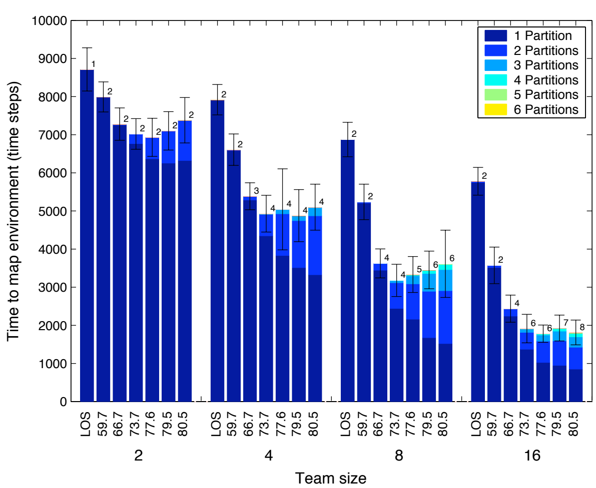

Fig. 11. **Mean number of time steps to map dense environment versus τ .** τ is the threshold that determines when the $\phi_f^{LOS_l}$ is used in the signal strength control method. The bars are grouped by number of robots, and each bar is labeled with its corresponding threshold value τ, or marked as LOS for line of sight control. The error bars represent one standard deviation, and the numbers to the upper left of each bar indicate the maximum number of network partitions present at one time in any of the 25 trials represented by the bar. The data used to produce this graph is the same as that used to produce Figure 9.

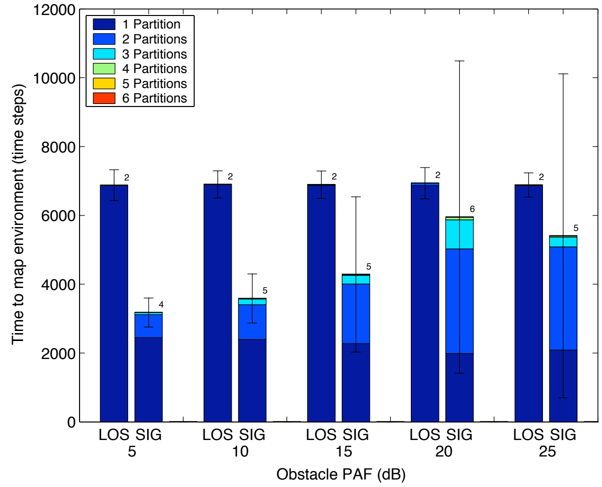

Fig. 12. **Performance as obstacle PAF increases:** A comparison of the line of sight method and the signal strength method as the power absorbed by obstacles varies. All experiments were done with a team of 8 robots in the dense environment. One trial in the 20dB case and two trials in the 25dB case timed-out due to the previously discussed problem with the dynamic topology. The error bars represent one standard deviation, and the numbers to the upper left of each bar indicate the maximum number of network partitions present at one time in any of the 25 trials represented by the bar.

than for line of sight coordination. Under line of sight coordination, the robots are relatively close together and are likely experiencing little path loss (for an unobstructed pair of robots with a separation of $S = 8m$, the path loss between the robots is around 51dB). Introducing one obstacle with a PAF of 25dB between a pair of robots in this case is not very likely to cause the threshold to exceed $R = 81.5$dB. In the case of signal strength coordination, since the robots are often further apart spatially then allowed under line of sight coordination, the path loss between pairs of robots may be much closer to R before an obstacle is introduced between the robots. In this case, the introduction of an obstacle with a PAF of 25dB may be very likely to cause the path loss between the pair of robots to exceed R, especially if $R - \tau < 25dB$.

For the cases where PAF is 20dB or 25dB, the performance of signal strength coordination is quite close to that of line of sight coordination. In the both the 20dB and 25dB case, there is no statistically significant difference between the performance of line of sight coordination and signal strength coordination ($p < 0.2934$ for 20dB, and $p < 0.126$ for 25dB). There is a statistically significant difference in the 5,10, and 15dB cases ($p < 2.7 \times 10^{-6}$). Part of the reason for the relatively poor performance of signal strength coordination in these cases is that one or more trials timed out at 20,000 time steps due to the previously mentioned issue with the adaptive topology in both the 20dB and 25dB cases. Additionally, when the PAF is higher, it may be the case that the performance of the signal strength based method is more sensitive to the value of τ. For these experiments, τ was set to 73.7dB which may not be the optimal value. Exploring this issue will be the subject of future work.

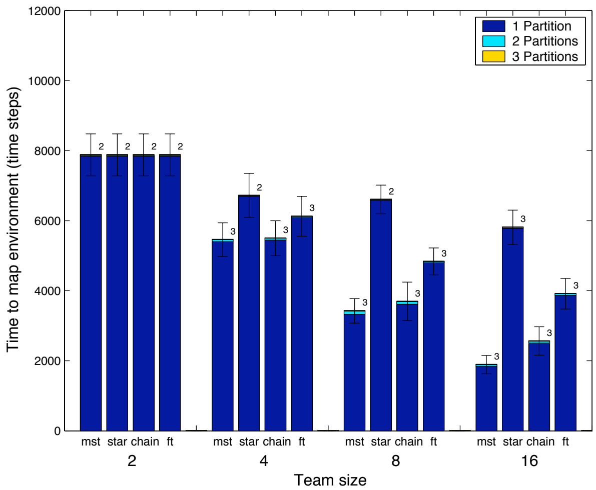

Fig. 13. **Performance of topologies using signal strength coordination in the sparse envrionment:** A comparison the "minimum spanning tree" (mst), "star", "chain", and "fixed tree" (ft) topologies using the signal strength method in the sparse environment with $\tau = 73.7$dB. The error bars represent one standard deviation, and the numbers to the upper left of each bar indicate the maximum number of network partitions present at one time in any of the 25 trials represented by the bar.

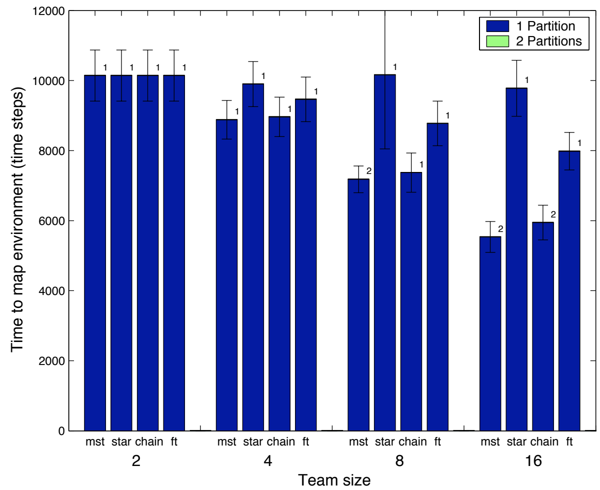

Fig. 14. **Performance of topologies using line of sight coordination in the sparse envrionment:** A comparison the "minimum spanning tree" (mst), "star", "chain", and "fixed tree" (ft) topologies using the line of sight method in the sparse environment. For four robots in the "star" topology, one trial timed out due to issues regarding the use of harmonic functions in simulation. The error bars represent one standard deviation, and the numbers to the upper left of each bar indicate the maximum number of network partitions present at one time in any of the 25 trials represented by the bar.

D. *Significance of fluid topologies*

To determine the effectiveness of the flexible signal strength based team topology, we perform experiments where the leader-follower relationship between robots is held fixed and compare the results to those of obtained with the flexible topology. Fixed leader-follower relationships means that robots do not change their leaders and will always try to maintain a path loss less than τ or a line of sight to their leader, depending on the coordination method employed. We assume that the network topology is still ad-hoc and is determined in the same way as in all previous experiments. This means that for a fixed topology it is still possible for the network to be fully connected even when one or more leader-follower pairs is not in direct communication. Thus, having every leader-follower pair able to communicate directly is sufficient, but not necessary, for the network to be connected. We chose to allow network routing to remain flexible in order to find the maximum possible performance of the fixed topologies.

We tested three different fixed topologies. In a *star* topology all robots (except the team leader) are followers of the team leader. In a *chain* topology, the robots are arranged in a leader-follower chain where every robot except the one at the end of the chain has exactly one follower. In a *fixed tree* topology, the robots form a tree with a branching factor of two, with the team leader as the root of the tree. Each robot is a follower of its parent in the tree. We tested these fixed topologies in the sparse and dense environments, using both signal strength coordination (with $\tau = 73.7$dB) and line of sight coordination. The results are shown in Figures 13-16.

For a team size of two, all of the topologies perform equivalently which is expected since all of the topologies are equivalent for just two robots. As the number of robots increases, the adaptive minimum spanning tree and chain topologies outperform the star and fixed tree topologies. Since the star topology does not allow robots to disperse very far from the team leader, additional robots do not result in a large increase in task performance. This effect is more pronounced when signal strength coordination is used. For the same reason though, the star topology maintains better network connectivity than the other three topologies. One reason that the fixed tree topology performs worse than the chain or minimum spanning tree topologies could be that the maximum separation between any two robots (not necessarily in a direct leader-follower relationship) is much less in the fixed tree topology than in either the chain or minimum spanning tree topology. The disparity in maximum separation grows as the team size increases. This restricts the amount of area that the team in a fixed tree topology can cover simultaneously. Also, each "parent" robot in the fixed tree topology places the same constraints on its two "child" robots. This can cause the two "child" robots to remain close together and to search the same area of the environment as each other. This is clearly not an efficient use of resources and contributes to the poor performance of the fixed tree topology.

In all of the situations examined, the minimum spanning tree topology performs as well as, or better than, the chain topology. There is a statistically significant difference ($p < 0.0053$) between the performance of the adaptive minimum spanning tree and the fixed chain topology for teams of 16 robot in all four scenarios. There is also a significant difference ($p < 0.0280$) between the performance of the minimum spanning tree topology and the chain topology for teams of 8 robots, in either environment, when signal strength coordination is used. Observations of teams using the adaptive minimum spanning tree topology show that the resulting topology is often a chain,

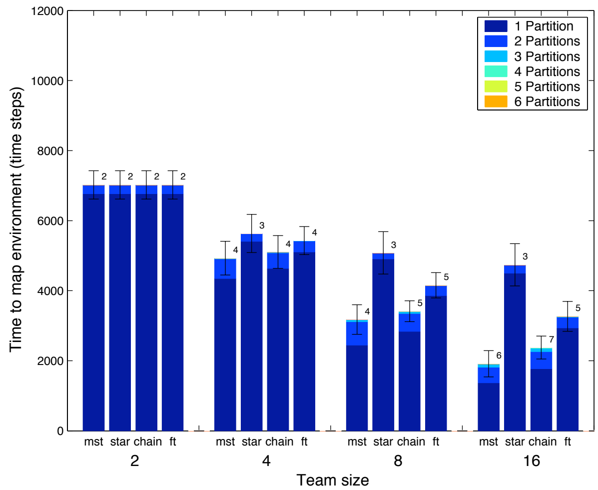

Fig. 15. **Performance of topologies using signal strength coordination in the dense envrionment:** A comparison the "minimum spanning tree" (mst), "star", "chain", and "fixed tree" (ft) topologies using the signal strength method in the dense environment with $\tau = 73.7$dB. The error bars represent one standard deviation, and the numbers to the upper left of each bar indicate the maximum number of network partitions present at one time in any of the 25 trials represented by the bar.

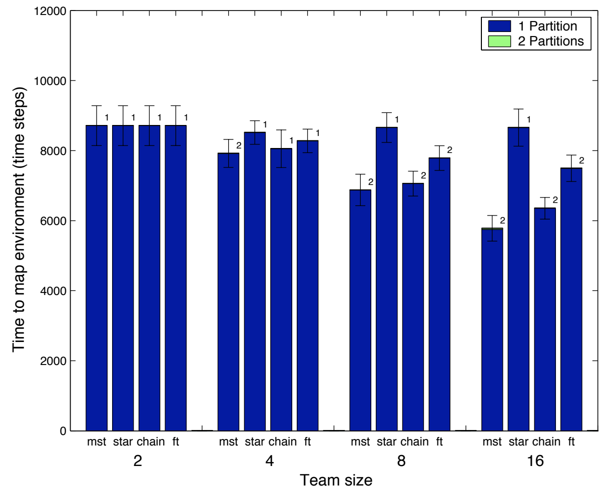

Fig. 16. **Performance of topologies using line of sight coordination in the dense envrionment:** A comparison the "minimum spanning tree" (mst), "star", "chain", and "fixed tree" (ft) topologies using the line of sight method in the dense environment. The error bars represent one standard deviation, and the numbers to the upper left of each bar indicate the maximum number of network partitions present at one time in any of the 25 trials represented by the bar.

or a formation that is close to a chain, especially when line of sight coordination is used. This explains why the two topologies are so close in terms of performance. Observations also indicate that in some situations, the chain topology can force follower robots to take the same path around obstacles as their leaders. This diminishes the effectiveness with which the follower robots address the mapping task. This behavior is not as prominent when the minimum spanning tree topology is used. This may partially account for the difference in the performance of the two topologies.

VI. Discussion

In this paper we have demonstrated, in simulation, the effectiveness of our signal strength coordination method. Compared to line of sight coordination, signal strength coordination completes mapping tasks much quicker, with usually minimal disruptions in network connectivity among the robots for teams ranging in size from 2 to 32 robots. Our experiments suggest that signal strength coordination does not require extensive tuning in order to perform well in a variety of situations, which is important if teams using this method are to be deployed in unknown environments. A network topology that adapts based on the signal strength between pairs of robots was introduced. We have shown that such an adaptive topology aids the performance of a team using signal strength coordination when compared to a variety of static formations. This topology, combined with signal strength coordination provides a way of allowing members of a team to be in near constant communication while still maintaining a high level of task performance.

There is still much work to be done regarding the coordination of a team of robots based on the signal strength of wireless communication. First, the results obtained in simulation need to be verified using actual robots and wireless communication equipment. There are also a number of questions to be further explored in simulation as well. We will determine the effect of τ in environments with obstacles that have high PAF values. Experiments will also be conducted in environments with office building like structures (i.e. an environment with rooms, hallways, doorways, etc...). The algorithm for the adaptive topology will be changed to remove the possibility of the "live lock" situation that we occasionally encountered in the course of the experiments presented here. We also plan to explore signal strength based coordination in the context of other tasks. These tasks could include finding wireless nodes in the environment and maintaining a network between them, or maintaining connectivity between two or more nodes cooperatively addressing some task. Both of these tasks require that additional constraints be placed upon the teams. In these tasks, any of the wireless nodes addressing task goals would act much like the team leader, but since there are two or more of such nodes, control methods will need to be developed to address any conflicting constraints imposed by leader nodes. These control methods may need to position robotic resources in anticipation of the possible motions of the leader nodes to prevent the loss of communication. There may also need to be additional constraints placed on leader nodes to prevent situations where it would be impossible for the other nodes to maintain a connected network. This may require explicit coordination between the multiple leaders.

VII. Acknowledgments

The authors would like to gratefully acknowledge Rod Grupen for his contributions to this paper. Preparation of this manuscript was supported by grants from NSF (#CDA 9703217), NSF (#ANI-0133055), NASA

(DARPA-MARS subcontract #NAG9-1445), DARPA-MARS (#DABT63-99-1-0004), DARPA-MARS (#DABT63-00-1-004), and DARPA-SDR(#DABT63-99-1-0022). Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Aeronautics and Space Administration, the National Science Foundation, or the Defense Advanced Research Projects Agency.

References

[1] W. Burgard, M. Moors, C. Stachniss, and F. Schneider, "Coordinated multi-robot exploration," *IEEE Transactions on Robotics and Automation (to appear).*

[2] J. Sweeney, T. Brunette, Y. Yang, and R. Grupen, "Coordinated teams of reactive mobile platforms," in *Proceedings of the 2002 IEEE Conference on Robotics and Automation*, Washington D. C., 2002, pp. 299–304.

[3] R. C. Arkin and J. Diaz, "Line-of-sight constrained exploration for reactive multiagent robotic teams," in *AMC 7th International Workshop on Advanced Motion Control*, 2002.

[4] R. Simmons, D. Apfelbaum, W. Burgard, D. Fox, M. Moors, S. Thrun, and H. Younes, "Coordination for multi-robot exploration and mapping," in *Proceedings of the National Conference on Artificial Intelligence*, Austin, TX, August 2000.

[5] D. F. Hougen, M. D. Erickson, P. E. Rybski, S. A. Stoeter, M. Gini, and N. Papanikolopoulos, "Autonomous mobile robots and distributed exploratory missions," in *Distributed Autonomous Robotic Systems 4, Proceedings of the 5th International Symposium on Distributed Autonomous Robotic Systems*, L. E. Parker, G. Bekey, and J. Barhen, Eds. Tokyo: Springer, 2000, pp. 221–230.

[6] L. E. Parker, "Current state of the art in distributed autonomous mobile robotics," *Distributed Autonomous Robotic Systems*, vol. 49, pp. 3–12, 2000.

[7] L. Chaimowicz, M. F. M. Campos, and V. Kumar, "Dynamic role assigment for cooperative robots," in *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, Washington D. C., May 2002, pp. 293–298.

[8] N. Miyata, J. Ota, T. Arai, and H. Asama, "Cooperative transport by multiple mobile robots in unknown static environments associated with real-time task assignment," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 769–780, October 2002.

[9] M. Powers and T. Balch, "Value-based communication preservation for mobile robots," in *7th International Symposium on Distributed Autonomous Robotic Systems*, 2004.

[10] S. O. Anderson, R. Simmons, and D. Goldberg, "Maintaining line of sight communications networks between planetary rovers," in *Proceedings of the 2003 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems*, Las Vegas, Nevada, October 2003, pp. 2266 – 2272.

[11] Y. Yang, O. Brock, and R. A. Grupen, "Exploiting redundancy to implement multi-objective behavior," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Taipei, Taiwan, May 2003, pp. 3385 – 3390.

[12] L. Chaimowicz, T. Sugar, V. Kumar, and M. F. M. Campos, "An architecture for tighly coupled multi-robot cooperation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Seoul, Korea, 2001.

[13] S. Carpin and L. E. Parker, "Cooperative leader following in a distributed multi-robot system," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Washington D. C., May 2002, pp. 2994–3001.

[14] A. Wagner and R. Arkin, "Multi-robot communication-sensitive reconnaissance," in *Proceedings of the IEEE International Conference on Intelligent Robotics and Systems*, New Orleans, LA, April 2004, pp. 4674–4681.

[15] T. S. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. Prentice Hall, 2001.

[16] Proxim Corporation, "Orinoco 11b client pc card." [Online]. Available: http://www.proxim.com/learn/library/datasheets/11bpccard.pdf

[17] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction To Algorithms*. Cambridge: The MIT Press, 1990.

[18] C. I. Connolly and R. A. Grupen, "On the application of harmonic functions to robotics," *Journal of Robotic Systems*, vol. 7, no. 10, pp. 931–946, 1993.

[19] C. I. Connolly, "Harmonic functions and collision probabilities," in *Proceedings of the IEEE International Conference on Robotics and Automation*, San Diego, CA, 1994, pp. 3015–3019.

[20] J.-C. Latombe, *Robot Motion Planning*. Boston, MA: Kluwer Academic Publishers, 1991.

[21] J. D. Sweeney, H. Li, R. A. Grupen, and K. Ramamritham, "Scalability and schedulability in large, coordinated, distributed robot systems," in *Proceedings of the 2003 IEEE Confrence on Robotics and Automation*, Taipei, Taiwan, Sept. 2003, pp. 4074–4079.

[22] Y. Nakamura, *Advanced Robotics: Redundancy and Optimization*. Reading, MA: Addison-Wesley, 1991.

[23] M. Huber and R. A. Grupen, "A feedback control structure for on-line learning tasks," *Journal of Robotics and Autonomous Systems*, vol. 22, no. 3-4, pp. 303–315, 1997.

[24] R. Platt, Jr., A. H. Fagg, and R. A. Grupen, "Nullspace composition of control laws for grasping," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2002.

[25] D. D. Jensen and P. R. Cohen, "Multiple comparisons in induction algorithms," *Machine Learning*, vol. 38, no. 3, 2000.