Signal Strength Coordination for Cooperative Mapping

Bryan J. Thibodeau  Andrew H. Fagg 1  Brian N. Levine
Department of Computer Science
University of Massachusetts Amherst
{thibodea,fagg,brian}@cs.umass.edu

Abstract—Many mobile robot tasks can be performed in parallel. Multi-robot teams have the potential to complete the task more quickly than a single robot. Communication and coordination 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. The robot formation will be determined by and coordinated using the signal strength between pairs of robots. 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 robots can avoid duplicating the effort of other robots. An example of this is sharing map information so multiple robots do not explore the same area of an environment [3, 20, 7]. However, robots can only exchange information when they are in communication range of each other.

Maintaining wireless communication among a team of robots moving through an unknown environment is a fundamental problem in multi-agent mobile robotics. 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 which can over-constrain the relative movement of the robots [22, 2]. 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. However wireless communication does not often require a line-of-sight. Line-of-sight methods also require that the distance between communicating robots not exceed some maximum distance. We assume this distance is equal to twice the sensing radius of a robot, as this is the maximum separation for which two communicating robots can verify that there are no obstacles present along the line-of-sight between them. This can 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 and, unlike the sensors, wireless communication can propagate through obstacles.

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. 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. Using signal strength to coordinate teams of robots presents a number of challenges. Not only does signal strength behave stochastically, it also doesn’t directly correspond to the spatial relationship between robots. The lack of a deterministic mapping from signal strength to spatial relationship complicates repairing the network of robots when the signal strength between robots becomes too low.

In simulation, we compare a method that relies on maintaining a line-of-sight between robots to our method which relies on maintaining suitable signal strength between robots. In some situations the average time to explore an environment using signal strength coordination is less than one third 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 [21, 22, 11, 15].

1Now at University of Oklahoma School of Computer Science
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 [3]. Communicating robots are also better able to address tasks requiring explicit coordination such as cooperative transport [6, 14], 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 [22, 2, 17, 1]. 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. 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 [22, 25, 5, 4, 2] 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 [22]. In this paper, we use leader-follower relationships to determine which pairs of robots must maintain wireless links to each other.

The leader-follower style of coordination has traditionally assumed a fixed topology of robots. This assumption can be overly-restrictive: the breaking of a leader-follower relationship does not necessarily mean that the network has become partitioned. Rather than trying to repair a lost leader-follower relationship under conditions in which the network is still connected, we propose allowing a new leader-follower relationship to form. Such a fluid topology approach enables the robots to expend less effort maintaining network connectivity.

Sheng et al. (2004) show that performance in a multi-robot exploration task can be improved by increasing connectivity among members of a robot team [20]. In their work, robots share information whenever they are within a certain radius of each other, but no explicit attempt is made to maintain connectivity between members of the team. Sheng et al. (2004) compare the case when the robots are biased toward exploring areas near other robots (in general improving connectivity) to the case where no such bias exists. They found that the improved connectivity resulting from such a bias decreases the amount of time required to explore an environment.

Wagner and Arkin (2004) propose an approach combining planning and reactive behavior to maintain communication in a team of robots performing a reconnaissance task [24]. 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 (2004) 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 [17]. 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 (2004) demonstrate that it is feasible to use the signal strength of wireless signals to maintain communication in a team of robots, but the potential benefits of non-line-of-sight communications is not considered.

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 in simulation 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 through 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.

In this work, we compare the performance of signal strength coordination to 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

We perform all experiments in a simulated 200m-by-200m square environment. We use sparse and dense environments for experiments. The sparse environment has 20 line-segment obstacles placed uniformly at random throughout the environment. The obstacles have 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. Thus, we must rely on extremely simplified models of signal propagation.

We make the following assumptions 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 per second). We also assume that the robots can accurately measure the signal strength of any signal they receive and can determine with which of their team members they can currently communicate. 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 the robots is less than some parameter \( R \). To simulate path loss in an environment with obstacles, we use the following model from Rappaport (2001) [19]:

\[
\mathcal{P}L(d) = \mathcal{P}L(d_0) + 20\log\left(\frac{d}{d_0}\right) + \alpha d + \sum_i PAF_i; \tag{1}
\]

where:
- \( d \) is the distance between the transmitter and the receiver;
- \( \mathcal{P}L(d_0) \) 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;
- \( \alpha \) is a constant that depends on the type of environment in which the signal is traveling (i.e., office building, warehouse, or outdoors);
- \( PAF_i \) 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. We state the parameters used in our simulation below.

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. 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 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 [19]. 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. We modeled multi-path effects in our simulation by adding noise to the model in Eq. 1 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, a very small amount of 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, we use the following parameters: \( d_0 = 1m, PL(d_0) = 30dB, \) and \( \sigma = 0.35. \) We chose these parameters 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 (the parameters chosen were not empirically verified) [18]. The model generated by these parameters does not match the specification 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 [19].

Figure 3 shows the path loss determined by the above model for a transmitter at the center of a sample environment. Lighter shades represent lower path loss and black grid squares contain obstacles. No noise was added in this case. Figure 4 shows the result of adding noise equal to \( |N|, \) where \( N \) is a Gaussian random variable with \( \mu = 0dB \) and \( \sigma = 4dB, \) to the value given by Eq. 1 for grid squares not within line-of-sight of the center of the environment. Noise is added to the squares that are within line of sight of the center of the environment in a similar manner, except the Gaussian distribution has \( \sigma = 0.5dB. \)

Two robots can communicate when the path loss between them is less than \( R = 81.5dB. \) Even though actual wireless communication hardware (such as 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

We assume 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. The robot team is represented as a graph in which the edges represent path loss. Prim’s algorithm [10] is used to construct a minimum spanning tree of the set of robots. The number of partitions in the network can be determined by counting the number of links in the minimum spanning tree that have a path loss greater than \( R = 81.5dB. \) It is important to note that the minimum spanning tree does not necessarily represent the complete routing topology of the ad hoc network, rather it is used to determine the network connectivity. The robots also use the minimum spanning tree to determine the leader-follower relationships between robots. We provide the details of the minimum spanning tree construction in Section IV.

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 \( 16m \) apart. The robots move at a constant speed of \( 0.25m \) per time step. A simulation time step is equal to 1 second. The robots are localized and always know their current position in the world.

IV. Algorithm

In this section, 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 unnecessarily 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 following properties. It should be the case that if every follower is in direct communication with its leader, then the network of robots is connected. This simplifies the problem of global connectivity by reducing it to one of maintaining pairwise relationships between robots. It is also important that every robot have only one leader. 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, and so on). The presence of a team leader helps ensure that the team will always be making some amount of progress toward task objectives.

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 [10]. 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 that 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 \( R = 81.5 \text{dB} \), 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

We use a set of controllers based on harmonic path planners to implement the above coordination method. 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 [9, 8].

In this work, a robot’s configuration consists of its coordinates in a planar world. For the purposes of computing harmonic functions, we represent configuration space as a discrete grid where every grid square is designated as freespaces, 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 [9].

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 [13] 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.

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 \( \phi_i^g \), where:

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

\( g \) may refer to sensory information at any level of abstraction. We use sensory abstractions at the level of configuration space
maps for specific objectives. Harmonic functions are used to generate the artificial potential $\phi$, or in cases where the local gradient of the harmonic function cannot be determined, $\phi$ is determined using the NFI1 function. The effectors always consist of single robots.

The controllers are similar to those described by Sweeney, et al. (2002,2003) [22, 23]. Robot $r$ uses the controller $\phi_r^{\text{EXP}}$ for exploration. The sensory abstraction $\text{EXP}_r$ 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 senses 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. $\phi_r^{\text{EXP}}$ will generate trajectories that avoid obstacles and move the robot toward unobserved areas of the world.

We use the controller, $\phi_j^{\text{LOS}}$, to bring robot $j$ to a location where it is in line-of-sight of robot $i$ ($i \neq j$). The sensory abstraction $\text{LOS}_i$ 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, $\phi_j^{\text{LOS}}$ 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 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 affect the performance of superior tasks. 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 [12, 16]. The “subject-to” operator [12] is used to combine the actions of disparate controllers. For controllers $\phi_\alpha$ and $\phi_\beta$, $\phi_\beta \triangleleft \phi_\alpha$ (read “$\phi_\beta$ subject-to $\phi_\alpha$”) means that the actions of $\phi_\beta$ are projected onto the equipotential manifold of controller $\phi_\alpha$’s artificial potential. Thus, the actions generated by $\phi_\beta$ do not interact destructively with the progress of $\phi_\alpha$ toward its minimum.

In this work, we 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 $\phi_\alpha$, $\phi_\beta$ is used to generate motion commands. When the system is not in a goal state of $\phi_\alpha$, then $\phi_\alpha$ 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. We use the parameter $\tau$ defined below to determine the bound on disturbances.

Leader-Follower Relationship for Line of Sight Coordination:

Under line-of-sight coordination, robot $f$ uses $\phi_f^{\text{EXP}} \triangleleft \phi_f^{\text{LOS}}$, where robot $l$ is robot $f$’s leader. 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 $\phi_f^{\text{LOS}}$ to move toward the region where it would be within line-of-sight of robot $l$. For the team leader, $\phi_l^{\text{LOS}}$ is undefined (because the team leader has no leader), and $\phi_f^{\text{EXP}}$ is always used for control.

Leader-Follower Relationship for Signal Strength Coordination:

For signal strength coordination, we will need to define one more controller. Let $\phi_f^{\text{SIG}}$ be a controller that is the same as $\phi_f^{\text{LOS}}$, except the goal region generated by $\text{SIG}_l$ is all points in configuration space where the path loss to robot $l$ is less than some threshold $\tau$. $\tau$ is always less than $R$, the maximum path loss at which communication is still possible. For signal strength coordination, robot $f$ uses $\phi_f^{\text{EXP}} \triangleleft \phi_f^{\text{SIG}}$ 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 $\tau$. Otherwise, robot $f$ will move toward the region of space where the path loss to robot $l$ is less than $\tau$.

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 $\tau$. However, it is still possible for robot $f$ to directly sense (by measuring the strength of the signal from robot $l$) whether it is in a goal state of $\phi_f^{\text{SIG}}$. Because we cannot
compute the goal set of $\phi_f^{SIG_1}$, when robot $f$ is not in a goal set of $\phi_f^{SIG_1}$ (i.e., when the path loss between robots $f$ and $l$ is greater than $\tau$), $\phi_f^{LOS_l}$ will be used for control. Note that $\phi_f^{LOS_l}$ is not guaranteed to monotonically decrease the path loss between robots $l$ and $f$. However, as long as $\tau$ 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 $\tau$ for all configurations in the goal set of $\phi_f^{LOS_l}$. Thus, $\phi_f^{LOS_l}$ is a conservative approximation of $\phi_f^{SIG_1}$, 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 whose 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.

Even though we guarantee that network partitions will always be temporary, the progress of the search task can be adversely affected by partitions in the network. Since the group leader stops whenever there is a partition, the area of the environment that is reachable by the group is limited until all partitions have been repaired. Also, since map information is not shared across partitions, the robots that are not connected to the group leader’s partition may not act efficiently since they lack the map knowledge that other robots have discovered while the partition exists.

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 performance 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. We used the same 25 randomly generated instances of each environment for every experiment. The PAF of the obstacles in the environment are varied depending on the experiment. In the following graphs, each bar represents the average of 25 trials, one in each instance of the appropriate environment type. We found that in most environment, the performance of the signal strength coordination method compared favorably with that of the line-of-sight coordination method. Furthermore, the signal strength coordination method does not appear to be extremely sensitive to algorithmic parameters. We also found that a fluid topology was beneficial to both the line-of-sight and signal strength coordination methods.

A. Number of Robots

We first conducted experiments to determine how team size effects the performance of 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.

The average time to fully search the environment using both the line-of-sight and signal strength coordination methods is shown in Figures 6 and 7. For signal strength coordination we set $\tau = 77.6$ dB. The bars labeled “LOS” correspond to line-of-sight coordination, and the bars labeled “SIG” correspond to signal strength coordination. The bars labeled “$R=\infty$” correspond to the case where signal strength coordination is used and communication range is unlimited ($\tau$ 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 773 time steps; for about 295 time steps there were two partitions;
for roughly 81 time steps there were three partitions; and for less than 20 time steps there were four, five, six, or seven 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, but line-of-sight coordination is 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 (even though both methods are subject to the same networking constraints), 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 3.3–4.0 times quicker than teams of two robots. When signal strength coordination is used, teams of 32 robots complete the search task approximately 5.8–6.8 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.

The cases where the communication range is assumed to be infinite (\(R=\infty\)) 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 “\(R=\infty\)” case is statistically insignificant (\(p=0.058\) using a paired t-test) for 32 robots in the sparse environment. For every other case, the difference in performance between signal strength coordination and the “\(R=\infty\)” case is statistically significant. 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.

Another interesting effect that can be seen in Figures 6 and 7 is that the network connectivity seems to worsen as \(n\) increases for \(n \leq 8\), but for \(n \geq 8\), the network connectivity appears to improve. We attribute this effect to the teams of 16 and 32 robots “saturating” the environment. As the environment becomes saturated with robots, we would expect less change in search times as more robots are added since there is not enough room for the robots to spread out and cover independent portions of the environment. But we would expect the connectivity to continue improve as robots are added since a higher spatial density of nodes makes a network partition less likely to occur.

### B. Signal strength threshold

The signal strength coordination method requires choosing a value for the threshold \(\tau\) that determines when robots will switch from the \(\phi_i^{\text{EXP}}\) controller to the \(\phi_i^{\text{LOS}}\) controller. In order to evaluate the signal strength coordination method we need to find both the optimum value for \(\tau\) and how sensitive task performance and network connectivity are to different choices of \(\tau\). The optimum value for \(\tau\) in a given situation demonstrates the potential 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 \(\tau\) as environmental characteristics change. If the sensitivity is too high then it may be difficult for a system designer to pick an appropriate value for \(\tau\), especially if the characteristics of the environment are unknown, which would reduce the utility of the method.

Experiments were performed in both the sparse and dense environments, varying the value of \(\tau\). Since communication is broken when the path loss exceeds 81.5\,dB, we only consider values of \(\tau\) less than or equal to 81.5\,dB.

Figures 8 and 9 show the search times in the sparse and dense environments, respectively. Figure 10 shows the search
times in the dense environment with a $P_{AF}$ of 25dB. Each line in these graphs corresponds to a particular team size, and shows the change in search performance as $\tau$ increases. The leftmost point of each line shows the performance of the line-of-sight method for the corresponding number of robots.

In the environments tested, $\tau$ can be set around 77dB with essentially no adverse effect on task performance. This demonstrates that signal strength coordination does not require extensive parameter tuning in order to outperform the line-of-sight method in the three environments tested.

These experiments also demonstrate how the value of $\tau$ effects connectivity in the network of robots. Figures 11, 12, and 13 display the time to map the environment for each team size and value of $\tau$, broken down by the number of partitions in the network. We can see from Figure 8 that $\tau$ has less of an effect on network connectivity in the sparse environment. The network is partitioned only slightly more often as $\tau$ increases, which is expected since a higher value of $\tau$ allows the robots to disperse more. Figures 12 and 13 show that in the dense environment (with $P_{AF} = 5$ or 25dB) the value of $\tau$ has a much greater effect on the network connectivity. This suggests that in denser environments, or environments where the obstacles have a higher $P_{AF}$, network connectivity may be more sensitive to the value of $\tau$ than task performance. Thus, in some cases, $\tau$ may need to be chosen more carefully if continuous network connectivity is a high priority.

### C. Obstacle Composition

The partition absorption factor, or $P_{AF}$, 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 mostly considered obstacles with a $P_{AF}$ of 5 or 25dB. In real environments, robot teams are likely to encounter obstacles with a large range of $P_{AF}$s, since the $P_{AF}$ of an obstacle is dependent upon the obstacle’s composition and thickness.

For this set of experiments, the mapping task was performed in the dense environment with a team of 8 robots with the $P_{AF}$ of every obstacle set to 5, 15, 25, 35, or $\infty$dB. This range of $P_{AF}$s covers many different materials, including those that are opaque to wireless signals. As stated before, a $P_{AF}$ of 5dB might be expected from obstacles such as empty cardboard boxes or storage racks, whereas $P_{AF}$s from 20-25dB might be
expected from concrete block walls or metal obstacles [19]. The results of these experiments are shown in Figure 14.

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.

Surprisingly, the performance of the signal strength based coordination method degrades very little as the PAF increases. The performance appears to be essentially constant for PAFs of 15dB and above since there is no statistically significant difference between any of cases where $\tau \geq 15$ ($p \geq 0.137$). This means that even when obstacles are opaque, the signal strength coordination method outperforms the line-of-sight method. This is due to the larger distances between team members allowed by the signal strength method. It is important to note that these results were achieved by using the same switching threshold for the signal strength method, $\tau = 77.6$dB. This suggests that careful optimization of $\tau$ is not necessary for the signal strength method to outperform the line-of-sight method for a wide range of obstacle types.

The drawback of the signal strength method is that it allows for many more temporary partitions to occur in the network of robots. For the task considered here, these intermittent breaks do not significantly harm the task performance (or at least the harm caused by the intermittent breaks is outweighed by the benefits of a bigger coverage area).
D. Significance of fluid topologies

To determine the effectiveness of the flexible signal strength based team topology, experiments are performed where the leader-follower relationship between robots is held fixed. The results of these experiments are compared to similar experiments where the fluid topology is used. Fixed leader-follower relationships mean that robots do not change their leaders and will always try to maintain a path loss less than $\tau$ 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 examined these fixed topologies in the sparse and dense environments, using both signal strength coordination (with $\tau = 77.6$ dB) and line-of-sight coordination. The results are shown in Figures 15-18.

For a team size of two, the performance of all of the topologies is very similar, 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. The star topology is also very brittle since all of the other robots tend to form a tight cluster behind the group leader. This makes it possible for all of the follower robots to become partitioned from the group leader simultaneously.

One possible reason that the fixed tree topology performs worse than the chain or minimum spanning tree topologies is that the maximum separation between any two robots (not necessarily in a direct leader-follower relationship) is much
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 robots 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. Teams using the adaptive minimum spanning tree topology often take a chain-like structure, especially when line-of-sight coordination is used. This explains why the two topologies are so close in terms of performance. In some situations, the chain topology can also 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. Conclusion

In this paper we introduced a network topology that adapts based on the signal strength between pairs of robots. 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.

Our experiments have demonstrated, in simulation, the effectiveness of our signal strength coordination method. Compared to line-of-sight coordination, signal strength coordination usually completes mapping tasks much quicker for teams ranging in size from 2 to 32 robots, while the line-of-sight coordination method completes the task less efficiently but results in less temporary partitions of the network of robots. Our experiments suggest that signal strength coordination does not require extensive tuning in order to outperform the line-of-sight method in a variety of situations, which is important if teams using the signal strength coordination method are to be deployed in unknown environments. In the future we hope to validate these experiments with actual robots and wireless communication equipment.

We believe that our method could be tailored to other tasks requiring significant coordination between robots, such as tracking multiple targets. Such tasks will require different parameterizations of our algorithm depending on the importance of preventing network partitions relative to pursuing task goals. There are other tasks that may be able to be performed most effectively using signal strength coordination, but which require extensions to the algorithm presented here. Consider the task of maintaining a network connection between two or more uncontrolled agents. The uncontrolled agents could be humans, or other robots performing some task which requires them to communicate with each other. The controlled agents can be viewed as mobile repeaters tasked with keeping a network connection among the humans and/or robots carrying out the team’s primary task. Even though line-of-sight coordination is effective for preventing network partitions, the team may be
very restricted spatially if there is significant clutter in the environment. This suggests that signal strength coordination may provide better performance, especially when the uncontrolled nodes wish to move to spatially distant locations. Such an extension could also impose additional topological constraints on the team, requiring a modification of our algorithm for determining the team’s topology. Such an extension would also allow for additional topological constraints on the team, such as requiring every robot be connected to at least two other robots. This type of constraint could improve connectivity in a team that uses signal strength coordination, while allowing the team to be more spatially distributed in certain types of environments than a team that uses line-of-sight coordination.

VII. ACKNOWLEDGMENTS

The authors would like to thank 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


[21] Reid Simmons, David Apfelbaum, Wolfram Burgard,


