

Multi-Robot Coordination with Periodic Connectivity

Geoffrey Hollinger and Sanjiv Singh

Abstract—We consider the problem of multi-robot coordination subject to constraints on the configuration. Specifically, we examine the case in which a mobile network of robots must search, survey, or cover an environment while remaining connected. While many algorithms utilize continual connectivity for such tasks, we relax this requirement and introduce the idea of *periodic connectivity*, where the network must regain connectivity at a fixed interval. We show that, in some cases, this problem reduces to the well-studied NP-hard multi-robot informative path planning (MIPP) problem, and we propose an online algorithm that scales linearly in the number of robots and allows for arbitrary periodic connectivity constraints. We prove theoretical performance guarantees and validate our approach in the coordinated search domain in simulation and in real-world experiments. Our proposed algorithm significantly outperforms a gradient method that requires continual connectivity and performs competitively with a market-based approach, but at a fraction of the computational cost.

I. INTRODUCTION

A mobile network of robots must plan paths through an environment to gain information, or even simply to get from one place to another. Instances of this general problem include searching for a mobile target [1], estimating climate change [2], mapping an environment while remaining localized [3], and exploring multiple locations [4]. In some applications, the network may be able to communicate in any configuration. However, in many cases, constraints on the network prevent communication. For instance, obstacles or distance may prevent two robots from communicating.

Prior work in multi-robot coordination often enforces *continual* connectivity constraints. These constraints may be soft (breakable at a penalty) or hard (not breakable at all), but in either case, the goal is to maintain connectivity to all nodes in the network at all times. We introduce the idea of *periodic* connectivity. In many scenarios, it may be desirable to break connectivity if the network plans periodically to become connected in the future. Periodically regaining connectivity allows the network to (1) communicate information gathered, (2) coordinate the next phase of the plan, and/or (3) check that all robots are still operational. We limit the scope of this paper to information gathering tasks that fit into the informative path planning framework, but periodic constraints could potentially apply to any multi-robot domain.

We demonstrate that receding horizon planning with implicit coordination can incorporate periodic constraints, leading to significant improvements over algorithms that require continual connectivity. The novelties of this paper include: the formalization of multi-robot informative path planning

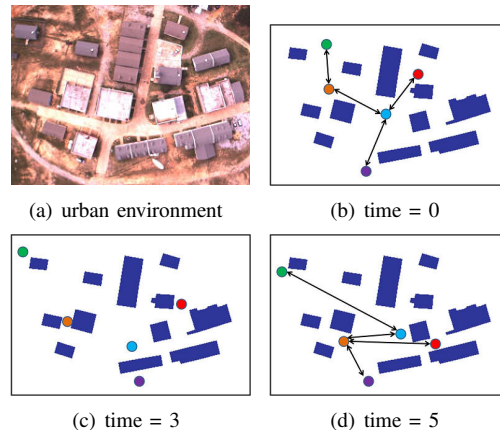


Fig. 1. Example of searching a simulated McKenna MOUT site (a) while maintaining periodic connectivity. The buildings prevent line-of-sight communication links (shown as arrows) between robots in the network (shown as circles). The robots start in a connected configuration (b); the network becomes completely disconnected and explores a small part of the environment (c); the network regains connectivity in a new configuration to replan and share information (d).

with periodic connectivity (MIPP-PC), a reduction of MIPP-PC to MIPP for the case of given connected configurations, and the introduction of a scalable, online algorithm for approximately solving MIPP-PC for unspecified connected configurations without the requirement of a centralized controller.

II. RELATED WORK

Multi-robot informative path planning is a challenging NP-hard problem [2]. The general informative path planning problem can be formulated as a Partially Observable Markov Decision Process (POMDP). POMDPs are notoriously difficult to solve, though near-optimal solvers are possible for POMDPs with several thousands of states [5]. However, the number of states in a multi-robot POMDP scales exponentially in the number of robots, which quickly leads to intractability even with state-of-the-art solvers.

Due to the intractability of solving multi-robot POMDPs, several approximation schemes have been developed for informative path planning. Singh et al. proposed a branch and bound strategy with sequential allocation that yields bounded performance in general informative path planning domains [2]. In our prior work, we showed that non-adversarial search fits into the informative path planning framework, and we proposed a receding horizon method that performs competitively with general POMDP solvers in this domain [1].

G. Hollinger and S. Singh are with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA {ghollin, ssingh}@ri.cmu.edu

There has been significant research in distributed control of mobile networks to maintain *continual* connectivity for path planning tasks. In some cases, continual connectivity is considered *hard* or unbreakable. Zavlanos and Pappas proposed a gradient-based optimization along with a market-based link deletion protocol to move a team of robots to a goal while maintaining hard connectivity constraints [6]. This work utilizes the gradient of the second smallest eigenvalue (Fiedler value) of the Laplacian to determine the connectivity of the network. A similar gradient-based algorithm was employed by Hsieh et al., which allowed for building the communication graph online [7].

Market-based techniques have been utilized to provide long-term planning for multi-robot path planning with continual connectivity. Kalra proposed the Hoplites framework that allows robots actively to coordinate by sharing plans [4]. Unlike gradient-based methods, long-term planning market-based methods like Hoplites can be applied to domains with periodic connectivity.

In some domains, the network connectivity constraints are considered to be *soft*, or breakable. Krause et al. examined the problem of static network placement with varying connectivity throughout the network [8]. Their proposed algorithm has theoretical guarantees and performs well in static networks, but there is no straightforward extension to mobile networks or periodic connectivity. Anisi et al. introduced the idea of recurrent connectivity, where robots regain connectivity only when they make observations [9]. They apply their approach to a coverage problem, but they do not generalize to broader problems.

Related work in multi-robot coordination and maintaining continual connectivity do not allow for scalable long-term planning for information gathering tasks with periodic constraints. Our work fills this gap.

III. PROBLEM SETUP

We are given a team of K robots with the task of planning paths through an environment that maximize some measure of informativeness. We assume that the environment is known and has been discretized into a graph $G(N, E)$ in which the nodes represent convex regions, and the edges represent connections between these regions.

We consider the planning space specified by the graph $G'(N', E')$, which is a time-unfolded version of the graph G . Each node in N' is a time-stamped copy of a node in N , i.e. each node $n' \in N'$ corresponds to (n, t) for a node $n \in N$ and time t . Each robot k plans a path $P_k = \{r_k(0), \dots, r_k(T)\}$ for discrete time steps $t = 0, \dots, T$, where $r_k(t) \in N'$. Let $C(P_k)$ be the cost of the path for robot k . We assume that all moves have cost one, so $C(P_k) = T$ for a horizon T ; however, an extension to more complex cost functions is fairly straightforward. The union of paths for all robots represents which nodes are visited at which time and is given by $P = P_1 \cup \dots \cup P_K$. Planning in this space allows modeling non-stationary reward functions as well as revisiting nodes several times in the path. For this paper, we assume that robots can occupy the same cell, and

that low-level collision avoidance is available. Alternatively, collision avoidance could be incorporated by disallowing robots from occupying the same cell at the same time.

The robots' goal is to plan paths that maximize an objective function $F(P)$. We assume that F is a known, deterministic function of the possible paths. Possible objective functions include discounted probability of capture of a target [1], mutual information in a Gaussian Process [2], and information gain in a tracking scenario. The MIPP optimization problem is given below:

$$\max_{P_k \subseteq N'} F(\cup_{k=1}^K P_k); \text{ s.t. } C(P_k) \leq T, \forall k \in \{1, \dots, K\} \quad (1)$$

In addition, we are given a graph $G_C(N, E_C)$ that describes the connectivity between the nodes in G (i.e., node u is connected to node v iff there exists an edge $uv \in E_C$). If the graph G_C is not given, but a more general form of connectivity constraints are available (e.g., range constraints or line-of-sight), G_C can be computed during pre-processing. We make no restrictions on the edges in graph G_C , which allows for any type of connectivity constraints. However, we do assume that connectivity is either "on" or "off." A configuration $P(t)$ is considered connected at time t if there exists a path in G_C from $r_i(t)$ to $r_j(t)$ for all i, j in which all nodes in the path are occupied.

For the MIPP-PC problem, the robots begin in a connected configuration $P(0)$ and must regain a connected configuration at times $P(T_I), P(2T_I), \dots, P(\tau T_I)$, where τ is the smallest integer such that $\tau T_I > T$. The variable T_I is the *disconnected interval* during which the network may choose to lose connectivity. In many cases, T_I will be dictated by the application, and in other cases it can be selected by the user. Note that when $T_I = 1$, the problem is equivalent to planning with continual constraints.

Given a fully connected graph G_C , the MIPP-PC problem is equivalent to the MIPP problem, and since MIPP is NP-hard [2], MIPP-PC is NP-hard. Thus, we cannot expect to solve MIPP-PC optimally with an efficient algorithm (unless $P = NP$).

IV. ALGORITHM DESIGN

We now design algorithms for solving the MIPP-PC problem approximately. We require that our algorithms are scalable to large teams, run online, and do not use a centralized controller.

A. Pre-Specified Connected Configurations

Here we assume that the configurations $P(T_I), P(2T_I), \dots, P(\tau T_I)$ are given either by an oracle or as part of the problem. For instance, robots may need to plan paths between pre-specified checkpoints where the checkpoints provide connected configurations. In this case, the MIPP-PC problem reduces to several instances of the unconstrained informative path planning problem:

Theorem 1: Given the connected configurations $P(0), P(T_I), P(2T_I), \dots, P(\tau T_I)$, multi-robot informative path planning with periodic constraints (MIPP-PC) reduces

to τ instances of unconstrained multi-robot informative path planning (MIPP).

Proof: The robots start in an initial configuration $P(0)$, and they must attain a configuration $P(T_I)$ at time T_I . The robots must maximize a function $F(\cup_{t=0}^{T_I} P(t))$ given a budget constraint on their paths $B = T_I$ and path constraints on the graph G' . This is equivalent to the MIPP problem with known starting and goal configurations and a path-constrained budget. The robots must solve τ instances of the MIPP problem to fill in the full path $P(0), \dots, P(\tau T_I)$. ■

Based on the reduction in Theorem 1, we can use algorithms from the MIPP domain to yield good approximations in the MIPP-PC domain with pre-specified connected configurations. For cases where a long lookahead is required and significant computation is available, the branch-and-bound algorithm from Singh et al. [2] can be applied. For cases where an online approach is desired, the FHPE+SA algorithm from Hollinger and Singh [1] can be utilized.

B. Unspecified Connected Configurations

In many informative path planning scenarios, the connected configurations will not be specified beforehand. We now design a scalable algorithm for estimating good connected configurations as well as informative paths. Our algorithm utilizes receding-horizon planning and implicit coordination to remain tractable in complex environments and with large teams.

Algorithm 1 gives a description of our method for determining informative paths with unknown connected configurations. This algorithm is called at the start and then each time the robots reach a connected configuration as specified in the previous plan. Thus, it runs online and can account for new information added at each connected configuration. The computation can be distributed among the robots, but it requires synchronization of the updated objective function and maintenance by all robots of a set ($V = \cup_{k=1}^K P_k$), which represents the areas that will be visited in the current plans. Since the network is guaranteed to be connected every T_I , the synchronization and replanning can occur at this time.

Our algorithm can be seen as a coordinate ascent in the space of robot paths. The first robot plans assuming that all robots will remain stationary, and it ensures that its goal remains connected to the starting configuration. The first robot then shares its plan with the other robots, and they can now plan their paths such that they regain connectivity with the new configuration. We refer to this as *implicit coordination* because the robots plan their own paths, and they share these paths to improve performance. Planning order is selected randomly in this paper by assuming that knowledge of which robots have planned is propagated through the network. More sophisticated ordering strategies could also be employed, including the use of neighbors in the network to determine the order. The coordinate ascent continues until convergence. Since the amount of reward collectable is finite, the number of possible paths is finite, and

Algorithm 1 Implicit coordination for MIPP-PC

```

1: Input: robots  $K$ , graph  $G$ , connectivity graph  $G_C$ , ob-
   jective  $F$ , interval  $T_I$ , initial configuration  $P(0)$ 
2: % Initialize all paths to remain stationary
3:  $P_k(1, \dots, T_I) \leftarrow P_k(0)$ , for all  $k$ 
4: %  $V \subseteq N'$  is the set of visited time-stamped nodes
5:  $V \leftarrow \cup_{k=1}^K P_k$ 
6: while not converged do
7:   for all robots  $k \in \{1, \dots, K\}$  do
8:     % Reset path for robot  $k$ 
9:      $V \leftarrow V / P_k$ 
10:    Enumerate some feasible paths  $A_k(1, \dots, T_I)$ 
11:    Discard paths disconnected at  $T_I$ 
12:     $A_k^*(1, \dots, T_I) \leftarrow \operatorname{argmax}_{A_k} F(V \cup A_k)$ 
13:    % Update path for robot  $k$ 
14:     $P_k \leftarrow A_k^*$ ,  $V \leftarrow V \cup A_k^*$ 
15:   end for
16: end while
17: Return  $P(0), P(1), \dots, P(T_I)$ 

```

the reward collected by the planned paths is monotonically increasing, our algorithm will always converge.

Algorithm 1 is similar to the FHPE+SA algorithm proposed in prior work for the unconstrained MIPP domain [1]. However, there are several key differences that allow its extension to the MIPP-PC domain. First, Algorithm 1 limits the planned paths to those that regain connectivity at T_I , which reduces the planning space and ensures periodic connectivity. In addition, FHPE+SA replans at every iteration, which is not feasible in the MIPP-PC domain because of connectivity breaks. Furthermore, FHPE+SA runs only a single iteration of planning instead of continuing to convergence. In the MIPP-PC domain, increasing the number of planning iterations allows for a piggyback effect where the planned paths of the robots eventually converge to a point far away from the start (see Figure 2 for an example). Note that Algorithm 1 assumes that planning to the next connected configuration is completed before execution starts. This restriction is not strictly necessary, and robots could continue planning during execution.

C. Running Time

Each iteration of the outer loop in Algorithm 1 requires K optimizations of a single path MIPP-PC instance. If we enumerate all paths to depth T_I , the computation is $O(Kb^{T_I})$, where b is the branching factor of the search graph. If T_I is too great to enumerate all paths, a stochastic sampling of paths can be employed. In practice, only a few iterations are required for convergence, and in many cases good performance is achievable with only a single iteration.

D. Performance Guarantees

For this section, we make the assumption that the underlying objective function F is nondecreasing and submodular on the ground set of time-stamped locations in the graph G' . We also assume that $F(\emptyset) = 0$.

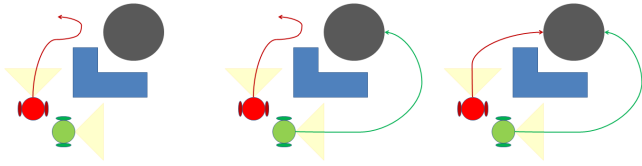


Fig. 2. Example of implicit coordination with periodic connectivity. The robots (green and red) must move around the obstacle (blue L-shape) to observe the area of high information gain (gray circle). They start in line-of-sight contact, and they must regain line-of-sight past the obstacle. The red robot first plans a path that remains connected to the green robot's initial position (left). Then the green robot plans a path that regains connectivity with the red robot past the obstacle (middle). Finally, the red robot replans to regain connectivity with the green robot's new path (right).

Definition 1: A function $F : \mathbf{P}(N') \rightarrow \mathfrak{R}_0^+$ is called *nondecreasing* iff for all $A, B \in \mathbf{P}(N')$, we have

$$A \subseteq B \Rightarrow F(A) \leq F(B).$$

Definition 2: A function $F : \mathbf{P}(N') \rightarrow \mathfrak{R}_0^+$ is called *submodular* iff for all $A, B \in \mathbf{P}(N')$ and all *singletons* $C = \{(n, t)\} \in \mathbf{P}(N')$, we have

$$A \subseteq B \Rightarrow F(A \cup C) - F(A) \geq F(B \cup C) - F(B).$$

Submodularity is the formalization of an intuitive diminishing returns quality (i.e., the more locations visited, the less incremental benefit from visiting new locations). Submodularity holds for many interesting reward functions including information gain and discounted reward in most cases (see Section III for examples).

Let $A_k(0, \dots, \tau T_I)$ be the path returned for robot k by the sequential application of a single-robot MIPP approximation algorithm with an approximation guarantee of κ to some end time τT_I (i.e., τ instances of periodic connectivity with disconnected interval T_I). For instance, exhaustive enumeration of paths yields the optimal single-robot path, and achieves $\kappa = 1$. Let $O_k(0, \dots, \tau T_I)$ be the optimal path. For the case of known connected configurations, we have the following performance guarantee:

Theorem 2: Sequential allocation on the K -robot submodular, nondecreasing MIPP-PC problem with pre-specified connected configurations achieves a lower bound of:

$$F(\cup_{k=1}^K A_k(0, \dots, \tau T_I)) \geq \frac{F(\cup_{k=1}^K O_k(0, \dots, \tau T_I))}{1 + \kappa} \quad (2)$$

Proof:

Singh et al. showed that any κ approximation of a single-robot instance of the MIPP problem on a submodular, nondecreasing function can be extended to a $\kappa+1$ guarantee for the multi-robot MIPP problem using sequential allocation [2]. Applying Theorem 1, we have τ instances of MIPP in which the reward and bound is additive. ■

Let ρ be the amount of reward outside the horizon $T_I/2$. For example, if the cost function F is discounted by γ , and R is the total possible reward, $\rho = R\gamma^{T_I/2+1}$. The Theorem below shows bounds in the case of unknown connected configurations:

Theorem 3: Algorithm 1 on the K -robot submodular, nondecreasing MIPP-PC problem with unspecified connected

configurations achieves a lower bound of:

$$F(\cup_{k=1}^K A_k(0, \dots, \tau T_I)) \geq \frac{F(\cup_{k=1}^K O_k(0, \dots, \tau T_I)) - \rho}{1 + \kappa} \quad (3)$$

Proof:

By construction, the robots always start in a connected configuration $P(0)$. Regardless of connectivity constraints, the robots can always return to this configuration. This gives $T_I/2$ steps to perform unconstrained sequential allocation. By applying Theorem 2, the bound is immediate. ■

The bound in Theorem 3 depends heavily on the constant offset imposed by ρ , which is determined by T_I . For long disconnected intervals (as $T_I \rightarrow \infty$), the guarantee approaches the bound in Theorem 2. For short disconnected intervals, the guarantee degrades fairly quickly. The quality of the solution also relates closely to the accuracy with which the connected configurations are determined. If they are chosen optimally, then Algorithm 1 achieves the bound in Theorem 2. However, if they are chosen poorly, the bound degrades to that in Theorem 3. Since Algorithm 1 performs a sequential coordinate ascent, it can be expected to perform well in cases where the constraints and cost function can be smoothly interpolated (i.e., they have a contour to follow towards the optimal).

V. EXPERIMENTAL RESULTS

We tested our algorithm in the non-adversarial search domain in which a team of robots must find a moving target with a known (or approximately known) motion model. As described above, we are given K searchers whose paths on the time-unfolded graph G' are denoted as $P = P_1 \cup \dots \cup P_K$. The searchers receive reward by moving onto the same node at the same time as the target. This reward is discounted by the time at which this occurs. Given that a target takes path Y , the searchers receive reward $F_Y(P) = \gamma^{t_P}$, where $t_P = \min \{t : (n, t) \in P \cap Y\}$ (i.e., the first time at which Y intersects P), with the understanding that $\gamma \in (0, 1)$, $\min \emptyset = \infty$, and $\gamma^\infty = 0$. Thus, if the paths do not intersect, the searchers receive zero reward.

We assume that the target's behavior is non-adversarial, allowing us to utilize a target motion model independent of the locations of the searchers. This yields a probability $P(Y)$ for all possible target paths $Y \in \Psi$. We can now define the optimization problem in Equation 4. For this paper, we assume that the target is moving according to a random walk, though any Markov motion model could be calculated efficiently within this framework.

$$F(P) = \sum_{Y \in \Psi} P(Y) F_Y(P) \quad (4)$$

To map the formulation above into the MIPP-PC domain, we set the cost function F to Equation 4 (note that discounted probability of capture is both submodular and nondecreasing [1]), with $\gamma = 0.95$, and the disconnected interval $T_I = 5$. Thus, the robots must maximize the discounted probability of capture as well as achieve a connected network

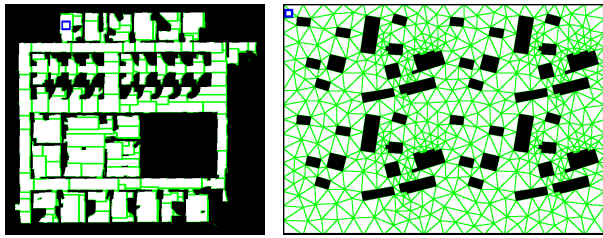


Fig. 3. Environments used for search trials with periodic connectivity. The SDR office (left) contains 188 cells and was discretized using a region growing technique. The McKenna MOUT site (right) contains 937 cells, and was discretized using a constrained Delaunay triangulation. Cell boundaries are shown as light green lines, and starting cells are denoted with a blue square.

every five timesteps. We examine the case for both distance and line-of-sight constraints.

A. Simulated Experiments

We tested our MIPP-PC algorithm in the environments shown in Figure 3. The first environment is a map of the SDR building from the Radish data set [10] discretized into 188 cells. The cells were found using a region growing approach that continues to expand cells until a large enough portion of the cell is occluded by obstacles or other cells. On this map, the robots are required to maintain periodic range constraints. The range constraints were set as $1/4$ the diagonal of the map. The second environment is a repeated version of the McKenna MOUT site tessellated into 937 cells using a constrained Delaunay triangulation. Here, the robots must maintain periodic line-of-sight constraints. The buildings serve as obstacles to obstruct line-of-sight. The map discretization includes only free space; the obstacles are not valid nodes on the map. Obstacle avoidance is handled trivially by moving between the centroids of the convex cells. It is assumed that multiple robots can exist in the same cell at the same time by taking advantage of low-level collision avoidance.

For comparison, we introduce several algorithms similar to those proposed in prior work:

- 1) Coupled gradient - enumerate a single step forward in the joint planning space and discard all paths that are disconnected.
- 2) Gradient+Fiedler - compute the coupled gradient augmented with a connectivity term represented by the second eigenvalue of the Laplacian matrix [6]. The two gradient approaches do not allow periodic connectivity.
- 3) Market coordination - compute a single-robot path and then a two-robot path for each team member in the joint planning space (allowing for periodic connectivity) and auction these plans to other robots [4].
- 4) Implicit coordination - the proposed approach (Algorithm 1) allowing for periodic connectivity.
- 5) Unconstrained baseline - the FHPE+SA algorithm [1] assuming full connectivity. The connectivity constraints are removed to generate a baseline for comparison.

Figure 4 shows simulated results using up to ten robots for the methods described above. The metric used is average-capture steps, where the robots are assumed to move one cell in a step. This intuitive metric relates to discounted capture probability. The disconnected interval remains fixed throughout these experiments. If the interval were reduced to close to one, the problem would resemble continual connectivity, which would likely result in smaller gains from the proposed algorithm over one-step approaches. If the disconnected interval were increased, exhaustive path enumeration may not be possible. However, the proposed algorithm could still be utilized using a stochastic sampling technique, which would maintain the benefits of periodic connectivity.

The coupled gradient performs poorly in SDR as the number of searchers increases, which is due to the algorithm falling into a local maximum that it cannot escape. Incorporating the Fiedler value removes this local maximum but sacrifices performance because the Fiedler heuristic does not directly relate to the objective function. On the MOUT map, both gradient-based approaches perform poorly because they cannot reason about periodic connectivity. On both maps, the market-based approach performs somewhat better than implicit coordination (on average 10% better in SDR and 5% better in MOUT), but requires over fifteen times more computation with ten robots. In addition, the gap between the MIPP baseline and implicit coordination with periodic connectivity is smaller in SDR (17% on average) than in MOUT (35% on average) due to periodic constraints preventing the team from spreading out in the larger environment. The gap shrinks as the team size grows, demonstrating the ability to spread into a chain topology.

Figure 4 shows the computation time to generate a 50-step plan for the various algorithms. The gradient approaches act in the coupled space; they start very fast but scale exponentially in the number of searchers. The market-based approach requires $O(K^2 b^{2d})$ computation to generate a combinatoric number of two-robot joint plans. Our proposed implicit coordination approach requires very few iterations to converge and is approximately linear in the number of searchers.

B. Mobile Robot Experiments

We implemented our MIPP-PC algorithm on a team of two mobile robots: the Serf and Sideswipe autonomous platforms (see Figure 5). Each robot plans its own path and maintains its own localization estimate. The inter-robot communications were handled using the Player software [11], and the robots communicate on a wireless network. We utilized ultra-wideband ranging radios for localization in an outdoor urban environment. The environment was automatically discretized into 52 cells using a constrained Delaunay triangulation. Several obstacles were present, and they were assumed to impede line-of-sight connectivity (see Figure 5).

The robots' task was to search for a stationary intruder using a laser scanner while maintaining periodic connectivity. This task is equivalent to the non-adversarial search problem

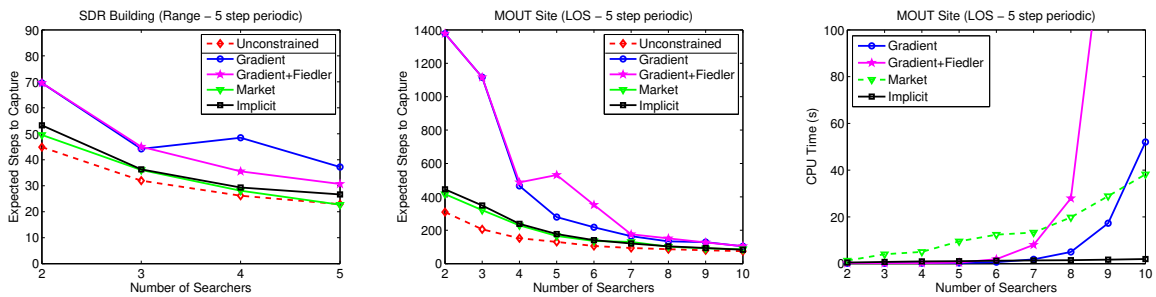


Fig. 4. Simulated comparison of various methods for multi-robot search with periodic connectivity. Our proposed implicit coordination method outperforms gradient-based methods in both the SDR (left) and MOUT (middle) environments. Market-based coordination provides slightly better capture times than implicit coordination (left and middle) but requires significantly more computation to generate a 50-step plan (right). Expected time to capture is calculated by assuming the target moves with a random walk.

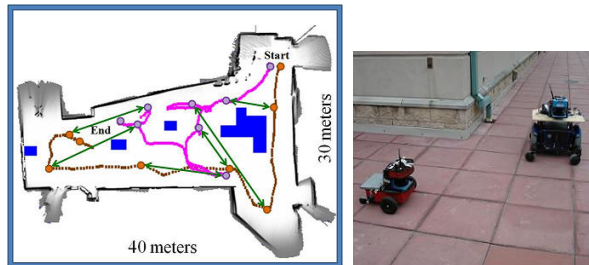


Fig. 5. Left: The paths of two robots (magenta and brown) performing a coverage task in an outdoor urban environment. The blue obstacles prevent line-of-sight connectivity. The robots regain connectivity approximately every 30 seconds at the locations connected with green arrows. Right: Two robots round a corner to regain line-of-sight connectivity while searching.

described above, but assuming a stationary target. The robots calculated a plan using Algorithm 1, and they successfully searched the entire area in less than three minutes while regaining connectivity at most every 30 seconds. Figure 5 shows the paths from this experiment, and the accompanying video shows a playback of the laser and position data. The one-step coupled gradient was run for comparison, and it was unable to find a coverage schedule due to becoming trapped in a local maximum in which neither robot could make progress without breaking connectivity. Utilizing the Fiedler value did not help here since the network contained two robots.

VI. CONCLUSIONS AND FUTURE WORK

We have introduced the problem of multi-robot informative path planning with periodic connectivity (MIPP-PC), and have presented a scalable, online algorithm for solving the problem approximately without a centralized controller. We have shown in the non-adversarial search domain that receding horizon planning with implicit coordination yields significant improvements over gradient approaches. Augmenting the one-step gradient with a Fiedler value heuristic eliminates some local maxima, but still does not allow for plans that temporarily lose connectivity. In addition, we have demonstrated that market-based approaches that explicitly coordinate only perform marginally better than implicit coordination even given significantly more computational time. Thus, similar to many other MIPP problems, planning in the

coupled path space is not necessary to find good strategies with periodic connectivity.

We are interested in extending our techniques to partially known and dynamic environments. In the case of dynamic environments, it may be necessary to develop contingency plans if the network does not regain connectivity due to changes in the environment. In addition, we are interested in extending theoretical guarantees into domains with short disconnected intervals.

VII. ACKNOWLEDGMENTS

The authors thank Athanasios Kehagias, Joseph Djgash, Pragnu Goyal, Benjamin Grocholsky, and Geoff Gordon for their insightful comments and assistance with experiments. The SDR map was obtained from the Robotics Data Set Repository (RADISH) [10]. Thanks to Andrew Howard for providing this data.

REFERENCES

- [1] G. Hollinger, S. Singh, J. Djgash, and A. Kehagias, "Efficient multi-robot search for a moving target," *Int. J. Robotics Research*, vol. 28, no. 2, pp. 201–219, 2009.
- [2] A. Singh, A. Krause, C. Guestrin, and W. Kaiser, "Efficient informative sensing using multiple robots," *J. Artificial Intelligence Research*, vol. 34, pp. 707–755, 2009.
- [3] R. Sim and N. Roy, "Global a-optimal robot exploration in SLAM," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2005.
- [4] N. Kalra, "A market-based framework for tightly-coupled planned coordination in multirobot teams," Ph.D. dissertation, Robotics Institute, Carnegie Mellon Univ., 2006.
- [5] S. Ong, S. Png, D. Hsu, and W. Lee, "POMDPs for robotic tasks with mixed observability," in *Proc. Robotics: Science and Systems Conf.*, 2009.
- [6] M. Zavlanos and G. Pappas, "Distributed connectivity control of mobile networks," *IEEE Trans. Robotics*, vol. 24, no. 6, pp. 1416–1428, 2008.
- [7] M. A. Hsieh, A. Cowley, V. Kumar, and C. Taylor, "Maintaining network connectivity and performance in robot teams," *J. Field Robotics*, vol. 26, no. 1–2, pp. 111–131, 2008.
- [8] A. Krause, C. Guestrin, A. Gupta, and J. Kleinberg, "Near-optimal sensor placements: Maximizing information while minimizing communication cost," in *Proc. Information Processing in Sensor Networks*, 2006.
- [9] D. Anisi, P. Ögren, and X. Hu, "Cooperative surveillance missions with multiple UGVs," in *IEEE Conf. Decision and Control*, 2008.
- [10] A. Howard and N. Roy, "The robotics data set repository (Radish)," 2003. [Online]. Available: <http://radish.sourceforge.net/>
- [11] B. Gerkey, R. Vaughan, and A. Howard, "The player/stage project: Tools for multi-robot and distributed sensor systems," in *Proc. Int. Conf. Advanced Robotics*, 2003, pp. 317–323.