

Leading Multi-Agent Teams to Multiple Goals While Maintaining Communication

Brian Reily*, Christopher Reardon†, and Hao Zhang*

*Colorado School of Mines, †U.S. Army Research Laboratory

breily@mines.edu, christopher.m.reardon3.civ@mail.mil, hzhang@mines.edu

Abstract—Effective multi-agent teaming requires knowledgeable robots to have the capability of influencing their teammates. Robots are able to possess information that their human and other agent teammates do not, such as by scouting ahead in dangerous areas. To work as an effective team, robots must be able to influence their teammates when necessary and adapt to changing situations in order to move to goal positions that only they may be aware of, while remaining connected as a team. In this paper, we propose the problem of multiple robot teammates tasked with leading a multi-agent team to multiple goal positions while maintaining the ability to communicate with one another. We define utilities of making progress towards goals, maintaining communications with followers, and maintaining communications with fellow leaders. In addition, we introduce a novel regularized optimization formulation that balances these utilities and utilizes structured sparsity inducing norms to focus the leaders’ attention on specific goals and followers over time. The dynamically learned utility allows our approach to generate an action for each leader at each time step, which allows the leaders to reach goals without sacrificing communication. We show through extensive synthetic and high-fidelity simulations that our method effectively enables multiple robotic leaders to guide a multi-agent team to different goals while maintaining communication.

I. INTRODUCTION

Members in multi-agent teams must be able to work collaboratively to accomplish tasks in order to accomplish multiple simultaneous goals or to operate over large areas. In scenarios such as disaster response, teams of human rescuers are able to work together fluidly, utilizing leadership roles to accomplish tasks. When robots join human teams, they must join into this collective workflow [8]. Human teammates prefer that robot teammates take proactive roles, operating as peers instead of subordinates [36]. Robot teammates must be able to lead and influence the team, even when teamed only with other robots.

The ability for robots to influence the behavior of their team members is vital to the success of a wide variety of real-world applications. For example, during a disaster response scenario such as the one seen in Figure 1, robots may have information that their teammates lack, such as the locations of injured humans that they have found in a search or the locations of immediate dangers that must be addressed [10, 15]. Additionally, robots may not be able to explicitly relay instructions to their teammates, as the robots may not have authority to control their machine teammates, or civilians being rescued may not have the training necessary to interact with the robot [44]. In these situations, robots must have the capability of influencing their teammates to follow them [22]. For example, a group of first responders can follow a drone

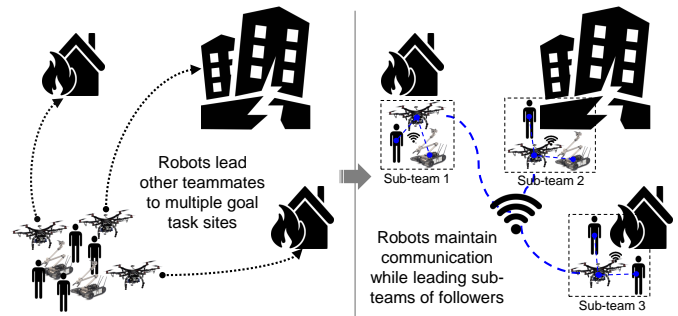


Fig. 1. An illustrative example in disaster response applications to motivate the problem of multiple robots tasked with leading other members in the team to multiple goals while adapting to changing situations as well as maintaining communication among sub-teams and also between leaders and followers.

they can see, and so the drone must behave in a way that keeps it visible to its followers while progressing towards a goal position, as illustrated in Figure 1.

One key challenge of effectively influencing teams is doing so with multiple competing objectives and multiple leaders that can affect team behavior. Several methods have been designed that identify leaders within a multi-robot team [31], but they do not address how leaders can affect the team. Recent research has investigated both the leading of individuals [20], as well as the leading [22] and herding [41] of teams of followers. While effective, these approaches have been limited to leading followers to single goals, and have not addressed the problem of multiple goals or incorporated the use of multiple leaders.

The second critical challenge is maintaining communication among leaders and also between leaders and followers, which is a necessary functionality for safety-critical scenarios, such as disaster response. The maintenance of communication within a multi-robot system or during deployment of a multi-robot team has been approached with several recent methods [4, 26]. However, the challenge of maintaining communication while leading a team of followers to multiple goals has not yet been addressed, either from the perspective of ensuring leaders stay connected to their followers or ensuring leaders stay connected to other leaders.

In this paper, we propose a principled approach to adaptively lead a multi-agent team to multiple goal positions while maintaining communication. We define the utility of various actions that leaders can take, based on (1) making progress

towards goals, (2) staying in communication with sub-teams of followers, and (3) staying in communication with other leaders. We then introduce a mathematical formulation based on the unified framework of regularized optimization to find weights that optimally balance these competing utilities, which incorporates new structured sparsity inducing norms to focus each leader’s attention on specific goals and sub-teams of followers, as well as regularization terms to enforce temporal consistency as time passes. Then, our approach identifies the optimal actions for each leader at each point in time after the optimization over these competing utilities, in order for each leader to effectively influence a team of followers.

This paper presents three novel contributions:

- We introduce an essential but not well addressed problem of influencing a team such that its members can be guided to goal positions by leaders that maintain communication with the followers as well as other leaders.
- We propose a principled approach with a novel mathematical formulation to address the problem by optimizing the competing utilities to make progress towards goals while maintaining communication, which integrates novel regularization terms to focus leaders attention on select goals and followers while maintaining temporal consistency.
- We introduce an iterative solver for this formulated objective, which is hard to solve due to equality constraints and non-smooth terms. This solver is theoretically guaranteed to converge to the optimal solution.

II. RELATED WORK

A. Robot Leading and Following

Collaborative robots require coordination among the robots in a team, and many teams must operate with uncertainty [25] or without previous coordination [38]. Although this problem has been approached by collective means such as beacons [21], pheromones [3], and graphs [18], it can be more effectively enabled by the presence of leader and follower roles. Several methods have been proposed to select leaders in a multi-robot system, such as identifying agents with different capabilities [16], basing leadership on knowledge of the environment [27], adherence to a goal formation [31], or inferring leadership based on the network topology of the multi-robot system [42]. After leaders in a multi-robot team have been determined, they can be used to influence the remaining team members. A number of strategies have been developed to enable a human to directly influence a robotic swarm using leader agents, such as managing followers’ positioning through ‘attractors’ [7], affecting only followers orientation [12], or varying information propagation methods from leaders to followers [2, 43]. Multiple approaches have been developed to manage the behavior of the follower agents in these scenarios, from potential fields in simulation [5, 24] to physical robot following through camera tracking [17] or steering by braking on towed robots [19]. Leadership in robot teams simplifies many tasks, such as collaborative transport [9, 32] and position deployment [1].

As it requires different approaches than leading teams of other robots, the leading and influencing of teams of humans

or animals has seen much recent research. Influencing animals to move to a goal state was introduced with the solution of ‘herding by caging’ [41], or surrounding the animal group as the leader robots moved to the goal state, which required precise control. Influencing humans has the additional complication of humans not always behaving like robots or animals would [35, 40], though simulated models of groups of human followers have been developed [24]. Similar to approaches that identify a robotic leader, recent work has attempted to identify the leader in a group of human followers and to lead the group by influencing this single human [22]. In autonomous cars, planning models have considered the effects that an action would have on a nearby car [33, 34]. While leading an individual human on foot has seen some research, such as utilizing multiple robots [20], most work concerning individual humans has seen the human as the leader, with robots learning to follow by predicting trajectories [23, 37], even from in front of the human [29].

Despite the breadth of research, most methods have focused on single leaders or single goals. The problem of how to use multiple robots to lead follower agents to multiple goals, while maintaining communication, has not been fully addressed.

B. Communication Maintenance in Multi-Robot Teams

A key component of effectively leading a team, whether it consists of humans, robots, or both, is the maintenance of communication, both with followers and with other leaders. While this has not previously been studied in a leader/follower context, it has seen much work in the context of the deployment of multi-robot systems. Early work deployed robots as relays to ensure that line of sight would not be interrupted, thus maintaining visibility-based communication [39]. Similarly, later work continued to employ the geometry of indoor spaces as a constraint, which attempts to optimally assign a group of robots that would be unable to fully deploy and maintain communication [4]. Planning for deployment into a coordinated formation has also used the constraint that communication must be maintained [28]. Graph-based techniques have also been used, where metrics like algebraic connectivity can be utilized [13, 14, 30], or where a minimum edge connectivity must be maintained [26].

Maintaining communication in a leader and follower context is an important open problem. Communication maintenance will allow a multi-robot team to continue coordinating as they lead followers toward multiple goals, while also ensuring that leaders stay connected with their followers and other leaders. However, this challenge is not well studied yet in the problem of leading multi-agent teams to multiple goals.

III. THE PROPOSED APPROACH

In this section, we introduce our novel approach to leading a multi-agent team to multiple goal positions while maintaining communication among the sub-teams. Specifically, we discuss our problem formulation, the principled optimization method to address the problem, and the optimization algorithm to solve the formulated regularized optimization problem.

Notation. We denote matrices as uppercase bold letters and vectors as lowercase bold letters. Given a matrix $\mathbf{X} = [x_{ij}] \in \mathbb{R}^{m \times n}$, we denote its i -th column as \mathbf{x}_i and its j -th row as \mathbf{x}^j . Given a vector \mathbf{y} , its i -th element is denoted y_i . The position of the i -th leader is defined as \mathbf{p}_i , with the leader's movements denoted respectively as $\mathbf{p}_{i \rightarrow j}^G$ (towards the j -th goal), $\mathbf{p}_{i \rightarrow j}^F$ (towards the j -th follower), and $\mathbf{p}_{i \rightarrow j}^L$ (towards the j -th leader).

A. Problem Formulation

We address the problem of multiple robot teammates tasked with leading a team to multiple goal positions while maintaining the ability to communicate with one another. We refer to robots that perform the leading task and influence the team as *leaders*, and other teammates who are influenced to follow the leaders as *followers*. Leaders consider three utilities that are gained by making progress towards the goals, maintaining communication with a subset of followers, and maintaining communication with other leaders in the team that lead other followers, respectively.

Formally, we assume there exist G goals in an environment, as well as N leaders and F followers. Each of the G goals is modeled as a density function centered on \mathbf{g}_i for the i -th goal, which is represented as $\phi_i(\cdot)$ and returns a scalar value based on \mathbf{g}_i . Each leader and follower has a position, defined as \mathbf{p}_i for the i -th leader and \mathbf{f}_i for the i -th follower. Each leader is able to take actions to move within the environment, and is aware of the locations of the goals. Followers are limited to only moving towards leaders and other followers, basing their motion on potential field models [5, 22].

Both followers and leaders are only aware of others within a communication distance R . Then, we introduce a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ describing the communication relationships among the leaders and followers. $\mathcal{V} = \{v_1, \dots, v_{N+F}\}$ represents the set of vertices, with each vertex corresponding to a leader located at \mathbf{p} or a follower located at \mathbf{f} . $\mathcal{E} = \{e_{ij}\}$ is the set of edges, where e_{ij} is an edge between the i -th and j -th vertices, and the magnitude of e_{ij} represents the strength of this connection (i.e., the smaller distance between vertices, the larger the edge weight). Edges only exist if vertices are able to communicate:

$$e_{ij} = \begin{cases} \frac{1}{\text{dist}(v_i, v_j)} & \text{if } \text{dist}(v_i, v_j) < R \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

While our implementation uses the widely used circular communication model [26, 28], other multi-robot communication models computed from a pair of nodes can also be used.

We propose three utilities associated with leaders to decide actions. The first utility $\mathbf{R} = [r_{ij}] \in \mathbb{R}^{N \times G}$ is defined as the utility for leaders moving towards goals, where r_{ij} defines the utility of the i -th leader moving towards the j -th goal. This utility is based upon the derivative of ϕ_j with respect to \mathbf{p}_i , which is denoted as $\mathbf{p}_{i \rightarrow j}^G$. Then, the goal utility is based upon moving in the direction of this derivative, or mathematically:

$$\mathbf{R} = [r_{ij}] = \frac{\phi_j(\mathbf{p}_i + \mathbf{p}_{i \rightarrow j}^G)}{\text{dist}(\mathbf{p}_i, \mathbf{g}_j)} \quad (2)$$

where $\text{dist}(\mathbf{p}_i, \mathbf{g}_j)$ is a scaling factor and denotes the distance between the i -th leader and the j -th goal.

Second, we define the utility $\mathbf{Q} = [q_{ij}] \in \mathbb{R}^{N \times F}$ for leaders moving in the direction of followers in order to maintain or strengthen communications with their followers. Given \mathcal{G} , we represent this graph with an adjacency matrix \mathbf{A} , where a_{ij} is equal to the edge weight e_{ij} . Then, we compute the Laplacian of the graph $\mathbf{L} = \mathbf{D} - \mathbf{A}$, where \mathbf{D} is the degree matrix. As the second smallest eigenvalue (λ_2) of \mathbf{L} corresponds to the graph's algebraic connectivity (i.e., an unconnected graph will have $\lambda_2 = 0$, and a complete graph will have λ_2 equal to its vertex connectivity [11]), we base the utility \mathbf{Q} of maintaining communication links on the eigenvector corresponding to this eigenvalue, which is denoted as $\mathbf{z} \in \mathbb{R}^{N+F}$. Each element z_i of \mathbf{z} corresponds to a vertex v_i . If two vertices are strongly connected, their values z_i and z_j in this vector will be similar; conversely, the less connected two vertices are the larger the difference between z_i and z_j will be. We utilize this vector to calculate the utility of leaders moving towards followers; if a leader is already strongly connected to a follower, the utility of moving to improve this connection will be low. The follower communication utility for a leader \mathbf{p}_i moving towards a follower \mathbf{f}_j can be computed by:

$$\mathbf{Q} = [q_{ij}] = \frac{(z_i - z_j)^2}{\text{dist}(\mathbf{p}_i, \mathbf{f}_j)} \quad (3)$$

Third, we define the utility $\mathbf{S} \in \mathbb{R}^{N \times N}$ for leaders moving in the direction of other leaders in order to maintain communication within leaders leading other sub-teams of followers. Similarly, we compute this leader communication utility based on the values in \mathbf{z} , which can be expressed as:

$$\mathbf{S} = [s_{ij}] = \frac{(z_i - z_j)^2}{\text{dist}(\mathbf{p}_i, \mathbf{p}_j)} \quad (4)$$

Given the three defined utilities \mathbf{R} , \mathbf{Q} and \mathbf{S} , the objective in our problem formulation is to maximize the overall utility $\mathcal{U}(\mathbf{R}, \mathbf{Q}, \mathbf{S})$, generating actions for leaders that enable them to lead the team to multiple goals while maintaining communication.

B. Regularized Optimization for Leader Action Generation

As each leader has competing utility based on movement towards goals, followers, and other leaders, we introduce a formulation based on regularized optimization which finds the optimal balance of these competing interests for each leader. Additionally, we introduce sparsity inducing norms to control which goals and followers a leader values, and regularization terms to ensure that each leader's utility balance is temporally consistent.

We first introduce the initial objective function $\mathcal{U}(\mathbf{R}, \mathbf{Q}, \mathbf{S})$:

$$\max_{\mathbf{W}, \mathbf{V}, \mathbf{U}} \|\mathbf{W} \odot \mathbf{R}\|_1 + \|\mathbf{V} \odot \mathbf{Q}\|_1 + \|\mathbf{U} \odot \mathbf{S}\|_1 \quad (5)$$

where $\mathbf{W} \odot \mathbf{R}$ represents the Hadamard or element-wise matrix product and $\|\cdot\|_1$ is the element-wise ℓ_1 -norm, a summation of the absolute values of matrix elements. \mathbf{W} is a weight matrix where w_{ij} represents the weight that the i -th leader assigns to

the j -th goal, \mathbf{V} is a weight matrix where v_{ij} represents the weight that the i -th leader assigns to the j -th follower, and \mathbf{U} is a weight matrix where u_{ij} represents the weight that the i -th leader assigns to the j -th leader. By finding values for these weight matrices that maximize Eq. (5), our approach is able to adapt to changing situations and maintain a balance of the competing utilities of moving towards goals while maintaining communication with followers and leaders.

We introduce constraints to limit the values in the matrices:

$$\begin{aligned} \mathbf{W}\mathbf{1}_G = \mathbf{1}_N & \quad \mathbf{W} \geq 0 \\ \mathbf{V}\mathbf{1}_F = \mathbf{1}_N & \quad \mathbf{V} \geq 0 \\ \mathbf{U}\mathbf{1}_N = \mathbf{1}_N & \quad \mathbf{U} \geq 0 \end{aligned} \quad (6)$$

where $\mathbf{1}_N$ represents a vector of 1s of length N . These constraints restrict the weights such that the total weight assigned to each agent sums to 1, in order to prevent unreasonably large weights, and that all weights must be nonnegative.

The first sparsity inducing norm introduced is based on the idea that each leader should ideally only value one goal at a time; that is, a leader that is attempting to move towards two goals at a time will make less progress than a leader that is moving directly towards a single goal. To integrate this into our objective formulation, we introduce a *goal* norm that induces sparsity in an individual leader's weights assigned to goals. We do this in the form of the ℓ_2 -norm on each row of \mathbf{W} :

$$\|\mathbf{W}\|_G = \sum_{n=1}^N \|\mathbf{w}^n\|_2 \quad (7)$$

where $\|\mathbf{w}\|_2 = \sqrt{\sum_{j=1}^J |w_j|^2}$. Because each row in \mathbf{W} is constrained to be nonnegative and sum to 1, this norm has an upper bound of 1 for \mathbf{w}_i , indicating that the i -th leader is valuing only a single goal. This goal results in a lower value for leaders that value multiple goals at once (e.g., $\|[0.5, 0.5]\|_2 < \|[0, 1]\|_2$, and values in \mathbf{W} are bounded by 1). Adding this norm to our initial objective in Eq. (5) rewards leaders that value a single goal much higher than the others.

Second, we want leaders to be focused on only a subset of followers, such that multiple leaders are not attempting to lead the same follower. For this we introduce the *leader* norm:

$$\|\mathbf{V}\|_L = \sum_{f=1}^F \|\mathbf{v}_f\|_2 \quad (8)$$

where $\|\mathbf{v}\|_2$ again indicates the ℓ_2 -norm and in this case acts on each column of \mathbf{V} . This norm penalizes cases where more than one leader has assigned weights to a single follower. This norm adds to our objective function in Eq. (5), where increasing the value of this regularization term should increase the overall effectiveness of our approach.

Finally, we introduce the regularization term that enforces temporal consistency between the weight matrices. This desire for temporal consistency is based on the idea that if a leader has selected a goal to move towards, it is inefficient to abruptly switch to another goal. Thus the weights that leaders have at time t should be similar to the weights that leaders then obtain

at time $t+1$. We introduce three terms utilizing the Frobenius norm to calculate the value of this change from the previous time step:

$$\mathcal{T}(\mathbf{W}, \mathbf{V}, \mathbf{U}) = \|\mathbf{W} - \hat{\mathbf{W}}\|_F^2 + \|\mathbf{V} - \hat{\mathbf{V}}\|_F^2 + \|\mathbf{U} - \hat{\mathbf{U}}\|_F^2 \quad (9)$$

where $\hat{\mathbf{W}}$, $\hat{\mathbf{V}}$, and $\hat{\mathbf{U}}$ denote the optimized weight matrices from the previous step and $\|\mathbf{W}\|_F^2 = \sum_{i=1}^I \sum_{j=1}^J |w_{ij}|^2 = \text{tr}(\mathbf{W}\mathbf{W}^\top)$. The three terms combined in \mathcal{T} also act as penalty terms to the objective in Eq. (5).

Our final problem formulation integrates all of these introduced terms into the following convex objective function, under the unified framework of regularized constrained optimization:

$$\begin{aligned} \max_{\mathbf{W}, \mathbf{V}, \mathbf{U}} & \quad \|\mathbf{W} \odot \mathbf{R}\|_1 + \|\mathbf{V} \odot \mathbf{Q}\|_1 + \|\mathbf{U} \odot \mathbf{S}\|_1 \\ & \quad + \gamma_1 \|\mathbf{W}\|_G + \gamma_2 \|\mathbf{V}\|_L - \gamma_3 \mathcal{T}(\mathbf{W}, \mathbf{V}, \mathbf{U}) \\ \text{s.t.} & \quad \mathbf{W}\mathbf{1}_G = \mathbf{1}_N, \mathbf{V}\mathbf{1}_F = \mathbf{1}_N, \mathbf{U}\mathbf{1}_N = \mathbf{1}_N, \\ & \quad \mathbf{W} \geq 0, \mathbf{V} \geq 0, \mathbf{U} \geq 0. \end{aligned} \quad (10)$$

where $\gamma_i, i \in \{1, \dots, 3\}$ are hyperparameters controlling the importance of the defined regularization terms.

After computing \mathbf{W} , \mathbf{V} , and \mathbf{U} that represent the optimal allocation of the identified utility (for example, an element $w_{ij} \in \mathbf{W}$ weights the corresponding utility $r_{ij} \in \mathbf{R}$, with \mathbf{V} weighting \mathbf{Q} and \mathbf{U} weighting \mathbf{S}), we use these weight matrices to identify the optimal action for each leader.

As described earlier, $\mathbf{p}_{i \rightarrow j}^G$ is the movement of the i -th leader towards the j -th goal. We also consider the movement of the i -th leader towards the j -th follower, defined as $\mathbf{p}_{i \rightarrow j}^F$, and the movement of the i -th leader towards the j -th leader, defined as $\mathbf{p}_{i \rightarrow j}^L$. The overall movement $\dot{\mathbf{p}}_i$ for the i -th leader is based on a combination of these movements, weighted by \mathbf{W} , \mathbf{V} , and \mathbf{U} :

$$\dot{\mathbf{p}}_i = \sum_{j=1}^G w_{ij} \mathbf{p}_{i \rightarrow j}^G + \sum_{j=1}^F v_{ij} \mathbf{p}_{i \rightarrow j}^F + \sum_{j=1}^N u_{ij} \mathbf{p}_{i \rightarrow j}^L \quad (11)$$

This overall movement update $\dot{\mathbf{p}}_i$ is scaled to unit length and added to the previous position \mathbf{p}_i to arrive at the new position:

$$\mathbf{p}_i = \mathbf{p}_i + \frac{\dot{\mathbf{p}}_i}{\|\dot{\mathbf{p}}_i\|_2} \quad (12)$$

C. Optimization Algorithm

The formulated objective in Eq. (10) is convex, yet hard to solve due to equality constraints and non-smooth regularization terms. We propose an iterative algorithm, which is proven to converge to the optimal solution.

First, we note that $\|\mathbf{W} \odot \mathbf{R}\|_1$ is the sum of the entries of the Hadamard product of \mathbf{W} and \mathbf{R} . This can be rewritten as the inner Frobenius product, which is equal to the trace of the matrix product of \mathbf{W}^\top and \mathbf{R} :

$$\|\mathbf{W} \odot \mathbf{R}\|_1 = \langle \mathbf{W}, \mathbf{R} \rangle_F = \text{tr}(\mathbf{W}^\top \mathbf{R}) \quad (13)$$

We then incorporate this as we rewrite our objective function as a minimization problem:

$$\begin{aligned} \min_{\mathbf{W}, \mathbf{V}, \mathbf{U}} \quad & -\text{tr}(\mathbf{W}^\top \mathbf{R}) - \text{tr}(\mathbf{V}^\top \mathbf{Q}) - \text{tr}(\mathbf{U}^\top \mathbf{S}) \\ & - \gamma_1 \|\mathbf{W}\|_G - \gamma_2 \|\mathbf{V}\|_L + \gamma_3 \mathcal{T}(\mathbf{W}, \mathbf{V}, \mathbf{U}) \\ \text{s.t.} \quad & \mathbf{W}\mathbf{1}_G = \mathbf{1}_N, \mathbf{V}\mathbf{1}_F = \mathbf{1}_N, \mathbf{U}\mathbf{1}_N = \mathbf{1}_N, \\ & \mathbf{W} \geq 0, \mathbf{V} \geq 0, \mathbf{U} \geq 0. \end{aligned} \quad (14)$$

To ensure we satisfy our equality constraints, we introduce a solution based on the Augmented Lagrange Multiplier (ALM) method, which solves problems of the form

$$\min f(\mathbf{X}) \quad \text{s.t.} \quad h(\mathbf{X}) = 0$$

by rewriting constraints as penalty terms to arrive at a generalized ALM form of

$$f(\mathbf{X}) + \frac{\mu}{2} \|h(\mathbf{X}) + \frac{1}{\mu} \mathbf{\Lambda}\|_F^2 \quad (15)$$

where μ is a penalty coefficient and $\mathbf{\Lambda}$ is a Lagrangian multiplier. We also introduce the additional constraints of $\mathbf{A} = \mathbf{W}$, $\mathbf{B} = \mathbf{V}$, and $\mathbf{C} = \mathbf{U}$. With these new variables, we can write our final objective in generalized ALM form as

$$\begin{aligned} \min_{\mathbf{W}, \mathbf{V}, \mathbf{U}, \mathbf{A}, \mathbf{B}, \mathbf{C}} \quad & -\text{tr}(\mathbf{W}^\top \mathbf{R}) - \text{tr}(\mathbf{V}^\top \mathbf{Q}) - \text{tr}(\mathbf{U}^\top \mathbf{S}) \\ & - \gamma_1 \|\mathbf{W}\|_G - \gamma_2 \|\mathbf{V}\|_L + \gamma_3 \mathcal{T}(\mathbf{W}, \mathbf{V}, \mathbf{U}) \\ & + \frac{\mu}{2} \|\mathbf{A}\mathbf{1}_G - \mathbf{1}_N + \frac{1}{\mu} \lambda_1 \mathbf{1}_G\|_2^2 + \frac{\mu}{2} \|\mathbf{W} - \mathbf{A} + \frac{1}{\mu} \mathbf{\Lambda}_2\|_F^2 \\ & + \frac{\mu}{2} \|\mathbf{B}\mathbf{1}_F - \mathbf{1}_N + \frac{1}{\mu} \lambda_3 \mathbf{1}_F\|_2^2 + \frac{\mu}{2} \|\mathbf{V} - \mathbf{B} + \frac{1}{\mu} \mathbf{\Lambda}_4\|_F^2 \\ & + \frac{\mu}{2} \|\mathbf{C}\mathbf{1}_N - \mathbf{1}_N + \frac{1}{\mu} \lambda_5 \mathbf{1}_N\|_2^2 + \frac{\mu}{2} \|\mathbf{U} - \mathbf{C} + \frac{1}{\mu} \mathbf{\Lambda}_6\|_F^2 \\ \text{s.t.} \quad & \mathbf{W} \geq 0, \mathbf{V} \geq 0, \mathbf{U} \geq 0. \end{aligned} \quad (16)$$

and find a solution by iteratively solving for \mathbf{W} , \mathbf{V} , \mathbf{U} , \mathbf{A} , \mathbf{B} , and \mathbf{C} . We initialize $\mathbf{W} = [w_{ij}] = \frac{1}{G}$, $\mathbf{V} = [v_{ij}] = \frac{1}{F}$, $\mathbf{U} = [u_{ij}] = \frac{1}{N}$, $\mathbf{A} = \mathbf{W}$, $\mathbf{B} = \mathbf{V}$, and $\mathbf{C} = \mathbf{U}$. This approach is defined in Algorithm 1 and detailed below.

Line 5. First, we solve for \mathbf{W} by taking the derivative of Eq. (16) w.r.t. \mathbf{W} and setting it equal to 0:

$$-\mathbf{R} + \gamma_1 \mathbf{D}\mathbf{W} + 2\gamma_3 \mathbf{W} - 2\gamma_3 \hat{\mathbf{W}} + \mu \mathbf{W} - \mu \mathbf{A} + \mathbf{\Lambda}_2 = 0$$

after rearrangement, we update \mathbf{W} by

$$\begin{aligned} \mathbf{W} &= (\gamma_1 \mathbf{D} + 2\gamma_3 \mathbf{I} + \mu \mathbf{I})^{-1} (\mathbf{R} + 2\gamma_3 \hat{\mathbf{W}} + \mu \mathbf{A} - \mathbf{\Lambda}_2) \\ \mathbf{W} &= \max(\mathbf{W}, 0) \end{aligned} \quad (17)$$

where $\mathbf{D} \in \mathbb{R}^{N \times N}$ is a diagonal matrix with the i -th diagonal element as $\frac{1}{2\|\mathbf{w}^i\|_2}$ and where $\max(\mathbf{W}, 0)$ satisfies the $\mathbf{W} \geq 0$ constraint.

Line 6. Next, we solve for \mathbf{V} by taking the derivative of Eq. (16) w.r.t. \mathbf{V} and setting it equal to 0:

$$-\mathbf{Q} - \gamma_2 \mathbf{V}\hat{\mathbf{D}} + 2\gamma_3 \mathbf{V} - 2\gamma_3 \hat{\mathbf{V}} + \mu \mathbf{V} - \mu \mathbf{B} + \mathbf{\Lambda}_4 = 0$$

where after rearrangement we see that the update to \mathbf{V} is

$$\begin{aligned} \mathbf{V} &= (\mathbf{Q} + 2\gamma_3 \hat{\mathbf{V}} + \mu \mathbf{B} - \mathbf{\Lambda}_4) (\mu \mathbf{I} + 2\gamma_3 \mathbf{I} - \gamma_2 \hat{\mathbf{D}})^{-1} \\ \mathbf{V} &= \max(\mathbf{V}, 0) \end{aligned} \quad (18)$$

where $\hat{\mathbf{D}} \in \mathbb{R}^{F \times F}$ is a diagonal matrix with the i -th diagonal element as $\frac{1}{2\|\mathbf{v}^i\|_2}$ and where again $\max(\mathbf{V}, 0)$ satisfies the $\mathbf{V} \geq 0$ constraint.

Line 7. Next, we solve for the third weight matrix \mathbf{U} by again taking the derivative of Eq. (16) w.r.t. \mathbf{U} :

$$-\mathbf{S} + 2\gamma_3 \mathbf{U} - 2\gamma_3 \hat{\mathbf{U}} + \mu \mathbf{U} - \mu \mathbf{C} + \mathbf{\Lambda}_6 = 0$$

and rearranging to show that the update to \mathbf{U} at each iteration is

$$\begin{aligned} \mathbf{U} &= (2\gamma_3 \mathbf{I} + \mu \mathbf{I})^{-1} (\mathbf{S} + 2\gamma_3 \hat{\mathbf{U}} + \mu \mathbf{C} - \mathbf{\Lambda}_6) \\ \mathbf{U} &= \max(\mathbf{U}, 0) \end{aligned} \quad (19)$$

where again we satisfy the $\mathbf{U} \geq 0$ constraint with $\max(\mathbf{U}, 0)$.

Line 8. Next we solve for the first introduced variable \mathbf{A} , where the derivative of Eq. (16) w.r.t. \mathbf{A} is:

$$\mu \mathbf{A}\mathbf{1}_G \mathbf{1}_G^\top - \mu \mathbf{1}_N \mathbf{1}_G^\top + \lambda_1 \mathbf{1}_G^\top - \mu \mathbf{W} + \mu \mathbf{A} - \mathbf{\Lambda}_2 = 0$$

and the update to \mathbf{A} at each iteration is:

$$\mathbf{A} = (\mathbf{W} + \frac{1}{\mu} \mathbf{\Lambda}_2 + \mathbf{1}_N \mathbf{1}_G^\top - \frac{1}{\mu} \lambda_1 \mathbf{1}_G^\top) (\mathbf{I} + \mathbf{1}_G \mathbf{1}_G^\top)^{-1} \quad (20)$$

Line 9. Similarly, we show that the derivative of Eq. (16) w.r.t. \mathbf{B} is:

$$\mu \mathbf{B}\mathbf{1}_F \mathbf{1}_F^\top - \mu \mathbf{1}_N \mathbf{1}_F^\top + \lambda_3 \mathbf{1}_F^\top - \mu \mathbf{V} + \mu \mathbf{B} - \mathbf{\Lambda}_4 = 0$$

and the update to \mathbf{B} is:

$$\mathbf{B} = (\mathbf{V} + \mathbf{1}_N \mathbf{1}_F^\top - \frac{1}{\mu} \lambda_3 \mathbf{1}_F^\top + \frac{1}{\mu} \mathbf{\Lambda}_4) (\mathbf{I} + \mathbf{1}_F \mathbf{1}_F^\top)^{-1} \quad (21)$$

Line 10. And that the derivative of Eq. (16) w.r.t. \mathbf{C} is:

$$\mu \mathbf{C}\mathbf{1}_N \mathbf{1}_N^\top - \mu \mathbf{1}_N \mathbf{1}_N^\top + \lambda_5 \mathbf{1}_N^\top + \mu \mathbf{C} - \mu \mathbf{U} - \mathbf{\Lambda}_6 = 0$$

so the update to \mathbf{C} at each iteration is:

$$\mathbf{C} = (\mathbf{U} + \mathbf{1}_N \mathbf{1}_N^\top - \frac{1}{\mu} \lambda_5 \mathbf{1}_N^\top + \frac{1}{\mu} \mathbf{\Lambda}_6) (\mathbf{I} + \mathbf{1}_N \mathbf{1}_N^\top)^{-1} \quad (22)$$

Lines 11 – 17. Finally, we update each of the Lagrangian multipliers λ_1 , $\mathbf{\Lambda}_2$, λ_3 , $\mathbf{\Lambda}_4$, λ_5 , and $\mathbf{\Lambda}_6$ by the generalized formula of $\lambda = \lambda + \mu h(\mathbf{X})$, with the specific updates for each multiplier in Algorithm 1, and update the penalty coefficient μ by $\mu = \rho \mu$, where ρ is chosen such that $1 < \rho < 2$.

Convergence. Our optimization algorithm converges to the optimal solution. To prove that Algorithm 1 converges, we first present a lemma:

Lemma 1: The generalized ALM approach, when applied to problems of the form in Eq. (15), reduces the objective value of the the constrained optimization problem at each iteration and converges to the optimal solution when $0 < \mu^k < \mu^{k+1}$ is satisfied for every iteration k [6].

Using Lemma 1, we can prove the following theorem:

Theorem 1: Algorithm 1 iteratively decreases the objective value of the regularized optimization problem in Eq. (10) and converges to the global optimal solution.

Proof: To begin Algorithm 1, our objective function in Eq. (10) is converted into the generalized ALM format of Eq. (15). In **Line 1** of Algorithm 1, we define $\mu^0 > 0$. Thus at

Algorithm 1: Our Algorithm to Solve Eq. (10).

```
1: Set  $1 < \rho < 2$  and  $k = 0$ . Initialize the penalty coefficient
    $\mu^0$  and the multiplier terms  $\lambda_1^0, \Lambda_2^0, \lambda_3^0, \Lambda_4^0, \lambda_5^0$ , and  $\Lambda_6^0$ .
   Initialize each weight matrix  $\mathbf{W}, \mathbf{V}, \mathbf{U}$ , and the introduced
   matrices  $\mathbf{A}, \mathbf{B}$ , and  $\mathbf{C}$ .
2: repeat
3:   Compute  $\mathbf{D} = \text{diag} \left( \frac{1}{2\|\mathbf{w}^i\|_2} \right)$ .
4:   Compute  $\hat{\mathbf{D}} = \text{diag} \left( \frac{1}{2\|\mathbf{v}_i\|_2} \right)$ .
5:   Compute  $\mathbf{W}^{k+1}$  by Eq. (17).
6:   Compute  $\mathbf{V}^{k+1}$  by Eq. (18).
7:   Compute  $\mathbf{U}^{k+1}$  by Eq. (19).
8:   Compute  $\mathbf{A}^{k+1}$  by Eq. (20).
9:   Compute  $\mathbf{B}^{k+1}$  by Eq. (21).
10:  Compute  $\mathbf{C}^{k+1}$  by Eq. (22).
11:  Update  $\lambda_1$  by  $\lambda_1^{k+1} = \lambda_1^k + \mu^k(\mathbf{A}\mathbf{1}_G - \mathbf{1}_N)$ .
12:  Update  $\Lambda_2$  by  $\Lambda_2^{k+1} = \Lambda_2^k + \mu^k(\mathbf{W} - \mathbf{A})$ .
13:  Update  $\lambda_3$  by  $\lambda_3^{k+1} = \lambda_3^k + \mu^k(\mathbf{B}\mathbf{1}_F - \mathbf{1}_N)$ .
14:  Update  $\Lambda_4$  by  $\Lambda_4^{k+1} = \Lambda_4^k + \mu^k(\mathbf{V} - \mathbf{B})$ .
15:  Update  $\lambda_5$  by  $\lambda_5^{k+1} = \lambda_5^k + \mu^k(\mathbf{C}\mathbf{1}_N - \mathbf{1}_N)$ .
16:  Update  $\Lambda_6$  by  $\Lambda_6^{k+1} = \Lambda_6^k + \mu^k(\mathbf{U} - \mathbf{C})$ .
17:  Update  $\mu$  by  $\mu^{k+1} = \rho\mu^k$ .
18:   $k = k + 1$ .
19: until convergence;
```

$k = 0$, Lemma 1 holds. Now, we prove by contradiction that Lemma 1 holds for every set of iterations k and $k + 1$.

First, we consider the possibility that for some iteration k , $\mu^{k+1} < \mu^k$. If this case were true, then $(\mu^{k+1}) / (\mu^k) < 1$. By rearranging **Line 17**, $(\mu^{k+1}) / (\mu^k) = \rho$, which would mean that $\rho < 1$. In **Line 1** ρ is initialized such that $1 < \rho < 2$ and the value of ρ is never altered in Algorithm 1, so this case cannot occur. Next, consider the possibility that for some iteration k , $\mu^{k+1} = \mu^k$. If this case were true, then $(\mu^{k+1}) / (\mu^k) = 1 = \rho$. Again, ρ is initialized such that $1 < \rho < 2$ in **Line 1** and its value is never altered, so the case that $\mu^{k+1} = \mu^k$ could not occur.

As $\mu^{k+1} = \mu^k$ cannot occur and $\mu^{k+1} < \mu^k$ cannot occur, then it follows that $\mu^{k+1} > \mu^k$. As this holds for any given iteration k , Lemma 1 is satisfied at each iteration and Algorithm 1 converges to the optimal solution. ■

Complexity. The computational complexity of our approach is dominated by **Lines 5 – 10**. We note that **Lines 3 and 4** are computed in linear time, as are the multiplier and coefficient updates in **Lines 11 – 18**. Each of **Lines 5 – 10** utilizes a matrix inverse and multiplication, with respective complexities of $\mathcal{O}(N^2G + N^3)$, $\mathcal{O}(NF^2 + F^3)$, $\mathcal{O}(2N^3)$, $\mathcal{O}(NG^2 + G^3)$, $\mathcal{O}(NF^2 + F^3)$, and $\mathcal{O}(2N^3)$ based on the dimensions of the matrices involved. As a result, the overall computational complexity of our described approach in Algorithm 1 is $\mathcal{O}(X^3)$ where $X = \max(N, G, F)$.

IV. EXPERIMENTAL RESULTS

A. Experiment Setup

We evaluate our approach using extensive synthetic simulations as well as a multi-robot simulator that incorporates Unity

for high-fidelity graphics with ROS for multi-robot control. We utilize three metrics for performance evaluation:

- 1) *Time Steps* is the average number of time steps required for followers to reach goal positions. We cap this at a maximum of 500 (i.e., if followers have not all reached goals at the 500th time step, we end the simulation).
- 2) *Average Follower to Goal Distance* is the average distance from each follower to the nearest goal: $\frac{\sum_{i=1}^F \text{dist}(\mathbf{f}_i, \mathbf{g}_j)}{F}$ where \mathbf{g}_j is the goal nearest to \mathbf{f}_i . The average initial follower to goal distance was 62.6, and the optimal final value would be 0 (as this indicates all followers have reached a goal).
- 3) *Average Goal to Follower Distance* is the average distance from each goal to the nearest follower: $\frac{\sum_{i=1}^G \text{dist}(\mathbf{g}_i, \mathbf{f}_j)}{G}$ where \mathbf{f}_j is the follower closest to \mathbf{g}_i . The average initial goal to follower distance was 64.3, and again the optimal final value is 0 (as this indicates each goal was reached by a follower). In cases where only a subset of goals are reached (e.g., if three goals exist, but leaders only guide followers to two of the goals), then the follower to goal distance could be 0 while the goal to follower distance is still high.

We compare to three approaches: (1) a *random* approach [22, 27] as it is often used to provide a grounding for the other approaches, where each leader moves in a random direction at each time step; (2) a *greedy* approach, where each leader moves directly towards the nearest goal position; and (3) a *baseline* approach, where we set $\gamma_1 = \gamma_2 = \gamma_3 = 0$ in order to evaluate the effectiveness of the described regularization terms in our formulation.

B. Results on Synthetic Simulations

We conduct synthetic simulations to provide thorough quantitative results based on different experiment setups. Followers and leaders were created at random initial positions, utilizing a uniform Gaussian distribution centered in the middle of the environment. Goals were also randomly created, based upon a uniform distribution describing a circle centered on the middle of the environment with a radius of 60.

We evaluate *Scenario A* with three leaders, nine followers, and three goals ($N = 3, F = 9, G = 3$). Figure 2 shows a sample synthetic simulation. The initial state of the system is shown in Figure 2(a), where the agents are generated in the middle of the environment and the three goals are generated on the outskirts of the environment. Figure 2(b) shows an ensuing time step, where each leader has begun to make progress towards a goal. It is observed that the left and right leaders have not headed directly for their nearest goals, but have also moved up to ensure they maintain communication with the middle leader. This time step also shows the initial follower team splitting up to follow individual leaders. Figure 2(c) again shows a later time step, where the middle leader has nearly reached a goal state and the left and right leaders have moved more in the direction of their nearest goals. Figure 2(d) shows the final state. At this point, the middle leader and

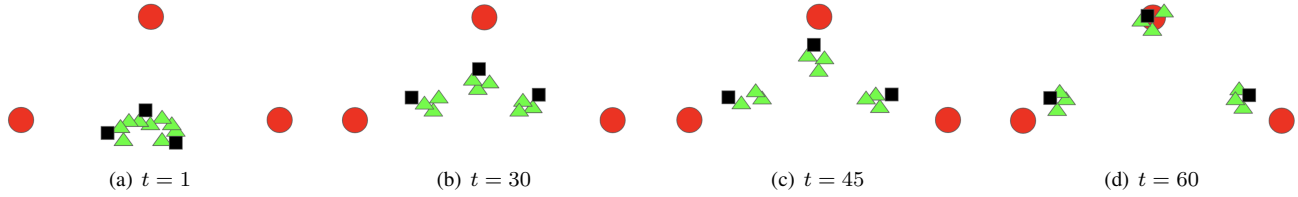


Fig. 2. Sample snapshots from a synthetic simulation. Leaders are marked by black squares, followers by green triangles, and goals by red circles.

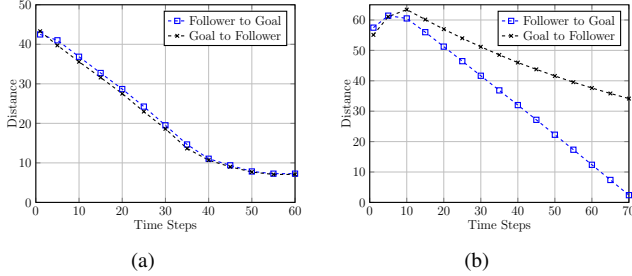


Fig. 3. Quantitative results of follower to goal distance and goal to follower distance for two simulations.

associated followers have reached a goal state. Both the left and right leaders are near goal states, but in order to maintain communication with the middle leader they will not progress further.

Figure 3(a) shows the follower to goal and goal to follower distances over time for the simulation shown in Figure 2. As leaders attempted to reach all three possible goals, these two distances track very closely. We can see in this graph that the slope is slightly flatter at the beginning of the simulation, as leaders identify what goals to progress towards and followers start tracking leaders. We also see the near the end of the simulation progress slows, as the left and right leaders cease progressing towards their goals. Figure 3(b) shows these distances over time for a simulation where leaders moved to only two out of three goals, with two leaders moving to a single goal, leaving one unapproached. Here, the follower to goal distance decreases linearly, as followers still move towards their local leaders at a constant rate. However, the goal to follower distance does not approach zero as one goal is not approached, and so as followers settle at the other two goal positions this value stabilizes at a level above zero.

Then, we also evaluate *Scenario B* with $N = 4, F = 16$, and $G = 4$; and *Scenario C* with $N = 5, F = 25$, and $G = 5$. These varying scenario sizes allow us to evaluate our approach as the number of team members present changes. Figure 4 shows the quantitative results for each approach on 20 simulations of each scenario. In Figure 4(a) we show the average number of time steps required for a simulation to complete. We see that our full approach significantly outperforms both the baseline version as well as the greedy and random methods. The full approach takes progressively longer as the scenario size increases, as this increases the chance of not being able to reach all goals. Our baseline approach is relatively consistent, but significantly worse than

the full approach, showing the value of the introduced terms. The greedy approach is relatively successful with the smaller number of robots in Scenario A, but performs significantly worse as the scenario size grows. This suggests that with a small number of followers, progressing directly towards goals without attempting to maintain communication can still be successful sometimes, but as the numbers of followers rises this becomes less likely. The random method consistently performs poorly, with only a single simulation for each scenario not reaching the maximum 500 time steps.

Figure 4(b) shows the average follower to goal distance at the end of each simulation. Again, we see that our approach performs consistently well, with average distances below 10 for all three scenarios. The baseline approach performs the second best, showing the importance of the regularization terms. The greedy method again declines in performance as the size of the scenario increases, showing again that larger numbers of followers cannot be led by leaders that do not attempt to maintain contact. Finally, the random method is again the worst, actually increasing the distance between followers and goals in all three scenarios.

Figure 4(c) shows the average goal to follower distance at the end of each simulation. Here, we see similar patterns as the follower to goal distance, where our full approach is consistently the best performer, and the baseline version is less effective but still outperforms the greedy and random approaches. Our approach does not decrease the goal to follower distance as effectively as the follower to goal distance. In some simulations, leaders are omitting individual goals, and leading followers to a subset of the multiple goals possible.

C. Results on High-Fidelity Simulations

We further evaluated our approach in a high-fidelity multi-robot simulator in order to implement the approach in ROS and understand its effectiveness in a more realistic environment. Figure 5 shows sample snapshots of a multi-robot team that must proceed through the urban environment to goal positions. In this simulation, the quadrotor robots are leaders and the wheeled Husky robots are followers.

Figure 5(a) shows the initial positions of the robots, with all eight being generated near a single point. In this frame, one goal position is straight across the intersection, and a second goal position is down the road that branches off to the left. Figure 5(b) shows the multi-robot team as it progresses into the intersection. Each leader has begun to head in the direction of an individual goal, and a subset of the Husky robots is

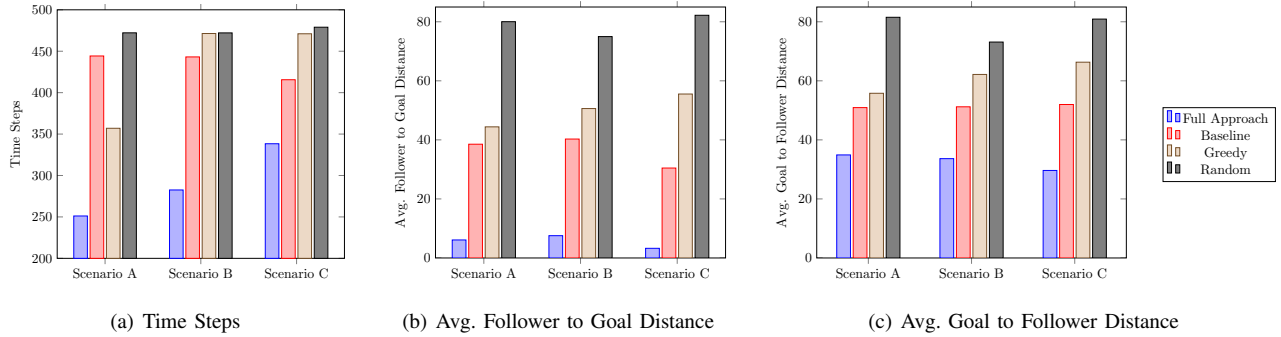


Fig. 4. Quantitative results for the four evaluated approaches on Scenarios A, B, and C.

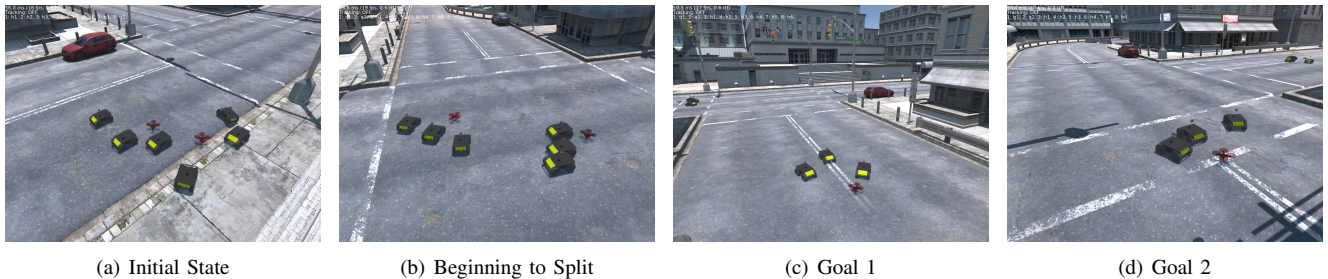


Fig. 5. Example snapshots from a high-fidelity multi-robot simulation.

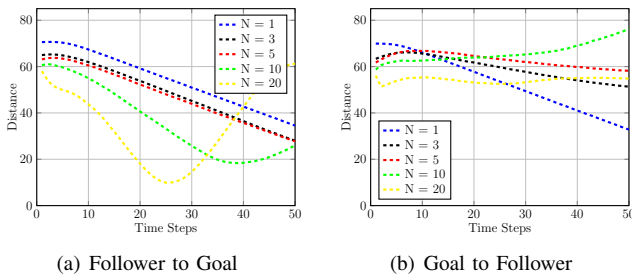


Fig. 6. Quantitative results of our approach's scalability.

following each leader. Figure 5(c) shows one team arriving at a goal position. In this frame, the remaining robots can be seen in the top left corner at the other goal position. Similarly, Figure 5(d) shows this second team at their goal position, with the first team being seen in the background in the top right, at the first goal position.

D. Discussion

Runtime. Since the optimization problem solved in each iteration in Eq. (16) is convex, the proposed algorithm converges to the global optimal solution very quickly. In practice, for a problem with 5 leaders, 25 followers, and 5 goals, Algorithm 1 uses on average < 50 iterations to converge. When running on a computer with an i5 processor and 8 Gb memory, our approach executes at a high speed of about 170 Hz for the same problem.

Scalability. Figure 6 demonstrates average follower to goal distance and goal to follower distance for the three evaluated scenarios as well as scenarios with 10 and 20 leaders. Each

scenario (with $G = N$ and $F = N^2$) was simulated 100 times. Figure 6(a) shows that our approach decreases follower to goal distance for all system sizes, but that larger systems do not preserve this low distance. Figure 6(b) reports a similar effect, where goal to follower distance decreases consistently for smaller systems. However, as the number of goals increases, the likelihood of a goal being ignored increases as well, and so larger systems do not show a consistent decline in this metric.

V. CONCLUSION

In this paper, we introduce the research problem of multiple robots tasked with leading other teammates to multiple goals while maintaining communication. We introduce a principled method based upon regularized optimization to learn the optimal balance of competing utilities, utilizing sparsity-inducing norms to focus leaders on specific goals and followers and regularization terms to enforce temporal consistency as robots progress through the environment. We use the learned balance of utilities to generate actions for each leader at each time step, which allow leaders to adaptively make progress towards the goal positions without losing communication both with followers and fellow leaders. The experimental results have shown that our approach enables robots to lead multi-agent teams, simultaneously progressing to multiple goals and maintaining communication.

ACKNOWLEDGEMENTS

This work was supported by NSF CAREER Award IIS-1942056 and ARL DCIST CRA W911NF-17-2-0181. The authors also thank the anonymous reviewers and the area chair for their feedback.

REFERENCES

- [1] Saurav Agarwal and Srinivas Akella. Simultaneous optimization of assignments and goal formations for multiple robots. In *International Conference on Robotics and Automation*, 2018.
- [2] Saman Amirpour Amraii, Phillip Walker, Michael Lewis, Nilanjan Chakraborty, and Katia Sycara. Explicit vs. tacit leadership in influencing the behavior of swarms. In *International Conference on Robotics and Automation*, 2014.
- [3] Farshad Arvin, Ali Emre Turgut, Tomáš Krajník, Salar Rahimi, Ilkin Ege Okay, Shigang Yue, Simon Watson, and Barry Lennox. Φ clust: pheromone-based aggregation for robotic swarms. In *International Conference on Intelligent Robots and Systems*, 2018.
- [4] Jacopo Banfi, Nicola Basilico, and Stefano Carpin. Optimal redeployment of multirobot teams for communication maintenance. In *International Conference on Intelligent Robots and Systems*, 2018.
- [5] Jerome Barraquand, Bruno Langlois, and J-C Latombe. Numerical potential field techniques for robot path planning. *Transactions on Systems, Man, and Cybernetics*, 22(2):224–241, 1992.
- [6] Dimitri P Bertsekas. *Constrained optimization and lagrange multiplier methods*. Academic press, 2014.
- [7] Daniel S Brown, Sean C Kerman, and Michael A Goodrich. Human-swarm interactions based on managing attractors. In *International Conference on Human-Robot Interaction*, 2014.
- [8] Filipa Correia, Sofia Petisca, Patricia Alves-Oliveira, Tiago Ribeiro, Francisco S Melo, Ana Paiva, et al. Groups of humans and robots: understanding membership preferences and team formation. In *Robotics: Science and Systems*, 2017.
- [9] Hector Garcia de Marina and Ewoud Smeur. Flexible collaborative transportation by a team of rotorcraft. In *International Conference on Robotics and Automation*, 2019.
- [10] Milan Erdelj, Michał Król, and Enrico Natalizio. Wireless sensor networks and multi-UAV systems for natural disaster management. *Computer Networks*, 124:72–86, 2017.
- [11] Miroslav Fiedler. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, 23(2):298–305, 1973.
- [12] Katie Genter, Noa Agmon, and Peter Stone. Ad hoc teamwork for leading a flock. In *International Conference on Autonomous Agents and Multi-Agent Systems*, 2013.
- [13] Arpita Ghosh and Stephen Boyd. Growing well-connected graphs. In *Conference on Decision and Control*, 2006.
- [14] Arpita Ghosh and Stephen Boyd. Upper bounds on algebraic connectivity via convex optimization. *Linear Algebra and Its Applications*, 418(2-3):693–707, 2006.
- [15] Alex Goldhoorn, Anaís Garrell, René Alquézar, and Alberto Sanfeliu. Searching and tracking people with cooperative mobile robots. *Autonomous Robots*, 42(4):739–759, 2018.
- [16] Michael A Goodrich, Sean Kerman, and Shin-Young Jun. On leadership and influence in human-swarm interaction. In *AAAI Fall Symposium Series*, 2012.
- [17] Dejun Guo, Hesheng Wang, Weidong Chen, Ming Liu, Zeyang Xia, and Kam K Leang. A unified leader-follower scheme for mobile robots with uncalibrated on-board camera. In *International Conference on Robotics and Automation*, 2017.
- [18] Meng Guo, Magnus Egerstedt, and Dimos V Dimarogonas. Hybrid control of multi-robot systems using embedded graph grammars. In *International Conference on Robotics and Automation*, 2016.
- [19] Yasuhisa Hirata, Ken Kimura, Shin Matsuzaki, Naoko Ogawa, and Takashi Kubota. Control of multiple passive-follower type robots based on feasible braking control region analysis. In *International Conference on Robotics and Automation*, 2018.
- [20] Piyush Khandelwal and Peter Stone. Leading the way: an efficient multi-robot guidance system. In *AAAI Fall Symposium Series*, 2014.
- [21] Andreas Kolling, Katia Sycara, Steven Nunnally, and Michael Lewis. Human-swarm interaction: An experimental study of two types of interaction with foraging swarms. *Journal of Human-Robot Interaction*, 2(2):103–129, 2013.
- [22] Minae Kwon, Mengxi Li, Alexandre Bucquet, and Dorsa Sadigh. Influencing leading and following in human-robot teams. In *Robotics: Science and Systems*, 2019.
- [23] Beom-Jin Lee, Jinyoung Choi, Christina Baek, and Byoung-Tak Zhang. Robust human following by deep bayesian trajectory prediction for home service robots. In *International Conference on Robotics and Automation*, 2018.
- [24] Tsai-Yen Li, Ying-Jiun Jeng, and Shih-I Chang. Simulating virtual human crowds with a leader-follower model. In *Conference on Computer Animation*, 2001.
- [25] Miao Liu, Kavinayan Sivakumar, Shayegan Omidshafiei, Christopher Amato, and Jonathan P How. Learning for multi-robot cooperation in partially observable stochastic environments with macro-actions. In *International Conference on Intelligent Robots and Systems*, 2017.
- [26] Wenhao Luo and Katia Sycara. Minimum k-connectivity maintenance for robust multi-robot systems. In *International Conference on Intelligent Robots and Systems*, 2019.
- [27] Wenhao Luo, Shehzaman S Khatib, Sasanka Nagavalli, Nilanjan Chakraborty, and Katia Sycara. Distributed knowledge leader selection for multi-robot environmental sampling under bandwidth constraints. In *International Conference on Intelligent Robots and Systems*, 2016.
- [28] Nathalie Majcherczyk, Adhavan Jayabalan, Giovanni Beltrame, and Carlo Pinciroli. Decentralized connectivity-preserving deployment of large-scale robot swarms. In *International Conference on Intelligent Robots and Systems*, 2018.
- [29] Payam Nikdel, Rakesh Shrestha, and Richard Vaughan. The hands-free push-cart: autonomous following in front by predicting user trajectory around obstacles. In *International Conference on Robotics and Automation*, 2018.
- [30] Brian Reily, Christopher Reardon, and Hao Zhang. Representing multi-robot structure through multimodal graph embedding for the selection of robot teams. In *International Conference on Robotics and Automation*, 2020.
- [31] Dany Rovinsky and Noa Agmon. Uncertain local leader selection in distributed formations. In *International Conference on Intelligent Robots and Systems*, 2018.
- [32] Michael Rubenstein, Adrian Cabrera, Justin Werfel, Golnaz Habibi, James McLurkin, and Radhika Nagpal. Collective transport of complex objects by simple robots: theory and experiments. In *International Conference on Autonomous Agents and Multi-Agent Systems*, 2013.
- [33] Dorsa Sadigh, Shankar Sastry, Sanjit A Seshia, and Anca D Dragan. Planning for autonomous cars that leverage effects on human actions. In *Robotics: Science and Systems*, 2016.
- [34] Dorsa Sadigh, Nick Landolfi, Shankar S Sastry, Sanjit A Seshia, and Anca D Dragan. Planning for cars that coordinate with people: leveraging effects on human actions for planning and active information gathering over human internal state. *Autonomous Robots*, 42(7):1405–1426, 2018.
- [35] Michaéla C Schippers, Deanne N Den Hartog, Paul L Koopman, and Daan van Knippenberg. The role of transformational leadership in enhancing team reflexivity. *Human Relations*, 61(11):1593–1616, 2008.
- [36] Ruth Schulz, Philipp Kratzer, and Marc Toussaint. Preferred interaction styles for human-robot collaboration vary over tasks with different action types. *Frontiers in Neurobotics*, 12:36, 2018.
- [37] Sina Solaimanpour and Prashant Doshi. A layered HMM for predicting motion of a leader in multi-robot settings. In *International Conference on Robotics and Automation*, 2017.

- [38] Peter Stone, Gal A Kaminka, Sarit Kraus, Jeffrey S Rosenschein, and Noa Agmon. Teaching and leading an ad hoc teammate: collaboration without pre-coordination. *Artificial Intelligence*, 203:35–65, 2013.
- [39] Ethan Stump, Nathan Michael, Vijay Kumar, and Volkan Isler. Visibility-based deployment of robot formations for communication maintenance. In *International Conference on Robotics and Automation*, 2011.
- [40] Fay Sudweeks and Simeon J Simoff. Leading conversations: communication behaviours of emergent leaders in virtual teams. In *Hawaii International Conference on System Sciences*, 2005.
- [41] Anastasiia Varava, Kaiyu Hang, Danica Kragic, and Florian T Pokorny. Herding by caging: a topological approach towards guiding moving agents via mobile robots. In *Robotics: Science and Systems*, 2017.
- [42] Bruno Luis Mendívez Vásquez and Jan carlo Barca. Network topology inference in swarm robotics. In *International Conference on Robotics and Automation*, 2018.
- [43] Phillip Walker, Saman Amirpour Amraii, Michael Lewis, Nilanjan Chakraborty, and Katia Sycara. Human control of leader-based swarms. In *International Conference on Systems, Man, and Cybernetics*, 2013.
- [44] Fereshta Yazdani, Gayane Kazhoyan, Asil Kaan Bozcuoğlu, Andrei Haidu, Ferenc Bálint-Benczédi, Daniel Beßler, Mihai Pomarlan, and Michael Beetz. Cognition-enabled framework for mixed human-robot rescue teams. In *International Conference on Intelligent Robots and Systems*, 2018.