Abstract—Multi-agent system protocols, such as consensus and formation control, require agents to converge rapidly to their desired states. Smooth convergence is achieved through selection of leader nodes (in leader-follower systems) or choosing optimal link weights (in leader-less systems). In this paper, we introduce a framework for minimizing convergence time through optimization of joint leader and link weight selection for a class of systems that include leader-follower and leader-less systems as special cases. We present a semidefinite programming approach for two problems: first, the problem of selecting a fixed number of leaders and the corresponding link weights in order to minimize the convergence time, and second, the problem of selecting the minimum number of leaders and corresponding link weights to achieve a bound on the convergence time. We prove that our approach is equivalent to classical results in discrete location theory, with provable optimality gaps, under special cases. Our results are illustrated through numerical simulations.

I. INTRODUCTION

Multi-agent systems (MAS) consist of distributed, autonomous nodes that coordinate to execute a common task. MAS can be classified as leader-follower systems or leader-less systems. In a leader-follower system, a set of leader nodes receive their states according to an autonomous process and influence the remaining follower nodes. In a leader-less system, each node exchanges information with its neighbors and updates its state accordingly. In both classes of system, the design goal is for the state of each node to converge to its desired value in minimum time and with minimal errors in the intermediate states.

Distributed linear algorithms play a pivotal role in applications of both types of networks, including formation maneuvers, localization and time synchronization, and data aggregation in sensor networks. In such algorithms, each node's state, which may consist of velocity, estimated position, or a combination of multiple parameters, is updated at each iteration to a weighted linear combination of the states of the neighboring nodes. Linear update algorithms are desirable due to their low computation time using only local information, provable convergence properties, and robustness to topology changes and node failures. While algorithms have been proposed that guarantee convergence to the desired state, the methodologies for minimizing the convergence time differ greatly between leader-less and leader-follower systems.

In a leader-less system, the convergence time is minimized by selecting the weight on each neighbor's state. In [1], the authors derive an eigenvalue upper bound on the convergence time, and develop a distributed convex optimization approach to minimizing the upper bound through weight selection.

Generalizations have been proposed to dynamic networks [2] and networks experiencing random failures [3].

Minimizing the convergence time in leader-follower systems is mainly achieved through selecting appropriate leader nodes, as the choice of leader nodes determines how rapidly information will propagate through the network. In [4], it was proved that the convergence error, defined as the deviation of the follower node states from their desired values at a given time, is a supermodular function of the leader set, leading to efficient solution algorithms with provable optimality bounds. In [5], the time required for convergence was studied by proving its equivalence to the mixing time of a Markov chain, which was approximately solved using a subgradient descent algorithm.

At present, however, there are no algorithms for optimal link weight selection in leader-follower systems. Moreover, there is no common framework for weight and leader selection to minimize convergence time that encompasses both leader-follower and leader-less systems.

In this paper, we propose a unifying framework for minimizing convergence time in MAS through a combination of choosing the optimal link weights and selecting leader nodes. Our framework reduces to the randomized gossip algorithm of [1] and the problem of leader selection to minimize convergence error [5] as special cases. We make the following specific contributions:

• We introduce a model for MAS that generalizes both leader-follower and leader-less systems. Under the model, each leader node maintains its current state at each iteration with probability \( \alpha \) and accepts the inputs from follower nodes and updates its state with probability \((1 - \alpha)\). We derive a closed-form solution for the steady-state of the system and prove asymptotic convergence to this value.

• We derive an eigenvalue upper bound on the convergence time and prove that it is a convex function of the weights used by each node. Based on this bound, we formulate the joint leader and weight selection problem as a semidefinite program, and present solution algorithms.

• For the special case in which leader nodes do not accept inputs from the followers (\( \alpha = 1 \)), corresponding to the traditional model of a leader-follower system, we derive the optimal weight selection for a given leader set and present efficient algorithms, based on shortest path computation, for joint leader and weight selection.

• We illustrate our results via a simulation study, and compare with algorithms based on individual optimization of weight selection or leader selection.

The paper is organized as follows. Section II reviews related work. Section III presents the system model and defines

1Corresponding author
the convergence time metric. The problem formulation and solution algorithms are contained in Section IV. Our results are illustrated through simulation study in Section V. Section VI concludes the paper.

II. RELATED WORK

The convergence rate of distributed linear protocols has been extensively studied. In [6], bounds on the convergence rate of linear consensus protocols are derived. In [7], the analysis of [6] was extended to networks experiencing stochastic delays. Convergence in the presence of link failures was studied in [8]. While these works quantify the speed of convergence of consensus protocols under different types of network, they do not consider the impact of leader nodes or include algorithms for optimal weight selection.

Controllability and convergence time of leader-follower systems was considered in [9]. While the authors proved bounds on the convergence time for a given set of leaders and given system dynamics, they did not address the problem of how to choose leaders to minimize the convergence time. In [4] and [5], efficient algorithms for minimizing convergence error and convergence time, respectively, were introduced. The problem of selecting optimal link weights for leader-follower systems, however, was not considered.

Our generalized model of the state dynamics of leader and follower nodes, in which leader nodes maintain their state values with probability $\alpha$, is similar to the model of influential nodes presented in [10]. While the work in [10] includes asymptotic bounds on the stationary distribution of such networks with given dynamics and leader nodes, it does not describe how leader nodes or weights should be selected for optimal convergence.

III. MODEL AND PRELIMINARIES

In this section, we define the model for the MAS, including the graph model and system dynamics. We then prove properties of the system dynamics, derive the steady-state behavior, and define the convergence time metric.

A. System Model

We consider a network of $n$ nodes, indexed in $V = \{1, ..., n\}$. Each node $i$ forms a communication link with any node $j$ within communication range. A link between $i$ and $j$ is denoted $(i, j)$, while the set $E$ denotes the set of links. The neighbor set $N(i)$ denotes the set of nodes within communication range of node $i$. It is assumed that all links are bidirectional, so that $(i, j) \in E$ implies $(j, i) \in E$. We assume that the network topology $G = (V, E)$ is fixed. Each node $i$ has an internal state $x_i(t) \in \mathbb{R}$. We consider a network with two classes of nodes, leaders and followers. Let $S$ denote the set of leader nodes.

At each discrete time point $t$, follower node $i$ updates its state according to the dynamics

$$x_i[t + 1] = \sum_{j \in N(i) \cup \{i\}} \Psi_{ij} x_j[t],$$  \hspace{1cm} (1)$$

where $\Psi$ is a symmetric stochastic matrix with $\Psi_{ij} = 0$ for $(i, j) \notin E$. If $i$ is a leader node, then it maintains its state at each time point with probability $\alpha$ and follows the protocol (1) with probability $(1 - \alpha)$, resulting in dynamics

$$x_i[t + 1] = \left\{ \begin{array}{ll} x_i[t], & \text{w.p. } \alpha \\
\sum_{j \in N(i) \cup \{i\}} \Psi_{ij} x_j[t], & \text{w.p. } (1 - \alpha) \end{array} \right.$$  \hspace{1cm} (2)$$

It is assumed that the probability of a leader node maintaining its state at each time step is independent of previous steps and of the states of other leaders and followers. Let $Y \in \mathbb{R}^{n \times n}$ be a 0-1 diagonal matrix with $Y_{ii} = 1$ if $i \in S$ and 0 otherwise. The expected value of the overall network dynamics can be written in matrix form as

$$E(x[t + 1]) = (\Psi + \alpha Y X(\Psi))x[t],$$  \hspace{1cm} (3)$$

where $X : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^{n \times n}$ is a matrix function defined by

$$[X(\Psi)]_{ij} = \left\{ \begin{array}{ll} -\Psi_{ij}, & i \neq j \\
1 - \Psi_{ii}, & i = j \end{array} \right.$$$$

and, for any matrix $A$, $[A]_{ij}$ denotes the $(i, j)$-entry of $A$. Since $\Psi$ is stochastic, $[X(\Psi)]_{ii} = \sum_{j \neq i} \Psi_{ij}$.

B. Steady-state Behavior and Convergence Metric

The steady-state behavior is defined as follows. Proofs for Propositions 1 and 2 and Lemma 1 are given in the appendix. The asymptotic value of the nodes states with dynamics in (3) is described by the following proposition.

Proposition 1: If $\Psi$ is symmetric, then the node dynamics (3) converge in expectation to $1 \pi^T x[0]$, where $\pi \in \mathbb{R}^n$ is defined by

$$\pi_i = \left\{ \begin{array}{ll} \frac{(1 - \alpha)}{\kappa + (1 - \alpha)(\alpha - \kappa)}, & i \notin S \\
1, & i \in S \end{array} \right.$$  \hspace{1cm} (4)$$

The following preliminary lemma is needed to define the convergence time metric.

Lemma 1: The matrix $(\Psi + \alpha X(\Psi))$ has $n$ real eigenvalues. The error at time $t$ is defined by $||x[t] - x^*||_2$. An upper bound on the convergence error is given as follows.

Proposition 2: Let $\sigma_{max}(\cdot)$ denote the largest singular value of a matrix. The deviation between the node states and their desired values at time $t$ satisfies

$$||x[t] - x^*||_2^2 \leq \sigma_{max}(\Psi + \alpha X(\Psi) - 1 \pi^T)^2 t ||x_0||_2^2.$$$$

We now define the convergence time metric.

Definition 1 (Convergence time): Let $Y$ be a diagonal matrix representing the leader set, and let $\Psi$ denote the matrix of link weights. Then the convergence time is defined as $\sigma_{max}(\Psi + \alpha X(\Psi) - 1 \pi^T)$.

Since $\pi$ is the stationary distribution only when $\Psi$ is symmetric, the convergence time metric implicitly assumes that $\Psi$ is symmetric.
IV. Problem Formulation - Joint Leader and Weight Selection

In this section, the problem formulation for selecting leaders and weights in order to minimize convergence time is presented. First, the problem of selecting the optimal set of link weights in order to minimize the convergence time for a given leader set is formulated. We then extend the approach to joint selection of link weights and leader nodes in order to minimize convergence time.

We consider two specific problems of this type. The first problem consists of selecting a set of up to \( k \) leaders and the link weights in order to minimize convergence time. The second problem consists of selecting the minimum number of leader nodes to achieve a given bound on the convergence time. The case of \( \alpha = 1 \), where leaders maintain a constant state, is treated as a special case.

A. Weight Selection

We first treat the case where the leader set \( S \) is given. The leader set is represented by the diagonal matrix \( Y \in \mathbb{R}^{n \times n} \), defined by \( Y_{ii} = 1 \) if \( i \in S \) and \( Y_{ii} = 0 \) otherwise. The optimization problem of selecting the weight matrix \( \Psi \) in order to minimize the convergence time is given as

\[
\begin{align*}
\min_{\Psi} & \quad \sigma_{\text{max}}(\Psi + \alpha Y X(\Psi) - 1\pi^T) \\
\text{s.t.} & \quad \Psi 1 = 1, \quad \Psi_{ij} = 0 \quad \forall (i,j) \notin E
\end{align*}
\]

(5)

Here \( \mathbb{S}_+^n \) denotes a positive semidefinite matrix. The condition \( \Psi \in \mathbb{S}_+^n \) ensures that the stationary distribution is \( \pi \) and hence that the objective function of (5) is well-defined. The following proposition leads to efficient algorithms for solving (5).

Proposition 3: The convergence time \( \sigma_{\text{max}}(\Psi + \alpha Y X(\Psi)) \) is a convex function of \( \Psi \).

Proof: The function \( \sigma_{\text{max}}(\Psi + \alpha Y X(\Psi)) \) can be restated as

\[
\sigma_{\text{max}}(\Psi + \alpha Y X(\Psi) - 1\pi^T) = \max_{\|v\|_2=1} \left\{ v^T (\Psi + \alpha Y X(\Psi) - 1\pi^T) v \right\}.
\]

(6)

We now show that \( v^T (\Psi + \alpha Y X(\Psi) - 1\pi^T) T (\Psi + \alpha Y X(\Psi) - 1\pi^T) v \) is convex as a function of \( \Psi \), which implies that \( \sigma_{\text{max}}(\Psi + \alpha Y X(\Psi) - 1\pi^T) \) is a pointwise maximum of convex functions and is therefore convex.

First, the function \( f(R) := R^T R \) is a convex function of \( R \in \mathbb{R}^{n \times n} \). Indeed, the Schur complement theorem implies that

\[
\{(R,t) : v^T R^T R v \leq t\} \Leftrightarrow \left\{ (A,t) : \left( \begin{array}{cc} I & A v \\ v^T A^T & t \end{array} \right) \geq 0 \right\}.
\]

(7)

Since the right hand side of (7) is convex, convexity of \( f(R) \) follows from the epigraph condition [11, Ch 3]. We then show that \( \Psi + Y X(\Psi) - 1\pi^T \) is an affine function of \( \Psi \). By definition, for any \( \beta, \gamma \) and \( \Psi, \Psi' \in \mathbb{S}_+^n \), and \( i \neq j \),

\[
[X(\beta \Psi + \gamma \Psi')]_{ij} = -\beta \Psi_{ij} - \gamma \Psi'_{ij} = \beta X(\Psi) + \gamma X(\Psi').
\]

Similarly, when \( i = j \),

\[
[X(\beta \Psi + \gamma \Psi')]_{ii} = -\beta \Psi_{ii} - \gamma \Psi'_{ii} = \beta X(\Psi) + \gamma X(\Psi').
\]

As a result,

\[
\beta \Psi + \gamma \Psi' + \alpha Y X(\beta \Psi + \gamma \Psi') = \beta (\Psi + \alpha Y X(\Psi)) + \gamma (\Psi' + \alpha Y X(\Psi')) \tag{8}
\]

is a convex function of \( \Psi \) and \( \Psi' \), as desired. The function \( v^T (\Psi + \alpha Y X(\Psi) - 1\pi^T) T (\Psi + \alpha Y X(\Psi) - 1\pi^T) v \) is therefore a composition of a convex function and an affine function, and hence is convex as a function of \( \Psi \).

Note that when \( \alpha = 0 \) or \( k = 0 \), (5) reduces to the weight selection formulation of [1]. It remains to find an efficient algorithm for solving (5). Such an algorithm is derived from the following theorem.

Theorem 1: The optimization problem (5) is equivalent to the semi-definite program

\[
\begin{align*}
\min_{s, \Psi} & \quad s \\
\text{s.t.} & \quad \left( \begin{array}{cc} I & \Omega \\ \Omega^T & s^2 I \end{array} \right) \geq 0 \\
& \quad \Psi 1 = 1, \quad \Psi_{ij} = 0 \quad \forall (i,j) \notin E \\
& \quad \Omega = \Psi - \alpha X(\Psi) - 1\pi^T
\end{align*}
\]

(9)

Proof: The problem (5) can be rewritten using the characterization of \( \sigma_{\text{max}}(\Psi - \alpha X(\Psi) - 1\pi^T) \) given by (6). Hence the constraint \( \sigma_{\text{max}}(\Psi - \alpha X(\Psi)) \leq s \) is equivalent to

\[
\Psi - \alpha X(\Psi) - 1\pi^T \preceq s I. \tag{10}
\]

Eq. (10) can be rewritten using the Schur complement as

\[
\left( \begin{array}{cc} I & \Psi - \alpha X(\Psi) - 1\pi^T \\ (\Psi - \alpha X(\Psi) - 1\pi^T)^T & s^2 I \end{array} \right) \geq 0. \tag{11}
\]

Using Eq. (11) and introducing \( s \) as a slack variable yields the formulation (9).

B. Joint Leader and Weight Selection with Fixed Number of Leaders

We next consider joint leader and weight selection, starting with the problem of jointly selecting up to \( k \) leader nodes and link weights in order to minimize the convergence time. The leader set is represented by the diagonal matrix \( Y \), which is now an optimization variable in the problem, formulated as

\[
\begin{align*}
\min_{\Psi, Y} & \quad \sigma_{\text{max}}(\Psi + \alpha Y X(\Psi) - 1\pi^T) \\
\text{s.t.} & \quad \Psi 1 = 1, \quad \Psi_{ij} = 0 \quad \forall (i,j) \notin E \\
& \quad Y \text{ diagonal, } \text{tr}(Y) \leq k, \quad Y_{ii} \in \{0,1\} \forall i
\end{align*}
\]

(12)

The constraint \( Y_{ii} \in \{0,1\} \) enforces the fact that being a leader node is a discrete, Boolean variable, while \( \text{tr}(Y) \leq k \) ensures that the number of leaders does not exceed \( k \). The constraints on \( \Psi \) remain from (5).
\textbf{k-leader-weights:} Algorithm for joint selection of up to \( k \) leaders and corresponding link weights.

**Input:** Graph \( G = (V, E) \), maximum number of leaders \( k \)

**Leader parameter \( \alpha \)**

**Output:** Leader set \( S \), weight matrix \( \Psi \)

**Initialization:** \( \Psi_0 \leftarrow \frac{1}{n}11^T \), \( Y_0 \leftarrow 0 \), \( S_0 \leftarrow \emptyset \), \( l \leftarrow 0 \)

\[
\text{while } l < k \\
\quad l \leftarrow l + 1 \\
\quad (\Psi_l, Y_l) \leftarrow \arg \min \left\{ \sigma_{\max}(\Psi + \alpha Y X(\Psi) - 1\pi^T) : \Psi \in S^n_+ \right\} \\
\quad v_i^* \leftarrow \max_{i \in S_{l-1}} \{[Y_l]_{ii}\} \\
\quad S_l \leftarrow S_{l-1} \cup \{v_i^*\} \\
\quad \text{for } i \in V \\
\quad \quad \text{if } i \in S_l \text{, } [Y_l]_{ii} \leftarrow 1, \text{ else } [Y_l]_{ii} \leftarrow 0 \\
\quad \text{end for} \\
\text{end while} \\
\text{return } S, \Psi
\]

The condition \( Y_{ii} \in \{0, 1\} \) makes (12) a mixed integer program, which is NP-hard to solve in general. As a result, the condition is relaxed to \( Y_{ii} \in [0, 1] \), and rounding techniques can be used to derive a feasible \( Y \). Furthermore, since the product \( YX(\Psi) \) is not jointly convex as a function of \( Y \) and \( X(\Psi) \), (12) is not convex.

To approximate the solution to (12), a sequence of \( k \) semidefinite programs is solved, with each semidefinite program used to select one leader node and the corresponding optimal set of link weights.

The algorithm is as follows. First, initialize \( Y_0 = 0 \), \( X_0 = \frac{1}{n}11^T \) (or any symmetric stochastic matrix). At stage \( r \), let \( S_r \) denote the set of nodes that have been designated as leaders (\( Y_{ii} = 1 \) for all \( i \in S_r \)), and find the node \( v_r \) such that

\[
v_r = \arg \min \left\{ \sigma_{\max}(\Psi_r + \alpha Y_{r-1} X(\Psi_{r-1}) - 1\pi^T) : v \in V \setminus S_r \right\}.
\]

The node \( v_r \) is then added to \( S_r \), resulting in \( S_{r+1} = S_r \cup \{v_r\} \).

The following optimization problem is then solved to obtain the optimal set of link weights corresponding to \( S_r \).

\[
\text{minimize } \sigma_{\max}(\Psi + \alpha Y_{r+1} X(\Psi) - 1\pi^T) \\
\text{s.t. } \Psi 1 = 1, \Psi_{ij} = 0, (i, j) \notin E
\]

Once \( |S| = k \), an optimization problem of the form (5) is solved to obtain the optimal weight matrix. Under this approach, one leader is selected per iteration, while the weight matrix \( \Psi \) is optimized simultaneously. A pseudocode description of the algorithm is given as \textbf{k-leader-weights}.

**Leader-weights-c-bound:** Algorithm for joint selection of leaders and corresponding link weights to achieve selection \( \alpha \) on convergence time.

**Input:** Graph \( G = (V, E) \), convergence time bound \( \epsilon \)

**Leader parameter \( \alpha \)**

**Output:** Leader set \( S \), weight matrix \( \Psi \)

**Initialization:** \( \Psi_0 \leftarrow \frac{1}{n}11^T \), \( Y_0 \leftarrow 0 \), \( S_0 \leftarrow \emptyset \), \( l \leftarrow 0 \)

\[
\text{while } \sigma_{\max}(\Psi_l + \alpha Y_l X(\Psi_l) - 1\pi^T) > \epsilon \\
\quad l \leftarrow l + 1 \\
\quad (\Psi_l, Y_l) \leftarrow \arg \min \left\{ \sigma_{\max}(\Psi + \alpha Y X(\Psi) - 1\pi^T) : \Psi \in S^n_+ \right\} \\
\quad v_i^* \leftarrow \max_{i \in S_{l-1}} \{[Y_l]_{ii}\} \\
\quad S_l \leftarrow S_{l-1} \cup \{v_i^*\} \\
\quad \text{for } i \in V \\
\quad \quad \text{if } i \in S_l \text{, } [Y_l]_{ii} \leftarrow 1, \text{ else } [Y_l]_{ii} \leftarrow 0 \\
\quad \text{end for} \\
\text{end while} \\
\text{return } S, \Psi
\]

The condition \( Y_{ii} \in \{0, 1\} \) makes (12) a mixed integer program, which is NP-hard to solve in general. As a result, the condition is relaxed to \( Y_{ii} \in [0, 1] \), and rounding techniques can be used to derive a feasible \( Y \). Furthermore, since the product \( YX(\Psi) \) is not jointly convex as a function of \( Y \) and \( X(\Psi) \), (12) is not convex.

To approximate the solution to (12), a sequence of \( k \) semidefinite programs is solved, with each semidefinite program used to select one leader node and the corresponding optimal set of link weights.

The algorithm is as follows. First, initialize \( Y_0 = 0 \), \( X_0 = \frac{1}{n}11^T \) (or any symmetric stochastic matrix). At stage \( r \), let \( S_r \) denote the set of nodes that have been designated as leaders (\( Y_{ii} = 1 \) for all \( i \in S_r \)), and find the node \( v_r \) such that

\[
v_r = \arg \min \left\{ \sigma_{\max}(\Psi_r + \alpha Y_{r-1} X(\Psi_{r-1}) - 1\pi^T) : v \in V \setminus S_r \right\}.
\]

The node \( v_r \) is then added to \( S_r \), resulting in \( S_{r+1} = S_r \cup \{v_r\} \).

The following optimization problem is then solved to obtain the optimal set of link weights corresponding to \( S_r \).

\[
\text{minimize } \sigma_{\max}(\Psi + \alpha Y_{r+1} X(\Psi) - 1\pi^T) \\
\text{s.t. } \Psi 1 = 1, \Psi_{ij} = 0, (i, j) \notin E
\]

Once \( |S| = k \), an optimization problem of the form (5) is solved to obtain the optimal weight matrix. Under this approach, one leader is selected per iteration, while the weight matrix \( \Psi \) is optimized simultaneously. A pseudocode description of the algorithm is given as \textbf{k-leader-weights}.

**C. Joint Leader and Weight Selection - Achieving Convergence Time Bound**

The optimization problem of selecting the minimum size set of leaders in order to satisfy a given bound \( \epsilon \) on the convergence time is given as

\[
\text{minimize } \text{tr}(Y) \\
\Psi, Y \in S^n_+ \\
\text{s.t. } \sigma_{\max}(\Psi + \alpha Y X(\Psi)) \leq \epsilon \\
\Psi 1 = 1, \Psi_{ij} = 0, (i, j) \notin E \\
Y \text{ diagonal, } Y_{ii} = 1 \forall i \in S_r, \text{ tr}(Y) \leq r
\]

(14)

Minimizing the number of leaders is equivalent to minimizing the trace of \( Y \), since \( Y_{ii} = 1 \) iff \( i \in S \) and \( Y_{ii} = 0 \) otherwise. As in problem (12), a continuous relaxation of the constraint \( Y_{ii} \in \{0, 1\} \) is used, and the next leader node is selected at each iteration based on the solution to a semidefinite program of the form (13). To solve (14), initialize \( Y_0 = 0 \) and \( \Psi_0 = \frac{1}{n}11^T \). At iteration \( r \), let \( S_r \) denote the leader set and solve the optimization problem (13). The algorithm terminates when \( \sigma_{\max}(\Psi_r + Y_r X(\Psi_r) - 1\pi^T) \leq \epsilon \). A description of this algorithm is given as \textbf{Leader-weights-c-bound}.

Algorithm \textbf{Leader-weights-c-bound} requires solving \((k + 2)\) semidefinite programs, each with runtime \( O(n^6) \), resulting in overall runtime of \( O(kn^6) \).

\textbf{D. Special Case:} \( \alpha = 1 \)

In the case where \( \alpha = 1 \), implying that the leader nodes maintain a constant state, the follower node states will converge to the state of the leader nodes. The following proposition gives a bound on the minimum possible convergence time.

**Proposition 4:** Let \( d_i(S) \triangleq \min_{j \in S} d_{ij} \), where \( d_{ij} \) is the shortest path length between nodes \( i \) and \( j \). Then

\[
\min_t \left\{ \max_{x(0)} \left\{ x_i(t) \right\} = x^* \right\} \geq d_{ij}.
\]

(15)
Proof: By (1), \( x_i(t) = [A^t x(0)]_i \). If \( t < d_i^* \), then \( [A^t]_{ij} = 0 \) for all \( j \in S \), and so \( x_i(t) \) lies in \( \text{conv}\{x_j(0) : j \notin S\} \). If the \( x^* \notin \text{conv}\{x_j(0) : j \notin S\} \), then \( x_i(t) \neq x^* \), giving the desired result.

The following procedure gives a set of weights that, for a given leader set, achieves the bound in Proposition 4. First, define \( V_s := \{i \in V : d_i^* = s\} \). At stage \( s \), for each node \( i \in V_s \), choose a neighbor \( j \in N(i) \) that is on a shortest path from \( i \) to the closest node in \( S \), and set \( A_{ij} = 1 \). The remaining entries of row \( i \) of \( A \) are set to 0.

**Proposition 5:** The matrix \( A \) selected by the above procedure achieves the lower bound in Proposition 4.

**Proof:** From the above analysis, \( [A^t]_{ij} = 1 \), so that \( x_i[t] = x^* \) for all \( t \geq d_i^* \).

Proposition 5 implies that the problem of selecting an optimal set of up to \( k \) leaders in order to minimize the convergence time is equivalent to

\[
\begin{align*}
\text{minimize} & \quad \max_i d_i^*(S) \\
\text{s.t.} & \quad |S| \leq k
\end{align*}
\]

(16)

Problem (16) is the classical \( k \)-center facility location problem from discrete location theory [12]. An algorithm for solving (16) is as follows. An upper and lower bound, \( d_{max} \) and \( d_{min} \), are initialized to \( |E| \) and 1, respectively. At each iteration, set \( d = (d_{max} + d_{min})/2 \). The algorithm then attempts to find a set \( S \) such that all nodes in the network are either contained in \( S \) or are connected to \( S \) via a set of edges \( \{e_1, \ldots, e_d\} \), where \( \{e_1, \ldots, e_{|E|}\} \) is any random ordering of the edges in \( E \). If the returned set \( S \) satisfies \( |S| \leq k \), then set \( d_{max} = d \), and set \( d_{min} = d \) otherwise. The algorithm terminates when \( d_{max} = d_{min} + 1 \). A pseudocode description is given as algorithm **Constant-state-leaders-weights**.

The main theorem of [12] implies that the set \( S \) returned by algorithm **Constant-state-leaders-weights** satisfies

\[
\max_{i \in V \setminus S} \min_{j \in S} d_{ij} \leq \max_{i \in V \setminus S^*} \min_{j \in S^*} d_{ij}
\]

where \( S^* \) is the optimal solution to (16).

V. SIMULATION

Our proposed joint leader and weight selection framework is evaluated through numerical simulation using Matlab. A network of 30 nodes is simulated, with nodes deployed uniformly at random over a square area of width 1800m. Each node has communication range of 300m. Follower nodes have the linear dynamics of (1), while leader nodes have dynamics (2) with varying values of \( \alpha \). The convergence time, as described in Definition 1, is evaluated to quantify the performance of the selected leaders and link weights. Each data point represents an average of 50 trials. Under this network topology, four algorithms are simulated: (a) Random selection of link weights and leader set, (b) Random selection of link weights and selection of leader nodes according to a greedy algorithm similar to [5], (c) Random selection of leader nodes and selection of link weights based on Section IV-A, and (d) Joint selection of leader nodes and link weights using the algorithm **k-leaders-weights**.

The simulations were designed to answer the following questions: (i) How much performance improvement is provided on average by joint selection of leaders and weights in comparison with approaches (a), (b), and (c)? (ii) What is the probability that, for a given network, the joint leader and weight selection will cause a significant improvement in convergence time? and (iii) How does the parameter \( \alpha \), which determines the extent to which the leaders are influenced by the follower nodes, affect the convergence time?

Question (i) is addressed by Figure 1(a), which shows the performance of each algorithm as the number of leader nodes, \( k \), increases. For each method, the convergence time appears roughly linear as a function of the number of leaders, with the joint leader and weight selection providing a 7% improvement in performance compared with leader selection alone, while leader selection on average provides 5% lower convergence time than weight selection alone. Furthermore, while joint leader and weight selection require \( O(kn^6) \) computations, the greedy leader selection algorithm requires only \( O(n^2) \) computations.

In order to address Question (ii), we compared the convergence time of each algorithm as the number of leaders varied (Figure 1(a)). We observed that, in 72% of instances, the joint leader and weight selection approach provided lower convergence time than both leader selection and weight selection individually. We also found that, as the number of leaders increased, the gap between joint leader and weight selection and the other approaches widened. For example, for networks with ten leaders, the joint selection algorithm provided lower convergence time in all trials.
Fig. 1. Simulation illustrating joint leader and weight selection for a network of 30 nodes deployed over a square region with width 1800 meters. Each plot compares weight, leader, and joint leader and weight selection algorithms. (a) Joint leader and weight selection outperforms the individual leader and weight selection algorithms. Leader selection alone provides 5% lower convergence time than weight selection alone. (b) Convergence time as a function of parameter $\alpha$. The joint leader and weight selection algorithm provides lower convergence time than the alternative methods. Convergence time reaches a local minimum at roughly $\alpha = 0.3$ and then increases.

Question (iii) is considered in Figure 1(b), which illustrates the impact of the parameter $\alpha$ on the convergence time. As $\alpha$ increases, the convergence time increases for each algorithm. The joint leader and weight selection algorithm, however, results in 10% lower convergence time than both other algorithms. The convergence time decreases initially as a function of $\alpha$ until roughly $\alpha = 0.3$, and then increases until $\alpha = 1$.

VI. Conclusion

In this paper, we studied the problem of minimizing the convergence time of a multi-agent system through joint selection of link weights and leader nodes. We formulated a model of leader-follower dynamics that incorporated traditional leader-follower and leader-less systems as special cases, analyzed its steady-state behavior, and derived a metric for quantifying the convergence time. As a first step, we formulated the problem of selecting link weights in order to minimize the convergence time when the set of leader nodes is given. We presented a semidefinite programming approach to finding the optimal set of link weights. We then formulated two joint leader and link weight selection problems, namely (1) selecting a set of up to $k$ leader nodes and link weights in order to minimize the convergence time, and (2) selecting the minimum-size set of leader nodes, along with the link weights, in order to achieve a bound on the convergence time. While the problem in its most general form is not convex, we demonstrated semidefinite programming heuristics for solving problems (1) and (2). In the particular case where the leader nodes maintain a constant state, the chosen weights are optimal and the optimality gap of the selected leaders can be characterized through a connection to discrete location theory. Our approach was demonstrated through simulation study, in which we compared our joint leader and link weight selection procedure to algorithms based on optimizing leader selection and link weights independently. Our results indicate that joint selection outperforms both individual schemes, and also suggest that much of the improvement is due to the incorporation of leader selection into the link weight algorithm.

The approaches presented in this paper are inherently centralized. In the future, we plan to study distributed leader and link weight selection, in which nodes select leaders and the weights for their neighbors based on local information exchanged with neighbors.

REFERENCES

This appendix contains proofs of Propositions 1 and 2, as well as Lemma 1. A proof of Proposition 1 is as follows.

**Proof:** The expected value of $x[t]$ is equal to $(\Psi + \alpha X(\Psi))^t x[0]$. Since the graph $G$ is connected and each leader has nonzero probability of adopting its own state at each time step, $\Psi$ is an irreducible stochastic matrix with period 1. Hence, by the Perron-Frobenius Theorem [13],

$$\lim_{t \to \infty} (\Psi + \alpha X(\Psi))^t = \mathbf{v} \mathbf{w}^T,$$

where $\mathbf{v}$ and $\mathbf{w}$ are the left and right eigenvectors of $(\Psi + \alpha X(\Psi))$ corresponding to eigenvalue 1. It therefore suffices to show that $\mathbf{v} = 1$ and $\mathbf{w} = \pi$.

Since $\Psi$ is assumed to be stochastic, $\Psi \mathbf{1} = \mathbf{1}$. It remains to show that $\pi^T \Psi = \pi^T$. For $i \in S$, we have

$$(\pi^T \Psi)_i = \sum_{j=1}^{n} \pi_j (\Psi_{ji} + \alpha X(\Psi)_{ji})$$

$$= \sum_{j \notin S} \pi_j \Psi_{ji} + \sum_{j \notin S} \pi_j \Psi_{ji} (1 - \alpha) + \pi_i ((1 - \alpha)\Psi_{ii} + \alpha)$$

$$= \frac{1}{k + (1 - \alpha)(n - k)}.$$

For $i \notin S$, we have

$$(\pi^T \Psi)_i = \sum_{j=1}^{n} \pi_j (\Psi_{ji} + \alpha X(\Psi)_{ji})$$

$$= \sum_{j \notin S} \pi_j \Psi_{ji} + \sum_{j \in S} \pi_j (1 - \alpha)\Psi_{ji},$$

implying that $\lim_{t \to \infty} (\Psi + \alpha X(\Psi))^t = 1 \pi^T$, as desired.

A proof of Proposition 2 is below.

**Proof:** The definition of the 2-norm implies that

$$|W^T x(0) - 1 \pi^T x(0)|^2 \leq |W^T - 1 \pi^T|^2 |x(0)|^2$$

$$= \sigma_{\max} (W^T - 1 \pi^T)^2 |x(0)|^2$$

$$= \sigma_{\max} (\Psi + \alpha X(\Psi) - 1 \pi^T)^2 |x(0)|^2$$

The following is a proof of Lemma 1.

**Proof:** The first step of the proof is to show that the matrix $B$, defined by $B_{ij} = \pi_i^{1/2} \pi_j^{-1/2} [\Psi + \alpha X(\Psi)]_{ij}$, is symmetric. Suppose that $i \notin S$, $j \notin S$. Then

$$B_{ij} = \pi_i^{1/2} \pi_j^{-1/2} (\Psi_{ij} + \alpha [X(\Psi)]_{ij}) = \Psi_{ij} = \Psi_{ji} = B_{ji}.$$