Leader Selection for Minimizing Convergence Error in Leader-Follower Systems: A Supermodular Optimization Approach

Andrew Clark, Linda Bushnell, and Radha Poovendran
Electrical Engineering Dept., University of Washington, Seattle, WA, 98195, USA
Email: {awclark, lb2, rp3}@uw.edu

***Abstract*—In leader-follower systems, follower nodes receive inputs from a set of leader nodes, exchange information, and update their states according to an iterative algorithm. In such algorithms, the node states may deviate from their desired values before the algorithm converges, leading to disruptions in network performance. In this paper, we study the problem of choosing leader nodes in order to minimize convergence errors. We first develop a connection between a class of weighted averaging algorithms and random walks on graphs, and then show that the convergence error is a supermodular function of the set of leader nodes. Based on the supermodularity of the convergence error, we derive efficient algorithms for selecting leader nodes that are within a provable bound of the optimum. Our approach is demonstrated through a simulation study.**

I. Introduction

A broad class of networks can be classified as leader-follower systems, where leader nodes are controlled directly by the network owner, while the remaining follower nodes update their internal states based on inputs from the leaders. Applications include vehicle formation control, in which unmanned vehicles follow the heading and velocity of a set of leaders [1].

In leader-follower systems, the follower nodes update their states through iterated, distributed algorithms, in which each node receives state information from its neighbors and performs computations to update and broadcast its own state to its neighbors [2]. The goal is to design algorithms such that the node states converge to their desired value within finite time. Moreover, in addition to ensuring that the node states converge to their desired final states, the deviations of the values of the intermediate states from the final values should be as small as possible.

A common approach to updating the follower node states is through weighted averaging of the states of the neighboring nodes [3]. This approach requires minimal computation at each iteration, and can be implemented by each node using only local information. Furthermore, the node states are guaranteed to converge when the underlying network graph is connected, even if the graph topology changes over time [2]. In [3], the authors further show that the convergence errors, defined as the ℓ^2-norm of the difference between the intermediate node states and their desired values, can be reduced by choosing the weights in order to maximize the algebraic connectivity of the network.

We note that the convergence errors of a leader-follower system are also affected by which agents are chosen as leaders; at present, however, there is no analytic framework for selecting leaders under this criterion in the existing literature. In this paper, we study the problem of selecting leader nodes in order to minimize convergence errors.

In doing so, we first define a family of metrics for convergence error, and derive bounds that are independent of the initial state of the network. Using a novel connection to random walks on graphs, we then show that this bound is a supermodular function of the set of leader nodes. We formulate two leader selection problems using a supermodular optimization framework. First, we study the problem of selecting up to k leaders in order to minimize the convergence error. Second, we study the problem of selecting the minimum number of leaders in order to meet a given bound on the convergence error. For both problems, we propose leader selection algorithms that are within a provable bound of the optimum. We illustrate our approach through a simulation study and compare it with other leader selection algorithms.

The paper is organized as follows. In Section II, the system model and background information are given. Section III introduces the problem formulation. Section IV presents the simulation study. Section V concludes the paper.

II. System Model and Background

In this section, the network model is presented, along with background on supermodular functions.

A. Network Model

We assume a network of n nodes, indexed $V = \{1, \ldots, n\}$. A link (i,j) between two nodes exists if node i is within communication range of node j. Let E denote the set of links, and let $N(i) \triangleq \{j : (i,j) \in E\}$ denote the set of neighbors of i. The degree of i is equal to $|N(i)|$. Let $G = (V, E)$ denote the resulting directed network graph.

Each node $i \in V$ has a time-varying internal state $x_i(t) \in \mathbf{R}$. A subset of *leader* nodes $S \subseteq V$ receive inputs directly from the network owner. The remaining follower nodes update their states according to

$$\dot{x}_i(t) = \sum_{j \in N(i)} W_{ij}(x_j(t) - x_i(t)) \qquad (1)$$

where W_{ij} are nonnegative update coefficients.

The overall system dynamics can then be described by introducing the Laplacian matrix L, defined by

$$L_{ij} = \begin{cases} -W_{ij}, & (i,j) \in E,\ i \in V \setminus S \\ \sum_{(i,j) \in E} W_{ij}, & i = j,\ i \in V \setminus S \\ 0, & \text{else} \end{cases} \qquad (2)$$

Letting $\mathbf{x} \in \mathbb{R}^n$ denote the vector of node states, the system dynamics are given in vector form by $\dot{\mathbf{x}}(t) = -L\mathbf{x}(t)$, so that $\mathbf{x}(t) = e^{-Lt}\mathbf{x}(0)$. Note that e^{-Lt} is a stochastic matrix for all t [2] (note that L is not stochastic in general, since the rows of L sum to 1). The following theorem establishes the asymptotic behavior of a network with dynamics given by (1).

Theorem 1: Let $\mathbf{x}^* \in \mathbb{R}^n$ be the target state of the network, with $\mathbf{x}^* = \mathbf{x}_{ref} + z_0\mathbf{1}$, where $\mathbf{x}_{ref}$ is a known reference point, $\mathbf{1}$ is the vector of all ones, and z_0 is unknown to the follower nodes. Suppose the underlying graph $G = (V, E)$ is strongly connected and that $x_j(t) \equiv \mathbf{x}_j^*$ (the j-th component of $\mathbf{x}^*$) for all $j \in S$ and $t \geq 0$. Then for all $i \in V$, $\lim_{t\rightarrow\infty} x_i(t) = \mathbf{x}_i^*$.

A proof of Theorem 1 can be found in [2]. While the system converges asymptotically to the state of the leaders, at each finite time t each node deviates from the leader state. A family of metrics, based on the ℓ^p vector norms, quantifying these convergence errors is given by the following definition.

Definition 1: For $1 \leq p < \infty$, the p-convergence error at time t when the leader set is S is defined as

$$\epsilon_t^p(S) \triangleq ||\mathbf{x}(t) - \mathbf{x}^*||_p \tag{3}$$

where $||y||_p = \left(\sum_{i \in V} |y_i|^p\right)^{1/p}$.

In what follows, we assume without loss of generality that $\mathbf{x}^* = z_0\mathbf{1}$; otherwise the subsequent analysis holds with $\hat{\mathbf{x}}(t) = \mathbf{x}(t) - \mathbf{x}_{ref}$.

B. Background – Supermodular Functions

Let V be a finite set, and let $f : 2^V \rightarrow \mathbf{R}_{\geq 0}$ be a function mapping subsets of V to real numbers. The following condition defines a supermodular function.

Definition 2: Let $A \subseteq B \subseteq V$, and let $v \in V \setminus B$. Then f is supermodular if and only if

$$f(A) - f(A \cup \{v\}) \geq f(B) - f(B \cup \{v\}) \tag{4}$$

It can be shown that nonnegative weighted sums of supermodular functions are supermodular. Moreover, the following result, which to the best of our knowledge does not appear in the existing literature, provides a composition rule for supermodular functions. The proof can be found in the appendix.

Lemma 1: Let $f : 2^V \rightarrow \mathbf{R}_{\geq 0}$ be a nonincreasing supermodular function, and suppose that for any $S \subseteq T$, $f(S) \geq f(T)$. Then for any nondecreasing, convex, and differentiable function $g : \mathbf{R}_{\geq 0} \rightarrow \mathbf{R}$, $h = g \circ f$ is a supermodular function.

III. Proposed Leader Node Selection Approach

In formulating the problem of leader selection to minimize convergence error, we observe that the function $\epsilon_t^p(S)$ depends on the initial state of the network $\mathbf{x}(0)$, which may not be known when the leaders are selected. We therefore introduce a metric $\tilde{\epsilon}_t^p(S)$, based on an augmented graph $\tilde{G}$, which we prove to be equivalent to $\epsilon_t^p(S)$. We then derive an upper bound on $\tilde{\epsilon}_t^p(S)$, denoted $\hat{\epsilon}_t^p(S)$, that is independent of $\mathbf{x}(0)$.

Using the objective function $\hat{\epsilon}_t^p(S)$, we address two leader selection problems. First, given a fixed number k, we study the problem of selecting up to k leaders in order to minimize the convergence error. Second, we study the problem of choosing the minimum number of leaders to achieve a given bound α on the convergence error.

A. Construction of Augmented Graph

We augment the graph G as follows. First, we expand the graph $G = (V, E)$ by adding a node with index 0. An augmented graph $\tilde{G} = (\tilde{V}, \tilde{E})$ is then obtained by setting $\tilde{V} = V \cup \{0\}$, removing all edges (i, j) with $i \in S$ and $j \in V \setminus S$, and adding an edge $(j, 0)$ for each $j \in S$. Each node $j \in S$ then updates its state according to $\dot{x}_j(t) = W_{0j}(x_0(t) - x_j(t))$. The state of node 0 is constant, with $x_0(t) \equiv x^*$ for all t. Define $\tilde{L}$ to be analogous to the matrix L of (2) for the augmented graph $\tilde{G}$.

Lemma 2: Let $\tilde{\epsilon}_t^p(S)$ denote the convergence error of the augmented network $\tilde{G}$. Then $\tilde{\epsilon}_t^p(S) = \epsilon_t^p(S)$ for any $t \geq 0$.

Proof: By assumption, $x_j(0) = x^*$ for all $j \in S$, so that $\dot{x}_j(t) = W_{0j}(x_0(t) - x_j(t)) = W_{0j}(x^* - x^*) = 0$ for all t. Hence the dynamics of $x_j(t)$ are unchanged for all $j \in S$. Furthermore, since $W_{0i} = 0$ for all $i \in V \setminus S$, the dynamics of the remaining nodes are unchanged as well. In particular, $\tilde{\epsilon}_t^p(S) = \epsilon_t^p(S)$ for all t. ■

As a result of Lemma 2, minimizing the convergence error $\epsilon_t^p(S)$ is equivalent to minimizing $\tilde{\epsilon}_t^p(S)$. We now give a bound on $\tilde{\epsilon}_t^p(S)$, which leads to an objective function that is independent of the initial state $\mathbf{x}(0)$.

Theorem 2: Let q satisfy $\frac{1}{p} + \frac{1}{q} = 1$, and suppose $||\mathbf{x}(0)||_q \leq K$. Define $P_t = e^{-\tilde{L}t}$, and let e_i denote the vector with a 1 in the i-th entry and 0s elsewhere. Then for any S, $\tilde{\epsilon}_t^p(S)$ satisfies

$$\tilde{\epsilon}_t^p(S) \leq K \left(\sum_{i \in V \setminus S} ||P_t^T e_i - e_0||_p^p \right)^{1/p} \tag{5}$$

where P_t^T indicates the matrix transpose of P_t.

Proof: We write $P = P_t$. We have

$$\begin{aligned}
\tilde{\epsilon}_t^p(S) &= ||\mathbf{x}(t) - \mathbf{x}^*||_p && (6)\\
&= \left(\sum_{i \in V \setminus S} |x_i(t) - \mathbf{x}_i^*|^p \right)^{1/p} && (7)\\
&= \left(\sum_{i \in V \setminus S} |e_i^T P\mathbf{x}(0) - e_0^T\mathbf{x}(0)|^p \right)^{1/p} && (8)\\
&= \left(\sum_{i \in V \setminus S} |(P^T e_i - e_0)^T\mathbf{x}(0)|^p \right)^{1/p} && (9)\\
&\leq \left(\sum_{i \in V \setminus S} \left(||P^T e_i - e_0||_p ||\mathbf{x}(0)||_q\right)^p \right)^{1/p} && (10)\\
&\leq K \left(\sum_{i \in V \setminus S} ||P^T e_i - e_0||_p^p \right)^{1/p} && (11)
\end{aligned}$$

where (8) follows from the fact that $\mathbf{x}(t) = e^{-Lt}\mathbf{x}(0)$, (10) follows from Hölder's inequality and the triangle inequality, and (11) follows by rearranging terms and the fact that $||\mathbf{x}(0)||_q \leq K$. ■

Let $\hat{\epsilon}_t^p(S) = \sum_{i \in V \setminus S} ||P^T e_i - e_0||_p^p$, and note that $\hat{\epsilon}_t^p(S)$ is independent of the initial states of the followers, and that minimizing $\hat{\epsilon}_t^p(S)$ is equivalent to minimizing the upper bound of Theorem 2. We will show that $\hat{\epsilon}_t^p(S)$ is a supermodular function of S, after first establishing a connection between $\hat{\epsilon}_t^p(S)$ and a random walk on $\tilde{G}$.

B. Connection to Random Walks on Graphs

We define a random walk on $\tilde{G}$ as follows. Choose δ such that $t = l\delta$ for some integer l, and define $X(l)$ to be a random walk with transition matrix P_δ, where P_δ is defined as in Theorem 2. Then the following theorem provides a mapping between $\hat{\epsilon}_t^p$ and $X(l)$.

Theorem 3: Let X and P_δ be defined as above. Then

$$|(e_i^T P_\delta^l - e_0^T)_j| = \begin{cases} Pr(X(l) = j | X(0) = i), & j \neq 0 \\ 1 - Pr(X(l) = 0 | X(0) = i), & j = 0 \end{cases}$$

Proof: From Section II-A, P_δ is a stochastic matrix, and hence defines a transition matrix for a random walk on $\tilde{G}$. Furthermore, e_i defines a probability distribution on $\tilde{G}$ corresponding to the case where $X(0) = i$. Hence $e_i^T P_\delta^l$ is the probability distribution of $X(l)$ conditioned on an initial distribution e_i^T, and so $(e_i^T P_\delta^l)_j = Pr(X(l) = j | X(0) = i)$. This yields

$$|(e_i^T P_\delta^l)_j - (e_0^T)_j| = |(e_i^T P_\delta^l)_j| = Pr(X(l) = j | X(0) = i)$$

when $j \neq 0$ and

$$\begin{aligned} |(e_i^T P_\delta^l)_j - (e_0^T)_j| &= |Pr(X(l) = j | X(0) = i) - 1| \\ &= 1 - Pr(X(l) = j | X(0) = i) \end{aligned}$$

when $j = 0$. ■

The following theorem establishes that the probability that the random walk has reached the node 0 is a supermodular function of S, and is an intermediate step towards proving that $\hat{\epsilon}_t^p(S)$ is supermodular.

Theorem 4: For any $i \in V$, define $g_{ij}^l(S)$ and $h_i^l(S)$ by $g_{ij}^l(S) \triangleq Pr(X(l) = j | X(0) = i)$ for $j \in V \setminus S$ and $h_i^l(S) = 1 - Pr(X(l) = 0 | X(0) = i)$. Then $g(S)$ and $h(S)$ are both supermodular functions of S.

Intuitively, Theorem 4 states that, as the set S increases, adding a node u to S results in a smaller decrease in the time for a random walk to reach any node in S. Consider a random walk that reaches a node $v \in T \setminus S$, followed by node u, and then reaches the set S. Adding u to S reduces the time to reach S. However, adding v to T does not change the time to reach T, since the walk reaches $v \in T \setminus S$ before node u. This argument is formalized as follows.

Proof of Theorem 4: Let $S \subseteq T$ and $u \in V \setminus T$, and consider $g_{ij}^l(S)$. Let $A_{ij}^l(S)$ denote the event that $X(l) = j$, $X(0) = i$, and $X(r) \notin S$ for all $0 \leq r \leq l$. Since, after arriving at S, the walk transitions to the absorbing state 0,

$$g_{ij}^l(S) = Pr(A_{ij}^l(S)) = \mathbf{E}(\chi(A_{ij}^l(S))), \tag{12}$$

where $\chi(\cdot)$ denotes the indicator function. Furthermore, let $B_{ij}^l(S, u)$ denote the event where $X(0) = i$, $X(l) = j$, $X(r) \notin S$ for $0 \leq r \leq l$, and $X(m) = u$ for some $0 \leq m \leq l$, so that $A_{ij}^l(S) = A_{ij}^l(S \cup \{u\}) \cup B_{ij}^l(S, u)$, $B_{ij}^l(S, u) \cap A_{ij}^l(S \cup \{u\}) = \emptyset$, and

$$\chi(A_{ij}^l(S)) = \chi(A_{ij}^l(S \cup \{u\})) + \chi(B_{ij}^l(S, u)).$$

We observe that, since $S \subseteq T$, $X(r) \notin T$ for all $0 \leq r \leq l$ implies $X(r) \notin S$ for all $0 \leq r \leq l$, i.e. $B_{ij}^l(T, u) \subseteq B_{ij}^l(S, u)$. We have

$$\begin{aligned} \chi(A_{ij}^l(S)) - \chi(A_{ij}^l(S \cup \{u\})) &= \chi(B_{ij}^l(S, u)) \\ &\geq \chi(B_{ij}^l(T, u)) \\ &= \chi(A_{ij}^l(S)) \\ &\quad -\chi(A_{ij}^l(S \cup \{u\})) \end{aligned}$$

which implies that $\chi(A_{ij}^l(S))$ is a supermodular function of S. Since $g_{ij}^l(S) = \mathbf{E}(\chi(A_{ij}^l(S)))$, where the expectation is over all possible sample paths of $X(k)$ for $0 \leq k \leq l$, $g_{ij}^l(S)$ is a finite nonnegative weighted sum of supermodular functions, and is therefore supermodular.

Now, let $C_i^l(S)$ denote the event that $X(0) = i$ and $X(r) \notin S$ for all $0 \leq r \leq l$. By a similar argument, we have that $\chi(C_i^l(S))$ is a supermodular function of S and $h_i^l(S) = \mathbf{E}(\chi(C_i^l(S)))$ is supermodular. ■

Theorem 5: $\hat{\epsilon}_t^p(S)$ is supermodular as a function of S.

Proof: By definition, $\hat{\epsilon}_t^p(S)$ is given by

$$\begin{aligned} \hat{\epsilon}_t^p(S) &= \sum_{i \in V \setminus S} ||P^T e_i - e_0^T||_p^p & (13) \\ &= \sum_{i \in V \setminus S} \left(\sum_{j \in \tilde{V}} |(P^T e_i - e_0^T)_j|^p \right) & (14) \\ &= \sum_{i \in V \setminus S} \left(\left(\sum_{j \neq 0} g_{ij}^l(S)^p \right) + h_i^l(S)^p \right) & (15) \end{aligned}$$

where (14) and (15) follow from Theorem 3 and the definitions of $g_{ij}^l(S)$ and $h_i^l(S)$ in Theorem 4. By Theorem 4 and Lemma 1, $g_{ij}^l(S)^p$ is a supermodular function of S. The error $\hat{\epsilon}_t^p(S)$ is therefore a sum of nonnegative supermodular functions, and hence is supermodular. ■

C. Problem Formulation

We now present two problem formulations. In the first case, the goal of the network owner is to choose a set of up to k leader nodes in order to minimize the convergence error. This problem can be stated as

$$\begin{array}{cl} \text{minimize} & \hat{\epsilon}_t^p(S) \\ \text{s.t.} & |S| \leq k \end{array} \tag{16}$$

In the second case, the minimum-size set of leader nodes must be chosen in order to achieve a given bound α on the convergence error. This problem can be stated as

$$\begin{array}{cl} \text{minimize} & |S| \\ \text{s.t.} & \hat{\epsilon}_t^p(S) \leq \alpha \end{array} \tag{17}$$

Algorithm k-**leaders**: Algorithm for selecting up to k leaders in order to minimize convergence error	**Algorithm leaders-error-**α: Algorithm for selecting minimum number of leaders to achieve a bound α on the convergence error
Input: Topology $G = (V, E)$, weight matrix W Number of leaders, k $S \leftarrow \emptyset$, $i \leftarrow 0$ **while** $i < k$ $v^* \leftarrow \arg\max \{\hat{\epsilon}_t^p(S) - \hat{\epsilon}_t^p(S \cup \{v\})\}$ $S \leftarrow S \cup \{v^*\}$, $i \leftarrow i + 1$ **end while** **return** S	**Input:** Topology $G = (V, E)$, weight matrix W Error bound, α $S \leftarrow \emptyset$, $i \leftarrow 0$ **while** $\hat{\epsilon}_t^p(S) > \alpha$ $v^* \leftarrow \arg\max \{\hat{\epsilon}_t^p(S) - \hat{\epsilon}_t^p(S \cup \{v\})\}$ $S \leftarrow S \cup \{v^*\}$, $i \leftarrow i + 1$ **end while** **return** S

An algorithm for selecting up to k leaders in order to minimize convergence error is as follows. The set of leader nodes S is initialized to the empty set, $\emptyset$. At the t-th iteration, the node that maximizes $\hat{\epsilon}_t^p(S) - \hat{\epsilon}_t^p(S \cup \{v\})$ is selected and added to the leader set. The algorithm terminates after k iterations. A pseudocode description is given as algorithm k-**leaders**.

Theorem 6: Let S^* denote the set of k leader nodes that minimizes the convergence error $\hat{\epsilon}_t^p(S)$. Then algorithm k-**leaders** returns a set S' satisfying

$$\hat{\epsilon}_t^p(S') \leq \left(1 - \frac{1}{e}\right) \hat{\epsilon}_t^p(S^*) + \frac{1}{e} \hat{\epsilon}_{t,max}^p \tag{18}$$

where $\hat{\epsilon}_{t,max}^p \triangleq \max_v \hat{\epsilon}_t^p(\{v\})$.

Proof: Proposition 4.1 of [4] states that, for a nonnegative, nondecreasing submodular function f, a greedy maximization algorithm returns a set S' with $f(S') \geq (1 - 1/e)f(S^*)$, where S^* is the optimal set. Algorithm k-**leaders** is equivalent to greedy maximization of the nonnegative, nondecreasing submodular function $\tau(S) = \hat{\epsilon}_{t,max}^p - \hat{\epsilon}_t^p(S)$, yielding the error bound $\tau(S') \geq (1 - 1/e)\tau(S^*)$. This implies that

$$\hat{\epsilon}_{t,max}^p - \hat{\epsilon}_t^p(S') \geq (1 - 1/e)(\hat{\epsilon}_{t,max}^p - \hat{\epsilon}_t^p(S^*)).$$

Rearranging terms gives the desired result. ■

In order to select the minimum number of leaders to achieve a bound α on the convergence error, a similar approach can be taken. The set of leader nodes $S = \emptyset$ initially. At the t-th iteration, the node v that maximizes $\hat{\epsilon}_t^p(S) - \hat{\epsilon}_t^p(S \cup \{v\})$ is added to the leader set. The algorithm continues until $\hat{\epsilon}_t^p(S) \leq \alpha$. A pseudocode description is given as algorithm **leaders-error-**α.

Theorem 7: Let S^* be the optimum set of leaders for problem (17), and let S' be the set of leaders returned by **leaders-error-**α. Then $\frac{|S'|}{|S^*|} \leq 1 + \log\left(\frac{\hat{\epsilon}_{t,max}^p}{\alpha}\right)$

Proof: The proof follows from Part iii of Theorem 1 of [5], applied to the submodular function $f(S) = \hat{\epsilon}_{t,max}^p - \hat{\epsilon}_t^p(S)$, and using the fact that $\hat{\epsilon}_t^p(S') \leq \alpha$. ■

IV. Simulation Study

A network of 200 nodes following the dynamics of (1) was simulated using Matlab. The nodes were assumed to be randomly deployed over a 1500m x 1500m rectangular area. A link (i, j) was assumed to exist if the distance between nodes i and j was no more than 300m. The weights W_{ij} were chosen uniformly at random from the range $[0, 50]$. The initial node states were chosen independently and at random from $[0, 1]$. Each data point represents an average of 70 trials.

The following four different leader node selection algorithms were simulated and compared. First, a set of leaders was chosen independently at random. Second, the nodes with the highest degree (i.e., largest number of neighbors) were chosen as leaders. Third, nodes with degree closest to the average were chosen. The fourth algorithm in each case was our proposed supermodular optimization approach.

The results of selecting up to k leaders in order to minimize convergence error are summarized in Figure 1(a). As the number of leaders varied from 1 to 20, the convergence error decreased for each algorithm considered. The supermodular optimization approach, however, consistently outperformed the other algorithms; for example, when 5 leaders were used, the convergence error of the supermodular approach was less than a third of the random and average degree-based methods. Similarly, when selecting leaders to meet a given error bound, the number of leaders needed to achieve the bound under the supermodular approach was half that of the other approaches. We also observed that random leader node selection consistently outperformed both degree-based selection algorithms, and that choosing the highest-degree nodes as leaders gave the worst overall performance.

V. Conclusion

In this paper, we introduced and studied the problem of leader selection to minimize convergence error, defined as the ℓ^p-norm of the difference between the actual and desired node states, in leader-follower systems. Prior to our work, existing approaches focused on choosing link weights in order to minimize convergence error; our approach introduces a new design dimension. We first constructed an augmented network graph, and proved that the convergence errors defined under the ℓ^p-norm on the original and augmented graphs are equivalent. We then derived a relationship between the convergence error and a random walk on the augmented graph, and used this insight to prove that the convergence error is a supermodular function of the set of leader nodes. We formulated two leader selection problems, namely the problem of selecting a fixed number of leaders, as well as the problem of selecting the minimum number of leaders to satisfy an upper bound on the convergence error. We showed that both problems can be solved within a supermodular optimization framework, leading to efficient algorithms that approximate the optimal leader set with provable bounds. Our results were illustrated through a simulation study, in which we showed that the supermodular optimization approach significantly outperformed other leader selection algorithms.

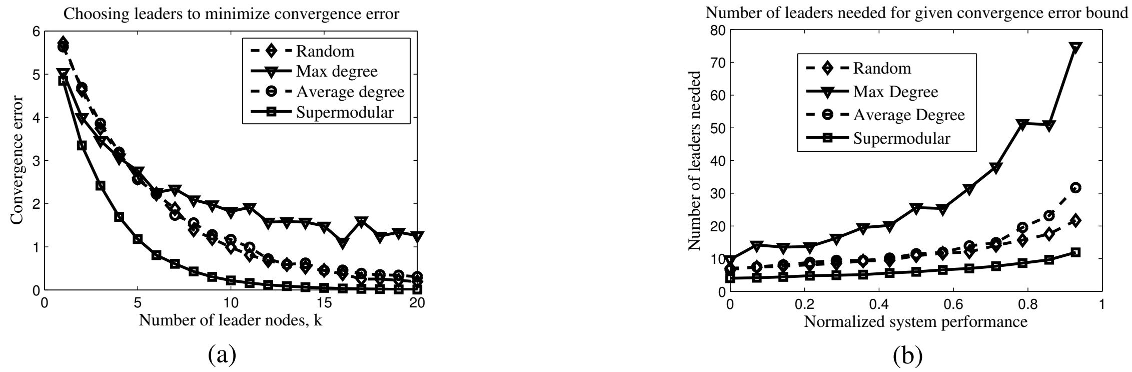

Fig. 1. Leader selection to minimize convergence error. (a) Selecting up to k leaders. The supermodular optimization approach provides lower error than random or degree-based selection. (b) Selecting the minimum number of leaders in order to meet an error bound. The supermodular optimization approach requires fewer leaders than other selection algorithms.

References

[1] W. Ren, R.W. Beard, and E.M. Atkins, "Information consensus in multivehicle cooperative control," *Control Systems, IEEE*, vol. 27, no. 2, pp. 71–82, 2007.

[2] A. Jadbabaie, J. Lin, and A.S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *Automatic Control, IEEE Transactions on*, vol. 48, no. 6, pp. 988–1001, 2003.

[3] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *Information Theory, IEEE Transactions on*, vol. 52, no. 6, pp. 2508–2530, 2006.

[4] G.L. Nemhauser, L.A. Wolsey, and M.L. Fisher, "An analysis of approximations for maximizing submodular set functions," *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.

[5] L.A. Wolsey, "An analysis of the greedy algorithm for the submodular set covering problem," *Combinatorica*, vol. 2, no. 4, pp. 385–393, 1982.

VI. Proof of Lemma 1

A proof of Lemma 1 is as follows. Let $S \subseteq T$ and $u \in V \setminus T$, and define $\rho_u(T) = f(T) - f(T \cup \{u\})$ (resp. $\rho_u(S) = f(S) - f(S \cup \{u\})$). By the definition of supermodularity, $\rho_u(S) \geq \rho_u(T)$, which implies that

$$f(S) = f(S \cup \{u\}) + \rho_u(S) \geq f(S \cup \{u\}) + \rho_u(T). \quad (19)$$

This yields

$$\begin{aligned} h(S) - h(S \cup \{u\}) &= g(f(S)) - g(f(S \cup \{u\})) \\ &= \int_{f(S\cup\{u\})}^{f(S)} g'(t)\ dt \\ &= \int_{f(S\cup\{u\})}^{f(S\cup\{u\})+\rho_u(T)} g'(t)\ dt \\ &\quad + \int_{f(S\cup\{u\})+\rho_u(T)}^{f(S)} g'(t)\ dt \quad (20) \\ &\geq \int_{f(S\cup\{u\})}^{f(S\cup\{u\})+\rho_u(T)} g'(t)\ dt \quad (21) \\ &\geq \int_{f(T\cup\{u\})}^{f(T\cup\{u\})+\rho_u(T)} g'(t)\ dt \quad (22) \\ &= h(T) - h(T \cup \{u\}), \quad (23) \end{aligned}$$

where (20) follows from (19). The fact that g is nondecreasing implies (21), while (22) follows from monotonicity of $f(S)$, which implies that $f(T \cup \{u\}) \leq f(S \cup \{u\})$. Eq. (23) establishes the supermodularity of $h(S)$.