

Multi-Hop Relay Protocols for Fast Consensus Seeking

Zhipu Jin and Richard M. Murray

Abstract—Consensus protocols are distributed algorithms in networked multi-agent systems. Based on the local information, agents automatically converge to a common consensus state and the convergence speed is determined by the algebraic connectivity of the communication network. In order to achieve a fast consensus seeking, we propose the multi-hop relay protocols, where each agent can expand its knowledge by employing multi-hop paths in the network. We demonstrate that multi-hop relay protocols can enlarge the algebraic connectivity without physically changing the network topology. Moreover, communication delays are discussed and a tradeoff is identified between the convergence speed and the time delay sensitivity.

Index Terms—Networked multi-agent systems, consensus protocol, multi-hop relay protocol, distributed algorithms, convergence speed, time delay.

I. INTRODUCTION

Consensus seeking of networked multi-agent systems has attracted researchers from different disciplines. Vicsek *et al.* [1] proposed a simple and popular model for self-driven particles alignment problem in which each agent updates its heading based on the average of its own heading and the neighbors'. Using simulation, they showed that all agents moved in the same direction eventually. A theoretical explanation for Vicsek's model is given by Jadbabaie *et al.* in [2]. Olfati-Saber and Murray [3] proposed a simple consensus protocol and showed that, for balanced directed graphs, this protocol solved the average consensus problem. Moreover, consensus seeking under general connected directed graphs was discussed in [4], [5].

When the agent dynamics is nontrivial, the consensus behavior can be treated as the synchronization problem of interconnected dynamical systems. Different approaches were employed, such as Lyapunov's direct method in [6] and Laplacian matrix decomposition method in [7], [8]. Also, sufficient conditions for interconnected dynamical systems synchronization over general interaction topology were discussed in [9].

Consensus seeking has many applications in peer-to-peer networks [10], sensor fusion [11], and distributed Kalman filtering [12]. The convergence speed is important. Xiao and Boyd treated a consensus process as an optimal linear iteration problem and showed that the convergence speed can be increased by finding the optimal weight associated with each communication link [13]. In their result, the global structure of the network must be known beforehand. Olfati-Saber proposed a "random rewiring" procedure to boost the

consensus process for large scale networks [14]. However, physically changing the topology may be difficult. So the question we want to ask is: can we get a better convergence performance without changing the network topology?

Fortunately, the answer is yes. In this paper, we propose multi-hop relay protocols based on multi-hop paths in networks. The idea is simple: each agent can get more information by passing its neighbors states to others. The improvement with the two-hop relay protocol is given explicitly. Since relay protocols do not change the topology, it is easy to be implemented. Furthermore, the effect of communication delays is considered and a tradeoff between convergence speed and delay sensitivity is discussed.

The remainder of this paper is organized as follows: In Section II, a brief review of concepts in algebraic graph theory and some preliminary results about consensus protocol are presented. We then propose multi-hop relay protocols for fast consensus seeking in Section III and emphasize on the two-hop relay protocol. Section IV is devoted to investigating consensus protocols with communication delays. Explicit results of delay margins are given. Examples and simulation results are provided in Section V, and conclusions are summarized in Section VI.

II. CONSENSUS PROTOCOLS AND CONSENSUS STATE

A directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is used to represent the communication topology in a networked multi-agent system where \mathcal{V} is a set of vertices and $\mathcal{E} \subseteq \mathcal{V}^2$ is a set of edges. Each edge in the graph is denoted by (v_i, v_j) , which means that agent v_i can receive information from agent v_j . For edge (v_i, v_j) , we call v_i the *head* and v_j the *tail*. A directed graph \mathcal{G} is called *symmetric* if, whenever $(v_i, v_j) \in \mathcal{E}$, $(v_j, v_i) \in \mathcal{E}$ as well. In a directed graph, the number of edges whose head is v_i is called the *out-degree* of node v_i ; the number of edges whose tail is v_i is called the *in-degree* of node v_i . The set of neighbors of vertex v_i is denoted by $N(v_i) = \{v_j \in \mathcal{V} : (v_i, v_j) \in \mathcal{E}\}$.

A *path* in a directed graph is a sequence $[u_0, \dots, u_r]$ of distinct vertices such that $(u_{i-1}, u_i) \in \mathcal{E}$ for i from 1 to r . A path is also called an *m-hop path* if it has $m+1$ vertices. A *weak path* is a sequence $[u_0, \dots, u_r]$ of distinct vertices such that either (u_{i-1}, u_i) or (u_i, u_{i-1}) belongs to \mathcal{E} . A directed graph is *weakly connected* if any two vertices in the graph can be joined by a weak path, and it is *strongly connected* if any two vertices can be joined by a path. If a directed graph is not weakly connected, it is *disconnected*.

If a graph is symmetric, there is no difference between weakly connected and strongly connected. In some literature,

Zhipu Jin is with the Department of Electrical Engineering, California Institute of Technology, Pasadena, CA 91125, USA. jzp@cds.caltech.edu

Richard M. Murray is with Faculty of the Division of Engineering and Applied Science, California Institute of Technology, Pasadena, CA, USA. murray@cds.caltech.edu

a connected and symmetric graph is simply called *connected*. Fig. 1 reveals the relationship between these concepts.

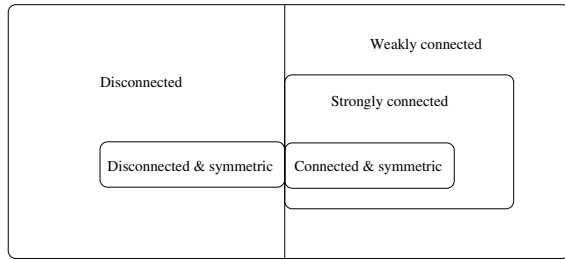


Fig. 1. Classification of directed graphs.

An *adjacency matrix* $A = \{\alpha_{ij}\}$ of \mathcal{G} with n vertices is defined as

$$\alpha_{ij} = \begin{cases} 1, & (v_i, v_j) \in \mathcal{E} \\ 0, & \text{otherwise.} \end{cases}$$

More generally, a *weighted adjacency matrix* $\mathcal{A} = \{a_{ij}\}$ is defined as

$$a_{ij} = \alpha_{ij} \cdot w_{ij}$$

where $w_{ij} > 0$ is the weight associated with edge (v_i, v_j) . Then, the out-degree of node v_i is $\sum_j a_{ij}$ and the in-degree of node v_i is $\sum_j a_{ji}$. Let D be the diagonal matrix with the out-degree of each vertex along the diagonal, the *Laplacian matrix* L is defined by $L = D - \mathcal{A}$.

Let x_i denote the state of agent v_i . A networked multi-agent system reaches a *consensus* if $x_i = x_j$ for all v_i and $v_j \in \mathcal{V}$. This common value is called the *consensus state*, which is depicted by η . A continuous consensus protocol based on the “nearest neighbor rule” is represented by

$$\dot{x}_i = - \sum_{j \in N(v_i)} w_{ij} (x_i - x_j), \quad (1)$$

and the consensus process is presented by

$$\dot{X} = -LX \quad (2)$$

where $X = [x_1, \dots, x_n]'$. It is known that for a connected and symmetric graph, protocol (1) solves the average consensus problem, i.e., $\eta = \sum_i x_i(0)/n$.

III. MULTI-HOP REPLAY PROTOCOLS

For a connected and symmetric graph, L is symmetric and the eigenvalues can be denoted as $0 = \lambda_1 < \lambda_2 \leq \dots \leq \lambda_n$. The consensus convergence speed is bounded by the second smallest eigenvalue λ_2 that is called “algebraic connectivity”. Clearly, λ_2 is determined by the topology and the edge weights. In this section, we introduce the multi-hop relay protocols, which employ multi-hop paths in the graph instead of changing the connectivity and the weights.

A. Two-Hop Relay Protocol

The distributed two-hop relay protocol is described as

$$\dot{x}_i = - \sum_{j \in N(v_i)} w_{ij} \left((x_i - x_j) + \sum_{k \in N(v_j)} w_{jk} (x_i - x_k) \right). \quad (3)$$

In this protocol, what vertex v_j sends to v_i is not only its own state, but also a collection of its instantaneous neighbors’ states. It is equivalent to adding virtual “two-hop” paths as additional edges to the original graph. We define the *two-hop directed graph* $\hat{\mathcal{G}} = (\mathcal{V}, \hat{\mathcal{E}})$ as the graph that has the same vertex set, and all the edges are “two-hop” paths of \mathcal{G} . Fig. 2 shows an example of the two-hop directed graph.

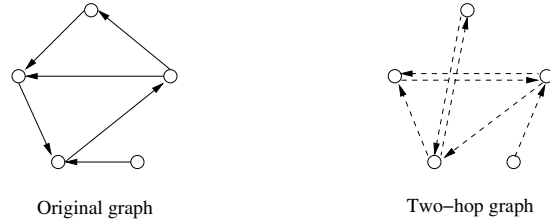


Fig. 2. A directed graph and its two-hop directed graph.

There may exist self-loops in two-hop graph $\hat{\mathcal{G}}$, i.e., the head and the tail of an edge are the same. This is common when \mathcal{G} is symmetric. However, according to Equ. (3), these self-loops have no contributions to the dynamics and are omitted. Moreover, multiple two-hop paths may exist between a pair of vertices. In $\hat{\mathcal{G}}$, we consider them as one edge in $\hat{\mathcal{G}}$ and the weight equals the sum of these two-hop paths’ weights. Thus, the adjacency matrix $\hat{\mathcal{A}} = \{\hat{a}_{ik}\}$ of $\hat{\mathcal{G}}$ is:

$$\hat{a}_{ik} = \begin{cases} \sum_j w_{ij} w_{jk}, & (v_i, v_k) \in \hat{\mathcal{E}} \\ 0, & \text{otherwise.} \end{cases}$$

The corresponding out-degree diagonal and Laplacian matrices for $\hat{\mathcal{G}}$ are denoted by \hat{D} and \hat{L} , respectively. For a directed graph \mathcal{G} with two-hop relay protocol, consider the joint graph $\tilde{\mathcal{G}} = \mathcal{G} \cup \hat{\mathcal{G}} = (\mathcal{V}, \mathcal{E} \cup \hat{\mathcal{E}})$, it is obvious that the dynamics of the consensus process is

$$\dot{X} = -\tilde{L}X. \quad (4)$$

Also, the two-hop relay protocol needs extra communication bandwidth. Assuming that the graph is static. We rewrite the protocol (1) as:

$$\dot{x}_i = -x_i \sum_{j \in N(v_i)} w_{ij} + \sum_{j \in N(v_i)} w_{ij} x_j \quad (5)$$

and the two-hop relay protocol (3) as

$$\dot{x}_i = -x_i \sum_{j \in N(v_i)} w_{ij} \left(1 + \sum_{k \in N(v_j)} w_{jk} \right) + \sum_{j \in N(v_i)} w_{ij} \left(x_j + \sum_{k \in N(v_j)} w_{jk} x_k \right). \quad (6)$$

For protocol (1), what link (v_i, v_j) transmits is the value of x_j . For protocol (3), what link (v_i, v_j) transmits is the value of x_j , $\sum w_{jk} x_k$, and $\sum w_{jk}$. However, for a static graph, $\sum w_{jk}$ is a constant and only needs to be transmitted once. Thus, the two-hop relay protocol needs twice the communication bandwidth except at the very beginning.

B. Convergence speed of two-hop relay protocol

In order to simplify the analysis, from now on we assume \mathcal{G} is symmetric and connected. Then, the two-hop graph $\hat{\mathcal{G}}$

is symmetric, and the joint graph $\tilde{\mathcal{G}}$ is also symmetric and connected.

Theorem 3.1: If graph \mathcal{G} is connected and symmetric, then

$$\lambda_2(L) \leq \lambda_2(\tilde{L}). \quad (7)$$

Proof: For any vector x , it is true that

$$\begin{aligned} x^T \tilde{L}x &= x^T Lx + x^T \hat{L}x \\ &= \sum_{(v_i, v_j) \in \mathcal{E}} w_{ij}^2 (x_i - x_j)^2 \\ &\quad + \sum_{(v_i, v_j) \in \mathcal{E}} w_{ij}^2 (x_i - x_j)^2. \end{aligned}$$

We pick a unit vector x and let it be orthogonal to $\mathbf{1}$, then

$$\frac{x^T Lx}{x^T x} = \frac{\sum_{(v_i, v_j) \in \mathcal{E}} (x_i - x_j)^2}{\sum_{v_i \in \mathcal{V}} x_i^2} \geq \lambda_2(L)$$

and the equation holds only when x is an eigenvector associated with $\lambda_2(L)$.

Combining these two results, if we take x to be a unit eigenvector of \tilde{L} , orthogonal to $\mathbf{1}$, associated with eigenvalue $\lambda_2(\tilde{L})$, then we have

$$\lambda_2(\tilde{L}) = \frac{x^T \tilde{L}x}{x^T x} = \frac{x^T (L + \hat{L})x}{x^T x} \geq \lambda_2(L) + \frac{x^T \hat{L}x}{x^T x}. \quad (8)$$

Theorem 3.1 shows that two-hop relay protocol increases the convergence speed. The improvement depends on the topology of $\hat{\mathcal{G}}$. It can be shown that the edge set of $\hat{\mathcal{G}}$ is not empty if the original graph has more than two vertices.

Proposition 3.2: If $\hat{\mathcal{G}}$ is connected,

$$\lambda_2(\tilde{L}) \geq \lambda_2(L) + \lambda_2(\hat{L}). \quad (9)$$

Proof: When $\hat{\mathcal{G}}$ is connected, $x^T \hat{L}x / x^T x \geq \lambda_2(\hat{L}) > 0$. Thus, two-hop relay protocol improves the algebraic connectivity by at least $\lambda_2(\hat{L})$. ■

However, it is not true that $\hat{\mathcal{G}}$ is always connected. Fig. 3 shows a simple example. The original graph on the left is symmetric and connected, but the two-hop graph on the right is composed of two disconnected subgraphs.

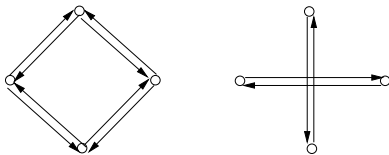


Fig. 3. An example of disconnected two-hop directed graph.

C. Multi-hop relay protocol

It is possible to extend the two-hop relay protocol to the multi-hop relay protocol. The protocol for m -hop relay protocol can be written as

$$\dot{x}_i = - \underbrace{\sum_j w_{ij} ((x_i - x_j) + \sum_k w_{jk} ((x_i - x_k) + \dots))}_{m \text{ layers}}. \quad (10)$$

Clearly it adds more virtual edges to the original graph and enforces the convergence speed. However, there are three drawbacks. First, the worst case computation complexity of

the m -hop relay protocol on each agent is $O(n^{m-1})$. For large scale networks, it quickly becomes infeasible. Second, at least m -times communication bandwidth are needed. Third, communication delays can accumulate along m -hop paths and we will discuss that how sensitive those protocols are to communication latency.

IV. TWO-HOP RELAY PROTOCOLS WITH TIME DELAYS

For communication delays, we consider the transfer function associated with edge (v_i, v_j) with latency τ_{ij} is $h_{ij}(s) = e^{-\tau_{ij}s}$. Delays will be accumulated in two-hop relay protocol. We focus on the simplest case where all delays are identical, i.e., $\tau_{ij} = \tau$ for any $(v_i, v_j) \in \mathcal{E}$, and the weights are uniform. The protocol (1) can be written as

$$\dot{x}_i = - \sum_{j \in N(v_i)} w_{ij} (x_i(t - \tau) - x_j(t - \tau)) \quad (11)$$

and the two-hop relay protocol is

$$\begin{aligned} \dot{x}_i &= - \sum_{j \in N(v_i)} w_{ij} \left((x_i(t - \tau) - x_j(t - \tau)) \right. \\ &\quad \left. + \sum_{k \in N(v_j)} w_{jk} (x_i(t - 2\tau) - x_k(t - 2\tau)) \right). \end{aligned} \quad (12)$$

Equations (2) and (4) change to

$$\dot{X} = -LX(t - \tau) \quad (13)$$

and

$$\dot{X} = -LX(t - \tau) - \hat{L}X(t - 2\tau), \quad (14)$$

respectively.

Let $Z = V^{-1}X$ where

$$V^{-1} = \begin{bmatrix} \mathbf{1} & \mathbf{1} \\ \mathbf{1} & -I \end{bmatrix}. \quad (15)$$

For the two-hop relay protocol, we have to

$$\dot{Z} = -V^{-1}LVZ(t - \tau) - V^{-1}\hat{L}VZ(t - 2\tau). \quad (16)$$

Note that

$$V^{-1}LV = \begin{bmatrix} 0 & \mathbf{0} \\ \mathbf{0} & L_{22} \end{bmatrix} \quad \text{and} \quad V^{-1}\hat{L}V = \begin{bmatrix} 0 & \mathbf{0} \\ \mathbf{0} & \hat{L}_{22} \end{bmatrix}$$

where L_{22} is full rank.

Assume that $X(t) = 0$ for any $t < 0$. All states except z_1 of the autonomous system (16) asymptotically converge to 0, i.e., x_i converges to $\eta = \sum x_i(0)/n$, if and only if the following characteristic polynomial

$$p_{22}(s, e^{-\tau s}) = \det(sI + L_{22}e^{-\tau s} + \hat{L}_{22}e^{-2\tau s}) \quad (17)$$

has no zero in the closed right half plane (RHP). This condition is equivalent to the case that the characteristic polynomial

$$p(s, e^{-\tau s}) = \det(sI + Le^{-\tau s} + \hat{L}e^{-2\tau s}) \quad (18)$$

has no zero in the closed RHP except the simple zero at the origin. In [15], $p(s, e^{-\tau s})$ is called a real *quasipolynomial* of s . In the rest of this paper, we will consistently use this name.

One of the essential properties of quasipolynomials is the continuity of the zeros with respect to delay. In other words,

when τ increases, zeros in left half plane (LHP) move to RHP. Time delay does not affect the zero $s = 0$. Thus, we need to find minimum value of τ such that the first stable zero crosses the imaginary axis. Besides, the conjugate symmetry property of quasipolynomials makes it possible to calculate the critical value of the time delay and the corresponding crossing frequency.

Definition 4.1: Given initial value $X(0)$ and assumption $X(t) = 0$ for any $t < 0$, the smallest value of τ such that the system cannot converge to a consensus is determined as

$$\tau^* = \min\{\tau > 0 \mid p(j\omega, e^{-j\tau\omega}) = 0 \text{ and } \omega \neq 0\}. \quad (19)$$

We call τ^* the *delay margin* of the consensus protocol.

It is true that, for any $\tau \in [0, \tau^*)$, the system of (13) or (14) converges to the average consensus state $\eta = \sum x_i(0)/n$.

Lemma 4.2: Let τ^* and $\tilde{\tau}^*$ indicate the delay margin of (13) and (14) respectively, then $\tau^* \geq \tilde{\tau}^*$.

Proof: First, let us find τ^* . According to the Schur theorem, there exists a unitary matrix T such that $U = T^{-1}LT$ is upper triangular with the eigenvalues along the diagonal. So we have

$$\begin{aligned} \det(sI + Le^{-\tau s}) &= \det(sI + TUT^{-1}e^{-\tau s}) \\ &= \det(T(sI + Ue^{-\tau s})T^{-1}) \\ &= s \cdot \prod_{i=2}^n (s + \lambda_i(L)e^{-\tau s}). \end{aligned}$$

We need to find the smallest $\tau > 0$ such that the first stable zero reaches the imaginary axis. Let $s = j\omega$ and we have

$$j\omega = -e^{\tau j\omega} \lambda_i(L). \quad (20)$$

Solving this equation gives us

$$\begin{cases} \omega &= \lambda_i(L) \neq 0 \\ \tau &= \pi/2\lambda_i(L). \end{cases} \quad (21)$$

So the delay margin

$$\tau^* = \min \pi/2\lambda_i(L) = \pi/2\lambda_n(L). \quad (22)$$

Next, we consider $\tilde{\tau}^*$. The above approach fails for $\det(sI + Le^{-\tau s} + \hat{L}e^{-2\tau s})$. However, it is obvious that $\tilde{\tau}^*$ should be no bigger than the delay margin for $\det(sI + (L + \hat{L})e^{-\tau s})$, which is $\pi/2\lambda_n(\tilde{L})$. Moreover, $\lambda_n(\tilde{L}) \geq \lambda_n(L)$ according to [16]. So we have

$$\tau^* = \pi/(2\lambda_n(L)) \geq \pi/(2\lambda_n(\tilde{L})) \geq \tilde{\tau}^*. \quad \blacksquare$$

Lemma 4.2 just shows us that the delay sensitivity of the two-hop relay protocol is no better than protocol (1). The following theorem gives us explicit results on $\tilde{\tau}^*$ by using the frequency-sweeping method.

Theorem 4.3: For system (14), define

$$\bar{\tau}_i = \min_{1 \leq k \leq n-1} \theta_k^i / \omega_k^i$$

when the generalized eigenvalues $\lambda_i(G(s), H)$ satisfy the following equation:

$$\lambda_i(G(j\omega_k^i), H) = e^{-j\theta_k^i}$$

for some $\omega_k^i \in (0, \infty)$ and $\theta_k^i \in [0, 2\pi)$, where

$$G(s) = \begin{bmatrix} 0 & I \\ -sI & -L_{22} \end{bmatrix} \text{ and } H = \begin{bmatrix} I & 0 \\ 0 & \hat{L}_{22} \end{bmatrix}.$$

Then the consensus delay margin of (14) is

$$\tilde{\tau}^* = \min_{1 \leq i \leq 2(n-1)} \bar{\tau}_i.$$

Proof: Finding a generalized eigenvalue for matrix pair (A, B) is the problem of finding a scalar λ_i and a nonzero vector y such that $Ay = \lambda_i B y$.

According to the aforementioned similarity transform, the system (14) converges to a consensus if the following system

$$\begin{bmatrix} \dot{z}_2 \\ \vdots \\ \dot{z}_n \end{bmatrix} = -L_{22} \begin{bmatrix} \dot{z}_2(t-\tau) \\ \vdots \\ \dot{z}_n(t-\tau) \end{bmatrix} - \hat{L}_{22} \begin{bmatrix} \dot{z}_2(t-2\tau) \\ \vdots \\ \dot{z}_n(t-2\tau) \end{bmatrix}$$

is stable.

According to the Schur determinant complement, we have

$$\begin{aligned} &\det(sI + L_{22}e^{-\tau s} + \hat{L}_{22}e^{-2\tau s}) \\ &= \det(e^{-\tau s}I) \det(\hat{L}_{22}e^{-\tau s} + L_{22} + sI \cdot e^{\tau s}) \\ &= \det \left(\begin{bmatrix} I \cdot e^{-\tau s} & -I \\ sI & \hat{L}_{22}e^{-\tau s} + L_{22} \end{bmatrix} \right) \\ &= \det \left(e^{-\tau s} \begin{bmatrix} I & \\ & \hat{L}_{22} \end{bmatrix} - \begin{bmatrix} 0 & I \\ -sI & -L_{22} \end{bmatrix} \right). \end{aligned} \quad (23)$$

Now the quasipolynomial with two delay terms transfers to a new quasipolynomial with a single delay term as

$$\det(sI + L_{22}e^{-\tau s} + \hat{L}_{22}e^{-2\tau s}) = \det(G(s) - e^{-\tau s}H) \quad (24)$$

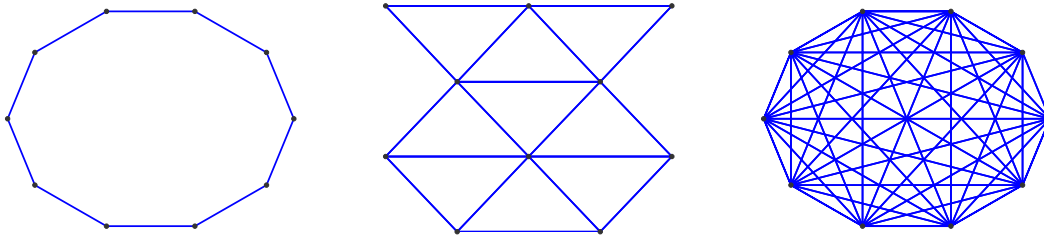
where

$$G(s) = \begin{bmatrix} 0 & I \\ -sI & -L_{22} \end{bmatrix} \text{ and } H = \begin{bmatrix} I & 0 \\ 0 & \hat{L}_{22} \end{bmatrix}. \quad (25)$$

Since $\tau \in \mathcal{R}$, whenever a zero is located on the imaginary axis, there exists $s = j\omega$ so that $e^{-j\omega\tau}$ is a generalized eigenvalue of $(G(s), H)$ with magnitude 1. Thus, we can transfer the problem of finding a τ so that the quasipolynomial has zeros with pure imaginary parts to the problem of finding a ω so that $(G(s), H)$ has a generalized eigenvalue with magnitude 1. Because $\text{rank}(H) = 2(n-1)$, there are at most $2(n-1)$ generalized eigenvalues of $(G(s), H)$. When s moves along the imaginary axis from 0 to $j\infty$, there exists at most $n-1$ frequency ω_k^i so that $\|\lambda_i(G(j\omega_k^i), H)\|_2 = \|e^{-j\theta_k^i}\|_2 = 1$. So the delay margin $\tilde{\tau}^*$ is the minimum value of all possible $\bar{\tau}_k^i = \theta_k^i / \omega_k^i$. \blacksquare

V. EXAMPLES AND SIMULATION RESULTS

In order to verify the efficiency of the two-hop relay consensus protocol, we test it on three graphs. Fig. 4 shows the topologies of \mathcal{G}_1 , \mathcal{G}_2 , and \mathcal{G}_3 from left to right. Topology \mathcal{G}_1 is a two-regular graph, \mathcal{G}_2 is a net in which each vertex connects to neighbors located inside a certain range, and \mathcal{G}_3 is a complete graph. All of them have ten vertices. They are symmetric and connected. Each pair of edges (v_i, v_j) and (v_j, v_i) belong to those graphs is denoted by a single link. For simplicity, we assume that $w_{ij} = 1$ for any edge.

Fig. 4. Three examples with different topologies: \mathcal{G}_1 , \mathcal{G}_2 , and \mathcal{G}_3 .TABLE I
PERFORMANCE VS. ROBUSTNESS FOR RELAY PROTOCOLS

	Algebraic connectivity λ_2		Delay margin τ^*	
	Without relay	With relay	Without relay	With relay
\mathcal{G}_1	0.382	1.7639	0.3927	0.1796
\mathcal{G}_2	0.9118	7.3846	0.2167	0.0396
\mathcal{G}_3	10	90	0.1571	0.0095

Fig. 5 to Fig. 7 show the simulation results of \mathcal{G}_2 with the same initial conditions and different delays. Note that, even though the system can become unstable, the sum of the states keeps constant. Table I shows the algebraic connectivities and delay margins for all three graphs with or without two-hop relay protocols. Delay margins without relays are calculated according to Lemma 4.2. Delay margins with relay are computed using frequency sweep method mentioned in Theorem 4.3. Note that the magnitudes of generalized eigenvalues inevitably exceed 1 after a certain ω , and the computation needs to be done only over a finite frequency interval. We actually run the computation twice. The first time we find an appropriate frequency interval. The second time we use much smaller frequency steps over the interval in order to find an accurate value of the delay margin.

For each graph, the relay protocol improves the convergence speed. However, time-delay robustness is impaired due to the delay accumulation along the two-hop paths. Moreover, along the columns of the table, we can tell that algebraic connectivity increases and delay margin decreases when the graph includes more links. We put these data in Fig. 8. For each bar, the right lower point corresponds to protocol (1) and the left upper point corresponds to the two-hop relay protocol. It is true that relay protocols actually boost convergence speed while sacrificing the delay margin.

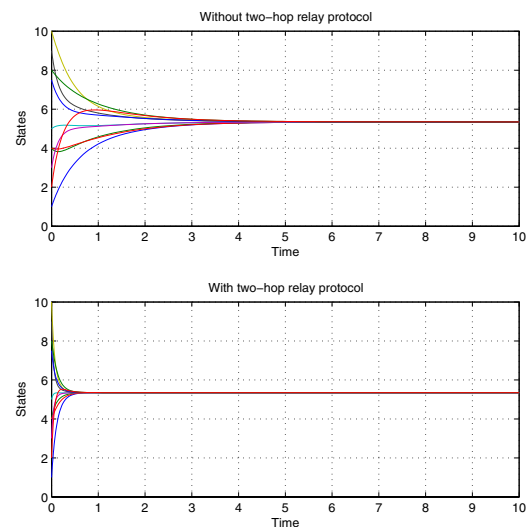
Our results can be extended to non-symmetric graphs. Fig. 9 shows simulation results for the non-symmetric graph in Fig. 2. It is clear that the two-hop relay protocol still converges faster, but the final value of the consensus state is changed.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we propose multi-hop relay protocols for fast consensus seeking and emphasize on the two-hop relay protocol with symmetric and connected graphs. Using relay

protocols can efficiently improve the convergence speed without physically changing the topology. But, we need extra communication bandwidth. A tradeoff between the performance of convergence speed and robustness of communication delay are identified by investigating three typical topologies with relay protocols. The more edges a graph includes, the faster the convergence speed is, while the more sensitive the protocol is to the time delay. We use the frequency sweep method to find the delay margin with multiple commensurate delays efficiently.

Future work includes extending our results to non-symmetric graphs. Also, it will be interesting that studying large group of graphs and putting their performance/robustness data into Fig. 8. Relationships between patterns and topology characters should be carefully examined. Comprehensively describing and deeply understanding this tradeoff can benefit us for topology and protocol design for multi-agent networks.

Fig. 5. States of graph \mathcal{G}_2 with no delay.

VII. ACKNOWLEDGEMENTS

The authors would like to thank Prof. Reza Olfati-Saber, from Dartmouth College, for the fruitful discussions. This research is partly supported by AFOSR grant FA9550-04-1-0169.

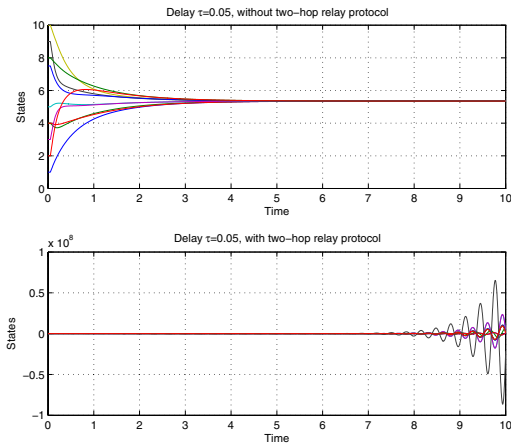


Fig. 6. States of graph \mathcal{G}_2 with delay $\tau = 0.05$.

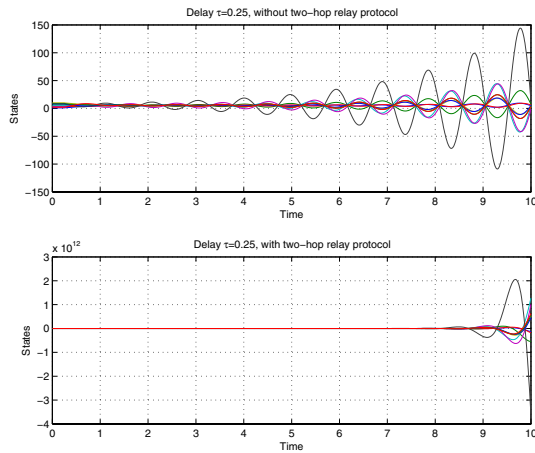


Fig. 7. States of graph \mathcal{G}_2 with delay $\tau = 0.25$.

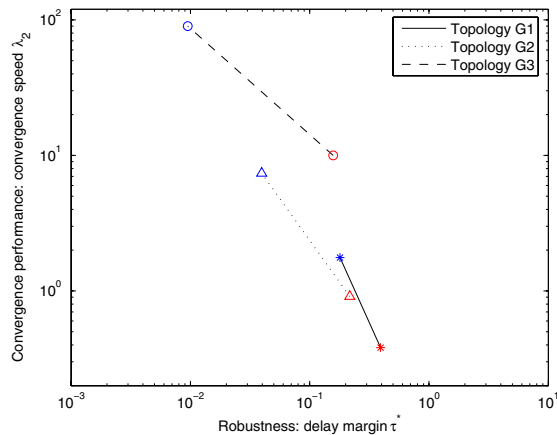


Fig. 8. Tradeoff between λ_2 and τ^* .

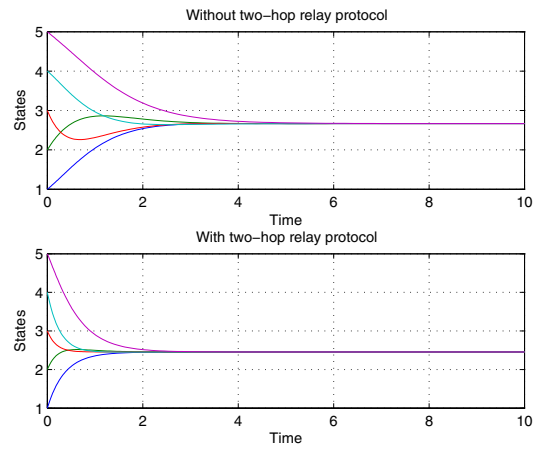


Fig. 9. Simulation for a non-symmetric graph.

REFERENCES

- [1] T. Vicsek, A. Czirok, E. B. Jacob, I. Cohen, and O. Schochet, "Novel type of phase transitions in a system of self-driven particles," *Physical Review Letters*, vol. 75, pp. 1226–1229, 1995.
- [2] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. Automat. Contr.*, vol. 48, no. 6, pp. 988–1001, Jun. 2003.
- [3] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Automat. Contr.*, vol. 49, no. 9, pp. 1520–1533, Sep. 2004.
- [4] W. Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Trans. Automat. Contr.*, vol. 50, no. 5, pp. 655–661, May 2005.
- [5] Z. Jin and R. M. Murray, "Consensus controllability for coordinated multiple vehicle control." Gainesville, FL: The 6th International Conference on Cooperative Control and Optimization, Feb. 2006.
- [6] C. W. Wu and L. O. Chua, "Synchronization in an array of linearly coupled dynamical systems," *IEEE Trans. Circuits Syst.*, vol. 42, no. 8, pp. 430–447, Aug. 1995.
- [7] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE Transactionis on Automatic Control*, vol. 49, no. 9, pp. 1465–1476, Sep. 2004.
- [8] Z. Jin and R. M. Murray, "Double-graph control strategy of multi-vehicle formations," in *Proceeding of 43rd IEEE Conference on Decision and Control*, vol. 2, Dec. 2004, pp. 1988–1994.
- [9] C. W. Wu, "Synchronization in networks of nonlinear dynamical systems coupled via a directed graph," *Nonlinearity*, vol. 18, pp. 1057–1064, 2005.
- [10] M. Mehyar, D. Spanos, J. Pongsajapan, S. H. Low, and R. M. Murray, "Distributed averaging on peer-to-peer networks," *Proceedings of the 44th IEEE conference on Decision and Control*, pp. 7446–7451, 2005.
- [11] R. Olfati-Saber and J. S. Shamma, "Consensus filters for sensor networks and distributed sensor fusion," *Proceedings of the 44th IEEE conference on Decision and Control*, pp. 6698–6703, 2005.
- [12] R. Olfati-Saber, "Distributed kalman filter with embedded consensus filters," *Proceedings of the 44th IEEE conference on Decision and Control*, pp. 8179–8184, 2005.
- [13] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems and Control Letters*, vol. 53, pp. 65–78, 2004.
- [14] R. Olfati-Saber, "Ultrafast consensus in small-world networks," in *Proceeding of 2005 American Control Conference*, Jun. 2005, pp. 2371–2378.
- [15] K. Gu, V. L. Kharitonov, and J. Chen, *Stability of Time-Delay Systems*. Birkhäuser, 2003.
- [16] C. Godsil and G. Royle, *Algebraic Graph Theory*, ser. Graduate Texts in Mathematics. Springer, 2001.