Consensus and Cooperation in Networked Multi-Agent Systems[*]

Reza Olfati-Saber[1], J. Alex Fax[2], and Richard M. Murray[3]

[1] *Thayer School of Engineering, Dartmouth College, Hanover, NH 03755,* `olfati@seas.ucla.edu`
[2] *Northrop Grumman NSD, 21240 Burbank Blvd., Woodland Hills, CA 91367,* `Alex.Fax@ngc.com`
[3] *Control and Dynamical Systems, California Institute of Technology, Pasadena, CA 91125,* `murray@caltech.edu`

Summary. This paper provides a theoretical framework for analysis of consensus algorithms for multi-agent networked systems with an emphasis on the role of directed information flow, robustness to changes in network topology due to link/node failures, time-delays, and performance guarantees. An overview of basic concepts of information consensus in networks and methods of convergence and performance analysis for the algorithms are provided. Our analysis framework is based on tools from matrix theory, algebraic graph theory, and control theory. We discuss the connections between consensus problems in networked dynamic systems and diverse applications including synchronization of coupled oscillators, flocking, formation control, fast consensus in small-world networks, Markov processes and gossip-based algorithms, load balancing in networks, rendezvous in space, distributed sensor fusion in sensor networks, and belief propagation. We establish direct connections between spectral and structural properties of complex networks and the speed of information diffusion of consensus algorithms. A brief introduction is provided on networked systems with nonlocal information flow that are considerably faster than distributed systems with lattice-type nearest neighbor interactions. Simulation results are presented that demonstrate the role of small-world effects on the speed of consensus algorithms and cooperative control of multi-vehicle formations.

Key words: multi-agent systems, consensus algorithms, information fusion, networked systems, cooperative control, algebraic connectivity, graph Laplacians, flocking, synchronization of coupled oscillators, small-world networks

1 Introduction: Consensus and Cooperation

Consensus problems have a long history in computer science and form the foundation of the field of *distributed computing* [51]. Formal study of consensus problems in groups of experts originated in *management science and statistics* in 1960's (See DeGroot [19] and references therein). The ideas of *statistical consensus theory* by DeGroot reappeared two decades later in aggregation of information with uncertainty obtained from multiple sensors[4][6] and medical experts [93].

Distributed computation over networks has a tradition in systems and control theory starting with the pioneering work of Borkar and Varaiya [10] and Tsitsiklis and Athens [90] on *asynchronous asymptotic agreement problem* for distributed decision-making systems. This effort was summarized in [7] with applications to parallel computing.

In networks of *agents* (or dynamic systems), "consensus" means to reach an agreement regarding a certain quantity of interest that depends on the state of all agents. A "consensus algorithm" (or protocol) is

---

[4] This is known as *sensor fusion* and is an important application of modern consensus algorithms that will be discussed later.

an interaction rule that specifies the information exchange between an agent and all of its neighbors on the network[5].

The theoretical framework for posing and solving *consensus problems* for networked dynamic systems was introduced by Olfati-Saber and Murray in [76, 70] building on the earlier work of Fax and Murray [27, 28]. The study of the *alignment problem* involving reaching an agreement—without computing any objective functions—appeared in the work of Jadbabaie *et al.* [38]. Further theoretical extensions of this work were presented in [59, 74] with a look toward treatment of directed information flow in networks as shown in Fig. 1 (a).

The common motivation behind the work in [10, 90, 70] is the rich history of consensus protocols in computer science [51], whereas Jadbabaie *et al.* [38] attempted to provide a formal analysis of emergence of alignment in the simplified model of flocking by Viscek *et al.* [91]. The setup in [70] was originally created with the vision of designing *agent-based amorphous computers* [1, 62] for collaborative information processing in networks. Later, [70] was used in development of flocking algorithms with guaranteed convergence and the capability to deal with obstacles and adversarial agents [66].

Graph Laplacians and their spectral properties [29, 58, 55, 32] are important graph-related matrices that play a crucial role in convergence analysis of consensus and alignment algorithms. Graph Laplacians are an important point of focus of this paper. It is worth mentioning that the second smallest eigenvalue of graph Laplacians called *algebraic connectivity* quantifies the speed of convergence of consensus algorithms. The notion of algebraic connectivity of graphs has appeared in a variety of other areas including low-density parity-check codes (LDPC) in information theory and communications [84], Ramanujan graphs [50] in number theory and quantum chaos, and combinatorial optimization problems such as the max-cut problem [58].

More recently, there has been a tremendous surge of interest—among researchers from various disciplines of engineering and science—in problems related to multi-agent networked systems with close ties to consensus problems. This includes subjects such as consensus [47, 9, 5, 15, 54, 8, 79], collective behavior of flocks and swarms [66, 80, 60, 95, 30], sensor fusion [64, 71, 33], random networks [34, 73], synchronization of coupled oscillators [81, 39, 72, 73, 14], algebraic connectivity[6]of complex networks [65, 12, 43], asynchronous distributed algorithms [54, 26], formation control for multi-robot systems [21, 68, 69, 24, 89, 88, 48, 96, 20], optimization-based cooperative control [75, 42, 37, 2], dynamic graphs [56, 61, 40, 99], complexity of coordinated tasks [36, 44, 52, 53], and consensus-based belief propagation in Bayesian networks [78, 67]. A detailed discussion of selected applications will be presented shortly.

In this paper, we focus on the work described in five key papers—namely, Jadbabaie, Lin, and Morse [38], Olfati-Saber and Murray [70], Fax and Murray [28], Moreau [59], and Ren and Beard [74]— that have been instrumental in paving the way for more recent advances in study of *self-organizing networked systems*, or *swarms*. These networked systems are comprised of locally interacting mobile/static *agents* equipped with dedicated *sensing, computing, and communication* devices. As a result, we now have a better understanding of complex phenomena such as flocking [66], or design of novel *information fusion* algorithms for *sensor networks* that are robust to node and link failures [64, 86, 98, 13, 78, 67].

Gossip-based algorithms such as the push-sum protocol [41] are important alternatives in computer science to Laplacian-based consensus algorithms in this paper. Markov processes establish an interesting connection between the information propagation speed in these two categories of algorithms proposed by computer scientists and control theorists [11].

The contribution of this paper is to present a cohesive overview of the key results on theory and applications of consensus problems in networked systems in a unified framework. This includes basic notions in information consensus and control theoretic methods for convergence and performance analysis of consensus protocols that heavily rely on matrix theory and spectral graph theory. A byproduct of this framework is to demonstrate that seemingly different consensus algorithms in the literature [38, 28, 70, 59, 74] are closely related. Applications of consensus problems in areas of interest to researchers in computer science, physics, biology, mathematics, robotics, and control theory are discussed in this introduction.

---

[5] The term "nearest neighbors" is more commonly used in physics than "neighbors" when applied to particle/spin interactions over a lattice (e.g. Ising model).

[6] To be defined in Section 2.1.

1.1 Consensus in Networks

The interaction topology of a network of agents is represented using a directed graph $G = (V, E)$ with the set of nodes $V = \{1, 2, \ldots, n\}$ and edges $E \subseteq V \times V$. The neighbors of agent i are denoted by $N_i = \{j \in V : (i, j) \in E\}$. According to [70], a simple consensus algorithm to reach an agreement regarding the state of n *integrator agents* with dynamics $\dot{x}_i = u_i$ can be expressed as an nth-order linear system on a graph:

$$\dot{x}_i(t) = \sum_{j \in N_i} (x_j(t) - x_i(t)) + b_i(t), \quad x_i(0) = z_i \in \mathbb{R}, b_i(t) = 0 \tag{1}$$

The *collective dynamics* of the group of agents following protocol (1) can be written as

$$\dot{x} = -Lx \tag{2}$$

where $L = [l_{ij}]$ is the *graph Laplacian* of the network and its elements are defined as follows:

$$l_{ij} = \begin{cases} -1 & i \neq j, \\ |N_i| & i = j. \end{cases} \tag{3}$$

Here, $|N_i|$ denotes the number of neighbors of node i (or *out-degree* of node i). Fig. 1 shows two equivalent forms of the consensus algorithm in equations (1) and (2) for agents with a scalar state. The role of the input bias b in Fig. 1 (b) is defined later.

According to the definition of graph Laplacian in (3), all row-sums of L are zero because of $\sum_j l_{ij} = 0$. Therefore, L always has a zero eigenvalue $\lambda_1 = 0$. This zero eigenvalues corresponds to the eigenvector $\mathbf{1} = (1, \ldots, 1)^T$ because $\mathbf{1}$ belongs to the null-space of L ($L\mathbf{1} = 0$). In other words, an equilibrium of system (2) is a state in the form $x^* = (\alpha, \ldots, \alpha)^T = \alpha\mathbf{1}$ where all nodes agree. Based on analytical tools from algebraic graph theory [32], we later show that x^* is a unique equilibrium of (2) (up to a constant multiplicative factor) for *connected* graphs.
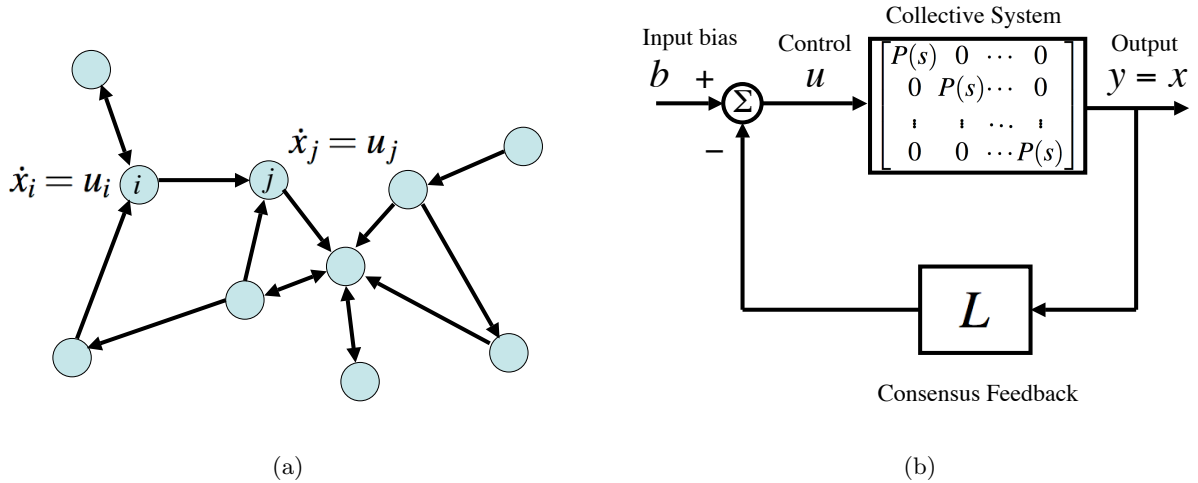

Fig. 1. Two equivalent forms of consensus algorithms: (a) a *network* of integrator agents in which agent i receives the state x_j of its neighbor, agent j, if there is a link (i, j) connecting the two nodes; and (b) the *block diagram* for a network of interconnected dynamic systems all with identical transfer functions $P(s) = 1/s$. The *collective networked system* has a diagonal transfer function and is a MIMO (multi-input multi-output) linear system.

One can show that for a connected network, the equilibrium $x^* = (\alpha, \ldots, \alpha)^T$ is globally exponentially stable. Moreover, the consensus value is $\alpha = 1/n \sum_i z_i$ that is equal to the average of the initial values.

This implies that irrespective of the initial value of the state of each agent, all agents reach an asymptotic consensus regarding the value of the function $f(z) = 1/n \sum_i z_i$.

While the calculation of $f(z)$ is simple for small networks, its implications for very large networks is more interesting. For example, if a network has $n = 10^6$ nodes and each node can only talk to $\log_{10}(n) = 6$ neighbors, finding the average value of the initial conditions of the nodes is more complicated. The role of protocol (1) is to provide a systematic consensus mechanism in such a large network to compute the average. There are a variety of functions that can be computed in a similar fashion using synchronous or asynchronous distributed algorithms (See [70, 67, 5, 13, 54]).

1.2 The *f*-Consensus Problem and Meaning of Cooperation

To understand the role of cooperation in performing coordinated tasks, we need to distinguish between unconstrained and constrained consensus problems. An unconstrained consensus problem is simply the alignment problem in which it suffices that the state of all agents asymptotically be the same. In contrast, in distributed computation of a function $f(z)$, the state of all agents has to asymptotically become equal to $f(z)$, meaning that the consensus problem is constrained. We refer to this constrained consensus problem as the *f-consensus problem*.

Solving the f-consensus problem is a cooperative task and requires willing participation of all the agents. To demonstrate this fact, suppose a single agent decides not to cooperate with the rest of the agents and keep its state unchanged. Then, the overall task cannot be performed despite the fact that the rest of the agents reach an agreement. Furthermore, there could be scenarios in which multiple agents that form a coalition do not cooperate with the rest and removal of this coalition of agents and their links might render the network disconnected. In a disconnected network, it is impossible for all nodes to reach an agreement (unless all nodes initially agree which is a trivial case).

From the above discussion, *cooperation* can be informally interpreted as "giving consent to providing one's state and following a common protocol that serves the group objective."

One might think that solving the alignment problem is not a cooperative task. The justification is that if a single agent (called a leader) leaves its value unchanged, all others will asymptotically agree with the leader according to the consensus protocol and an alignment is reached. However, if there are multiple leaders where two of whom are in disagreement, then no consensus can be asymptotically reached. Therefore, alignment is in general a cooperative task as well.

Formal analysis of the behavior of systems that involve more than one type of agent is more complicated, particularly, in presence of *adversarial agents* in non-cooperative games [31, 82]. The focus of this paper is on cooperative multi-agent systems.

1.3 Iterative Consensus Algorithms and Markov Chains

In Section 2, we show how an iterative consensus algorithm that corresponds to the discrete-time version of system (1) is a Markov chain

$$\pi(k + 1) = \pi(k)P \tag{4}$$

with $P = I - \epsilon L$ and a small $\epsilon > 0$. Here, the ith element of the row vector $\pi(k)$ denotes the probability of being in state i at iteration k. It turns out that for any arbitrary graph G with Laplacian L and a sufficiently small ϵ, the matrix P satisfies the property $\sum_j p_{ij} = 1$ with $p_{ij} \geq 0, \forall i, j$. Hence, P is a valid *transition probability matrix* for the Markov chain in (4). The reason *matrix theory* [35] is so widely used in analysis of consensus algorithms [38, 28, 70, 59, 74, 56] is primarily due to the structure of P in equation (4) and its connection to graphs[7].

There are interesting connections between this Markov chain and the speed of information diffusion in gossip-based averaging algorithms [41, 11].

---

[7] In honor of the pioneering contributions of Oscar Perron (1907) to the theory of nonnegative matrices, were refer to P as the *Perron Matrix* of graph G (See Section 2.3 for details).

One of the early applications of consensus problems was *dynamic load balancing* [18] for parallel processors with the same structure as system (4). To this date, load balancing in networks proves to be an active area of research in computer science.

1.4 Applications

Many seemingly different problems that involve interconnection of dynamic systems in various areas of science and engineering happen to be closely related to consensus problems for multi-agent systems. In this section, we provide an account of the existing connections.

Synchronization of Coupled Oscillators

The problem of *synchronization of coupled oscillators* has attracted numerous scientists from diverse fields including physics, biology, neuroscience, and mathematics [45, 57, 87, 25]. This is partly due to the emergence of synchronous oscillations in coupled neural oscillators. Let us consider the generalized *Kuramoto model* of coupled oscillators on a graph with dynamics

$$\dot{\theta}_i = \kappa \sum_{j \in N_i} \sin(\theta_j - \theta_i) + \omega_i \tag{5}$$

where θ_i and ω_i are the phase and frequency of the ith oscillator. This model is the natural nonlinear extension of the consensus algorithm in (1) and its linearization around the *aligned state* $\theta_1 = \ldots = \theta_n$ is identical to system (2) plus a nonzero input bias $b_i = (\omega_i - \bar{\omega})/\kappa$ with $\bar{\omega} = 1/n \sum_i \omega_i$ after a change of variables $x_i = (\theta_i - \bar{\omega}t)/\kappa$.

In [81], Sepulchre *et al.* show that if κ is sufficiently large, then for a network with all-to-all links, synchronization to the aligned state is globally achieved for all initial states. Recently, synchronization of networked oscillators under variable time-delays was studied in [72]. We believe that the use of convergence analysis methods that utilize the spectral properties of graph Laplacians will shed light on performance and convergence analysis of self-synchrony in oscillator networks [73].

Flocking Theory

Flocks of mobile agents equipped with sensing and communication devices can serve as *mobile sensor networks* for massive distributed sensing in an environment [17]. A theoretical framework for design and analysis of flocking algorithms for mobile agents with obstacle-avoidance capabilities is developed by Olfati-Saber [66]. The role of consensus algorithms in particle-based flocking is for an agent to achieve velocity matching with respect to its neighbors. In [66], it is demonstrated that flocks are networks of dynamic systems with a dynamic topology. This topology is a proximity graph that depends on the state of all agents and is determined locally for each agent, i.e. the topology of flocks is a state-dependent graph. The notion of state-dependent graphs was introduced by Mesbahi [56] in a context that is independent of flocking.

Fast Consensus in Small-Worlds

In recent years, network design problems for achieving faster consensus algorithms has attracted considerable attention from a number of researchers. In Xiao and Boyd [97], design of the *weights* of a network is considered and solved using semi-definite convex programming. This leads to a slight increase in algebraic connectivity of a network that is a measure of speed of convergence of consensus algorithms. An alternative approach is to keep the weights fixed and design the *topology* of the network to achieve a relatively high algebraic connectivity. A randomized algorithm for network design is proposed by Olfati-Saber [65] based on random rewiring idea of Watts and Strogatz [92] that led to creation of their celebrated *small-world model*. The random rewiring of existing links of a network gives rise to considerably faster consensus algorithms. This is due to multiple orders of magnitude increase in algebraic connectivity of the network in comparison to a lattice-type nearest-neighbort graph.

Rendezvous in Space

Another common form of consensus problems is *rendezvous* in space [3, 46]. This is equivalent to reaching a *consensus in position* by a number of agents with an interaction topology that is position induced (i.e. a proximity graph). We refer the reader to [16] and references therein for a detailed discussion. This type of rendezvous is an unconstrained consensus problem that becomes challenging under variations in the network topology. Flocking is somewhat more challenging than rendezvous in space because it requires both inter-agent and agent-to-obstacle collision avoidance.

Distributed Sensor Fusion in Sensor Networks

The most recent application of consensus problems is distributed sensor fusion in sensor networks . This is done by posing various distributed averaging problems require to implement a Kalman filter [64, 71], approximate Kalman filter [86], or linear least-squares estimator [98] as *average-consensus problems*. Novel low-pass and high-pass *consensus filters* are also developed that dynamically calculate the average of their inputs in sensor networks [71, 85].

Distributed Formation Control

Multi-vehicle systems are an important category of networked systems due to their commercial and military applications. There are two broad approaches to distributed formation control: i) representation of formations as rigid structures [69, 23] and the use of gradient-based controls obtained from their structural potentials [68] and ii) representation of formations using the vectors of relative positions of neighboring vehicles and the use of consensus-based controllers with input bias. We discuss the later approach here.

A theoretical framework for design and analysis of distributed controllers for multi-vehicle formations of type ii) was developed by Fax and Murray [28]. Moving in formation is a cooperative task and requires consent and collaboration of every agent in the formation. In [28], graph Laplacians and matrix theory were extensively used which makes one wonder whether relative-position based formation control is a consensus problem. The answer is yes. To see this, consider a network of *self-interested agents* whose individual desire is to minimize their local cost $U_i(x) = \sum_{j \in N_i} \|x_j - x_i - \mathbf{r}_{ij}\|^2$ via a distributed algorithm (x_i is the position of vehicle i with dynamics $\dot{x}_i = u_i$ and $\mathbf{r}_{ij}$ is a desired inter-vehicle relative-position vector). Instead, if the agents use gradient-descent algorithm on the collective cost $\sum_{i=1}^{n} U_i(x)$ using the following protocol

$$\dot{x}_i = \sum_{j \in N_i} (x_j - x_i - \mathbf{r}_{ij}) = \sum_{j \in N_i} (x_j - x_i) + b_i \tag{6}$$

with input bias $b_i = \sum_{j \in N_i} \mathbf{r}_{ji}$ (See Fig. 1 (b)), the objective of every agent will be achieved. This is the same as the consensus algorithm in (1) up to the nonzero bias terms b_i. This non-zero bias plays no role in stability analysis of system (6). Thus, distributed formation control for integrator agents is a consensus problem. The main contribution of the work by Fax and Murray is to extend this scenario to the case where all agents are multi-input multi-output linear systems $\dot{x}_i = Ax_i + Bu_i$. Stability analysis of relative-position based formation control for multi-vehicle systems is extensively covered in Section 4.

1.5 Outline

The outline of the paper is as follows. Basic concepts and theoretical results in information consensus are presented in Section 2. Convergence and performance analysis of consensus on networks with switching topology are given in Section 3. A theoretical framework for cooperative control of formations of networked multi-vehicle systems is provided in Section 4. Some simulation results related to consensus in complex networks including small-worlds are presented in Section 5. Finally, some concluding remarks are stated in Section 6.

2 Information Consensus in Networked Systems

Consider a network of decision-making *agents* with dynamics $\dot{x}_i = u_i$ interested in reaching a *consensus* via local communication with their neighbors on a graph $G = (V, E)$. By reaching a consensus, we mean asymptotically converging to a one-dimensional *agreement space* characterized by the following equation

$$x_1 = x_2 = \ldots = x_n.$$

This agreement space can be expressed as $x = \alpha\mathbf{1}$ where $\mathbf{1} = (1, \ldots, 1)^T$ and $\alpha \in \mathbb{R}$ is the *collective decision* of the group of agents. Let $A = [a_{ij}]$ be the adjacency matrix of graph G. The set of *neighbors* of a agent i is N_i and defined by

$$N_i = \{j \in V : a_{ij} \neq 0\}; \quad V = \{1, \ldots, n\}.$$

Agent i communicates with agent j if j is a neighbor of i (or $a_{ij} \neq 0$). The set of all nodes and their neighbors defines the edge set of the graph as $E = \{(i, j) \in V \times V : a_{ij} \neq 0\}$.

A *dynamic graph* $G(t) = (V, E(t))$ is a graph in which the set of edges $E(t)$ and the adjacency matrix $A(t)$ are time-varying. Clearly, the set of neighbors $N_i(t)$ of every agent in a dynamic graph is a time-varying set as well. Dynamic graphs are useful for describing the network topology of *mobile sensor networks* and flocks [66].

It is shown in [70] that the linear system

$$\dot{x}_i(t) = \sum_{j \in N_i} a_{ij}(x_j(t) - x_i(t)) \tag{7}$$

is a *distributed consensus algorithm*, i.e. guarantees convergence to a collective decision via local inter-agent interactions. Assuming that the graph is *undirected* ($a_{ij} = a_{ji}$ for all i, j), it follows that the sum of the state of all nodes is an invariant quantity, or $\sum_i \dot{x}_i = 0$. In particular, applying this condition twice at times $t = 0$ and $t = \infty$ gives the following result

$$\alpha = \frac{1}{n} \sum_i x_i(0).$$

In other words, if a consensus is asymptotically reached, then necessarily the collective decision is equal to the average of the initial state of all nodes. A consensus algorithm with this specific invariance property is called an *average-consensus* algorithm [76] and has broad applications in distributed computing on networks (e.g. sensor fusion in sensor networks).

The dynamics of system (7) can be expressed in a compact form as

$$\dot{x} = -Lx \tag{8}$$

where L is known as the *graph Laplacian* of G. The graph Laplacian is defined as

$$L = D - A \tag{9}$$

where $D = \text{diag}(d_1, \ldots, d_n)$ is the degree matrix of G with elements $d_i = \sum_{j \neq i} a_{ij}$ and zero off-diagonal elements. By definition, L has a right eigenvector of $\mathbf{1}$ associated with the zero eigenvalue[8] because of the identity $L\mathbf{1} = 0$.

For the case of undirected graphs, graph Laplacian satisfies the following sum-of-squares (SOS) property:

$$x^T Lx = \frac{1}{2} \sum_{(i,j) \in E} a_{ij}(x_j - x_i)^2. \tag{10}$$

By defining a quadratic *disagreement function* as

---

[8] These properties were discussed earlier in the introduction for graphs with 0-1 weights.

$$\varphi(x) = \frac{1}{2} x^T L x, \tag{11}$$

it becomes apparent that algorithm (7) is the same as

$$\dot{x} = -\nabla \varphi(x),$$

or the gradient-descent algorithm. This algorithm globally asymptotically converges to the agreement space provided that two conditions hold: 1) L is a positive semidefinite matrix and 2) the only equilibrium of (7) is $\alpha \mathbf{1}$ for some α. Both of these conditions hold for a connected graph and follow from the SOS property of graph Laplacian in (10). Therefore, an average-consensus is asymptotically reached for all initial states. This fact is summarized in the following lemma:

Lemma 1. *Let G be a connected undirected graph. Then, the algorithm in (7) asymptotically solves an average-consensus problem for all initial states.*

2.1 Algebraic Connectivity and Spectral Properties of Graphs

Spectral properties of Laplacian matrix are instrumental in analysis of convergence of the class of linear consensus algorithms in (7). According to Gershgorin theorem [35], all eigenvalues of L in the complex plane are located in a closed disk centered at $\Delta + 0j$ with a radius of $\Delta = \max_i d_i$, i.e. the *maximum degree* of a graph. For undirected graphs, L is a symmetric matrix with real eigenvalues and therefore the set of eigenvalues of L can be ordered sequentially in an ascending order as

$$0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n \leq 2\Delta \tag{12}$$

The zero eigenvalue is known as the trivial eigenvalue of L. For a connected graph G, $\lambda_2 > 0$ (i.e. the zero eigenvalue is isolated). The second smallest eigenvalue of Laplacian λ_2 is called *algebraic connectivity* of a graph [29]. Algebraic connectivity of the network topology is a measure of performance/speed of consensus algorithms [70].

Example 1. Fig. 2 shows two examples of networks of integrator agents with different topologies. Both graphs are undirected and have 0-1 weights. Every node of the graph in Fig. 2 (a) is connected to its 4 nearest neighbors on a ring. The other graph is a proximity graph of points that are distributed uniformly at random in a square. Every node is connected to all of its spatial neighbors within a closed ball of radius $r > 0$. Here are the important degree information and Laplacian eigenvalues of these graphs:

$$\begin{aligned} a)\ &\lambda_1 = 0, \lambda_2 = 0.48, \lambda_n = 6.24, \Delta = 4 \\ b)\ &\lambda_1 = 0, \lambda_2 = 0.25, \lambda_n = 9.37, \Delta = 10 \end{aligned} \tag{13}$$

In both cases, $\lambda_i < 2\Delta$ for all i.

2.2 Convergence Analysis for Directed Networks

The convergence analysis of the consensus algorithm in (7) is equivalent to proving that the agreement space characterized by $x = \alpha \mathbf{1}, \alpha \in \mathbb{R}$ is an asymptotically stable equilibrium of system (7). The stability properties of system (7) is completely determined by the location of the Laplacian eigenvalues of the network. The eigenvalues of the adjacency matrix are irrelevant to the stability analysis of system (7), unless the network is *k-regular* (all of its nodes have the same degree k).

   The following lemma combines a well-known rank property of graph Laplacians with Gershgorin theorem to provide spectral characterization of Laplacian of a fixed directed network G. Before stating the lemma, we need to define the notion of strong connectivity of graphs. A graph is *strongly connected* (SC) if there is a directed path connecting any two arbitrary nodes s, t of the graph[9].

---

[9] The notion of strong connectivity applies to directed graphs (or digraphs). For undirected graphs SC is the same as connectivity.
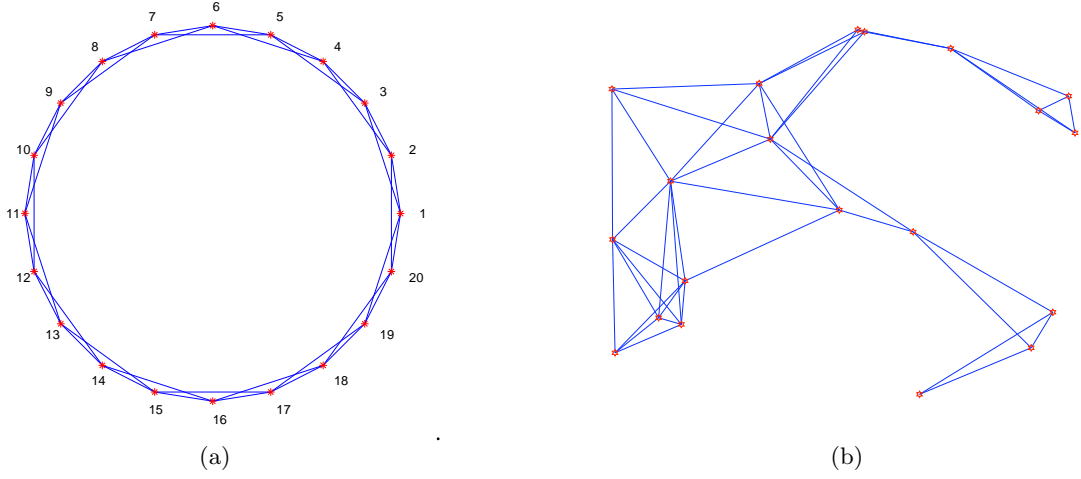
(a)   (b)

Fig. 2. Examples of networks with $n = 20$ nodes: a) a regular network with 80 links and b) a random network with 65 links.

Lemma 2. *(spectral localization) Let G be a strongly connected digraph on n nodes. Then $rank(L) = n - 1$ and all nontrivial eigenvalues of L have positive real parts. Furthermore, suppose G has $c \geq 1$ strongly connected components, then $rank(L) = n - c$.*

Proof. The proof of the rank property for digraphs is given in [70]. The proof for undirected graphs is available in the algebraic graph theory literature [32]. The positivity of the real parts of the eigenvalues follow from the fact that all eigenvalues are located in a Gershgorin disk in the closed right-hand plane that touches the imaginary axis at zero. The second part follows from the first part after relabeling the nodes of the digraph so that its Laplacian becomes a block diagonal matrix.   □

Remark 1. Lemma 2 holds under a weaker condition of existence of a directed spanning tree for G. G has a directed spanning tree if there exists a node r (a root) such that all other nodes can be linked to r via a directed path. This type of condition on existence of directed spanning trees have appeared in [38, 59, 74]. The *root* node is commonly known as a *leader* [38].

The essential results regarding convergence and decision value of Laplacian-based consensus algorithms for directed networks with a fixed topology are summarized in the following theorem. Before stating this theorem, we need to define an important class of digraphs that appear frequently throughout this section.

Definition 1. *(balanced digraphs [70]) A digraph G is called* balanced *if $\sum_{j \neq i} a_{ij} = \sum_{j \neq i} a_{ji}$ for all $i \in V$.*

In a balanced digraph, the total weight of edges entering a node and leaving the same node are equal for all nodes. The most important property of balanced digraphs is that $w = \mathbf{1}$ is also a left eigenvector of their Laplacian (or $\mathbf{1}^T L = 0$).

Theorem 1. *Consider a network of n agents with topology G applying the following consensus algorithm*

$$\dot{x}_i(t) = \sum_{j \in N_i} a_{ij}(x_j(t) - x_i(t)), \ x(0) = z \tag{14}$$

Suppose G is a strongly connected digraph. Let L be the Laplacian of G with a left eigenvector $\gamma = (\gamma_1, \ldots, \gamma_n)$ satisfying $\gamma^T L = 0$. Then

i) A consensus is asymptotically reached for all initial states;

ii) The algorithm solves the *f*-consensus problem with the linear function $f(z) = (\gamma^T z)/(\gamma^T \mathbf{1})$, *i.e. the group decision is* $\alpha = \sum_i w_i z_i$ *with* $\sum_i w_i = 1$;

iii) If the digraph is balanced, an average-consensus is asymptotically reached and $\alpha = (\sum_i x_i(0))/n$.

Proof. The convergence of the consensus algorithm follows from Lemma 2. To show part ii), note that the collective dynamics of the network is $\dot{x} = -Lx$. This means that $y = \gamma^T x$ is an invariant quantity due to $\dot{y} = -\gamma^T Lx = 0, \forall x$. Thus, $\lim_{t \to \infty} y(t) = y(0)$, or $\gamma^T(\alpha \mathbf{1}) = \gamma^T x(0)$ that implies the group decision is $\alpha = (\gamma^T z)/\sum_i \gamma_i$. Setting $w_i = \gamma_i/\sum_i \gamma_i$, we get $\alpha = w^T z$. Part iii) follows as a special case of the statement in part ii) because for a balanced digraph $\gamma = \mathbf{1}$ and $w_i = 1/n, \forall i$.   □

Remark 2. In [70], it is shown that a *necessary and sufficient condition* for L to have a left eigenvector of $\gamma = \mathbf{1}$ is that G must be a balanced digraph.

A challenging problem is to analyze convergence of a consensus algorithm for a *dynamic network* with a *switching topology* $G(t)$ that is time-varying. Various aspects of this problem has been addressed by several groups during the recent years [38, 70, 59, 74] and will be discussed in detail.

2.3 Consensus in Discrete-Time and Matrix Theory

An iterative form of the consensus algorithm can be stated as follows in discrete-time:

$$x_i(k+1) = x_i(k) + \epsilon \sum_{j \in N_i} a_{ij}(x_j(k) - x_i(k)). \tag{15}$$

The discrete-time collective dynamics of the network under this algorithm can be written as

$$x(k+1) = Px(k), \tag{16}$$

with $P = I - \epsilon L$ (I is the identity matrix) and $\epsilon > 0$ is the step-size. In general, $P = \exp(-\epsilon L)$ and the algorithm in (15) is a special case that only uses communication with first-order neighbors[10]. We refer to P as *Perron matrix* of a graph G with parameter ϵ.

Three important types of *non-negative matrices* are irreducible, stochastic, and primitive (or ergodic) matrices [35]. A matrix A is *irreducible* if its associated graph is strongly connected. A non-negative matrix is called row (or column) *stochastic* if all of its row-sums (or column-sums) are 1. An irreducible stochastic matrix P is *primitive* if it has only one eigenvalue with maximum modulus.

Lemma 3. *Let G be a digraph with n nodes and maximum degree* $\Delta = \max_i(\sum_{j \neq i} a_{ij})$. *Then, the Perron matrix P with parameter* $\epsilon \in (0, 1/\Delta]$ *satisfies the following properties:*

i) P is a row stochastic non-negative matrix with a trivial eigenvalue of 1;
ii) All eigenvalues of P are in a unit circle;
iii) If G is a balanced graph, then P is a doubly stochastic matrix;
iv) If G is strongly connected and $0 < \epsilon < 1/\Delta$, then P is a primitive matrix.

Proof. Since $P = I - \epsilon L$, we get $P\mathbf{1} = \mathbf{1} - \epsilon L\mathbf{1} = \mathbf{1}$ which means the row sums of P is 1. Moreover, 1 is a trivial eigenvalue of P for all graphs. To show that P is non-negative, notice that $P = I - \epsilon D + \epsilon A$ due to definition of Laplacian $L = D - A$. ϵA is a non-negative matrix. The diagonal elements of $I - \epsilon D$ are $1 - \epsilon d_i \geq 1 - d_i/\Delta \geq 0$ which implies $I - \epsilon D$ is non-negative. Since the sum of two non-negative matrices is a non-negative matrix, P is a non-negative row stochastic matrix. To prove part ii), one notices that all eigenvectors of P and L are the same. Let λ_j be the jth eigenvalue of L. Then, the jth eigenvalue of P is

$$\mu_j = 1 - \epsilon \lambda_j. \tag{17}$$

---

[10] The set of mth-order neighbors is the set of neighbors of node i on a graph with adjacency matrix A^m.

Based on Gershgorin theorem, all eigenvalues of L are in the disk $|s - \Delta| \leq \Delta$. Defining $z = 1 - s/\Delta$, we have $|z| \leq 1$ which proves part ii). If G is a balanced digraph, then $\mathbf{1}$ is the left eigenvector of L, or $\mathbf{1}^T L = 0$. This means that $\mathbf{1}^T P = \mathbf{1}^T - \epsilon \mathbf{1}^T L = \mathbf{1}^T$ which implies the column sums of P are 1. This combined with the result in part i) gives part iii). To prove part iv), note that if G is strongly connected, then P is an irreducible matrix [35]. To prove that P is primitive, we need to establish that it has a single eigenvalue with maximum modulus of 1. For all $0 < \epsilon < 1/\Delta$, the transformation $\mu = 1 - \epsilon s$ maps the circle $|s - \Delta| = \Delta$ into a circle that is located strictly inside a unit disk passing through the point $\mu = 1$. This means that only a single eigenvalue at $\mu_1 = 1$ can have a modulus of 1. $\qquad\square$

Remark 3. The condition $\epsilon < 1/\Delta$ in part iv) is necessary. If an incorrect step-size of $\epsilon = 1/\Delta$ is used. Then, P would no longer be a primitive matrix because it could have multiple eigenvalues of modulus 1. The counterexample is a directed cycle of length n with a Laplacian that has n roots on the boundary of the Gershgorin disk $|s - \Delta| \leq \Delta$. With the choice of $\epsilon = 1/\Delta = 1$, one gets a Perron matrix that is irreducible but has n eigenvalues on the boundary of the unit circle. This is a common mistake that is repeated by some of the researchers in the past.

The convergence analysis of the discrete-time consensus algorithm relies on the following well-known lemma in matrix theory.

Lemma 4. *(Perron-Frobenius, [35]) Let P be a primitive non-negative matrix with left and right eigenvectors w and v, respectively, satisfying $Pv = v$, $w^T P = w^T$, and $v^T w = 1$. Then $\lim_{k \to \infty} P^k = vw^T$.*

The convergence and group decision properties of iterative consensus algorithms $x \leftarrow Px$ with row stochastic Perron matrices is stated in the following result. It turns out that this discrete-time convergence result is almost identical to its continuous-time counterpart.

Theorem 2. *Consider a network of agents $x_i(k+1) = x_i(k) + u_i(k)$ with topology G applying the distributed consensus algorithm*

$$x_i(k + 1) = x_i(k) + \epsilon \sum_{j \in N_i} a_{ij}(x_j(k) - x_i(k)) \tag{18}$$

where $0 < \epsilon < 1/\Delta$ and Δ is the maximum degree of the network. Let G be a strongly connected digraph. Then

i) *A consensus is asymptotically reached for all initial states;*
ii) *The group decision value is $\alpha = \sum_i w_i x_i(0)$ with $\sum_i w_i = 1$;*
iii)*If the digraph is balanced (or P is doubly-stochastic), an average-consensus is asymptotically reached and $\alpha = (\sum_i x_i(0))/n$.*

Proof. Considering that $x(k) = P^k x(0)$, a consensus is reached in discrete-time, if the limit $\lim_{k \to \infty} P^k$ exists. According to Lemma 4, this limit exists for primitive matrices. Based on part iv) of Lemma 3, P is a primitive matrix. Thus, $\lim_{k \to \infty} x(k) = v(w^T x(0))$ with $v = \mathbf{1}$, or $x_i \to \alpha = w^T x(0)$ for all i as $k \to \infty$. Hence, the group decision value is $\alpha = \sum_i w_i x_i(0)$ with $\sum_i w_i = 1$ (due to $v^T w = 1$). If the graph is balanced, based on part iii) of Lemma 3, P is a column stochastic matrix with a left eigenvector of $w = (1/n)\mathbf{1}$. The group decision becomes equal to $\alpha = (1/n)\mathbf{1}^T x_i(0)$ and average-consensus is asymptotically reached. $\qquad\square$

So far, we have presented a unified framework for analysis of convergence of consensus algorithms for directed networks with fixed topology in both discrete-time and continuous-time. A comparison between the two cases of continuous-time and discrete-time consensus are listed in Table 1.

2.4 Performance of Consensus Algorithms

The speed of reaching a consensus is the key in design of the network topology as well as analysis of performance of a consensus algorithm for a given network. Let us first focus on balanced directed networks

Table 1. Continuous-Time vs. Discrete-Time Consensus

	CT	DT
Dynamics	$\dot{x} = -Lx$	$x(k+1) = Px(k)$
Key Matrix	L (Laplacian)	$P = I - \epsilon L$ (Perron)
Connected G	converges	converges
Decision (general)	$\sum_i w_i x_i(0)$	$\sum_i w_i x_i(0)$
Decision (balanced)	$\sum_i x_i(0)/n$	$\sum_i x_i(0)/n$

that include undirected networks as a special case. This is primarily due to the fact that the collective dynamics of the network of agent applying a continuous- or discrete-time consensus algorithm in this case has an invariant quantity $\alpha = (\sum_i x_i)/n$. To demonstrate this in discrete-time, note that $\mathbf{1}^T P = \mathbf{1}^T$ and

$$\alpha(k+1) = \frac{1}{n}\mathbf{1}^T x(k+1) = \frac{1}{n}(\mathbf{1}^T P)x(k) = \alpha(k)$$

which implies α is invariant in at iteration k. Let us define the *disagreement vector* [70]

$$\delta = x - \alpha\mathbf{1}, \tag{19}$$

and note that $\sum_i \delta_i = 0$, or $\mathbf{1}^T \delta = 0$. The consensus algorithms result in the following *disagreement dynamics*

$$\begin{aligned} &\text{CT: } \dot{\delta}(t) = -L\delta(t), \\ &\text{DT: } \delta(k+1) = P\delta(k). \end{aligned} \tag{20}$$

Based on the following lemma, one can readily show that $\Phi(\delta) = \delta^T \delta$ is a valid Lyapunov function for the CT system that quantifies the collective disagreement in the network.

Theorem 3. *(algebraic connectivity of digraphs) Let G be a balanced digraph (or undirected graph) with Laplacian L with a symmetric part $L_s = (L + L^T)/2$ and Perron matrix P with $P_s = (P + P^T)/2$. Then*

i) $\lambda_2 = \min_{\mathbf{1}^T \delta = 0} \dfrac{\delta^T L \delta}{\delta^T \delta}$ *with $\lambda_2 = \lambda_2(L_s)$, i.e. $\delta^T L\delta \geq \lambda_2 \|\delta\|^2$ for all disagreement vectors δ;*

ii) $\mu_2 = \max_{\mathbf{1}^T \delta = 0} \dfrac{\delta^T P \delta}{\delta^T \delta}$ *with $\mu_2 = 1 - \epsilon\lambda_2$, i.e. $\delta^T P\delta \leq \mu_2 \|\delta\|^2$ for all disagreement vectors δ.*

Proof. Since G is a balanced digraph, $\mathbf{1}^T L = 0$ and $L\mathbf{1} = 0$. This implies that L_s is a valid Laplacian matrix because of $L_s\mathbf{1} = (L\mathbf{1} + L^T\mathbf{1})/2 = 0$. Similarly, P_s is a valid Perron matrix which is a non-negative doubly stochastic matrix. Part i) follows from a special case of Courant-Fisher theorem [35] for a symmetric matrix L_s due to

$$\min_{\mathbf{1}^T \delta = 0} \frac{\delta^T L \delta}{\delta^T \delta} = \min_{\mathbf{1}^T \delta = 0} \frac{\delta^T L_s \delta}{\delta^T \delta} = \lambda_2(L_s).$$

To show part ii), note that for a disagreement vector δ satisfying $\mathbf{1}^T \delta = 0$, we have

$$\max_{\delta} \frac{\delta^T P \delta}{\delta^T \delta} = \max_{\delta} \frac{\delta^T P \delta}{\delta^T \delta} = \max_{\delta} \frac{\delta^T \delta - \epsilon\delta^T L\delta}{\delta^T \delta} = 1 - \epsilon\min_{\delta} \frac{\delta^T L\delta}{\delta^T \delta} = 1 - \epsilon\lambda_2(L_s) = \mu_2(P_s) \tag{21}$$

Corollary 1. *A continuous-time consensus is globally exponentially reached with a speed that is faster or equal to $\lambda_2 = \lambda_2(L_s)$ with $L_s = (L + L^T)/2$ for a strongly connected and balanced directed network.*

Proof. For CT consensus, we have

$$\dot{\Phi} = -2\delta^T L\delta \leq -2\lambda_2\delta^T \delta = -2\lambda_2\Phi.$$

Therefore, $\Phi(\delta) = \|\delta\|^2$ exponentially vanishes with a speed that is at least $2\lambda_2$. Since $\|\delta\| = \Phi^{1/2}$, the norm of the *disagreement vector exponentially vanishes* with a speed of at least λ_2.   □

Recently, Olfati-Saber [65] has shown that quasi-random *small-world networks* have extremely large λ_2 values compared to regular networks with nearest neighbor communication such as the one in Fig. 2(a). For example for a network, with $n = 1000$ nodes and uniform degree $d_i = 10, \forall i$, the algebraic connectivity of a small-world network can become more than 1500 times of the λ_2 of a regular network [65].

According to Theorem 3, μ_2 is the *second largest eigenvalue* of P_s—the symmetric part of the Perron matrix P. The speed of convergence of the iterative consensus algorithm is provided in the following result:

Corollary 2. *A discrete-time consensus is globally exponentially reached with a speed that is faster or equal to $\mu_2 = 1 - \epsilon\lambda_2(L)$ for a connected undirected network.*

Proof. Let $\Phi(k) = \delta(k)^T\delta(k)$ be a candidate Lyapunov function for the discrete-time disagreement dynamics of $\delta(k+1) = P\delta(k)$. For an undirected graph $P = P^T$ and all eigenvalues of P are real. Calculating $\Phi(k+1)$, one gets

$$\Phi(k + 1) = \delta(k + 1)^T\delta(k + 1)$$
$$= \|P\delta(k)\|^2 \le \mu_2^2\|\delta(k)\|^2$$
$$= \mu_2^2\Phi_k$$

with $0 < \mu_2 < 1$ due to the fact that P is primitive. Clearly, $\|\delta(k)\|$ exponentially vanishes with a speed faster or equal to μ_2.   □

Remark 4. The proof of Corollary 2 for balanced digraphs is rather detailed and beyond the scope of this paper.

2.5 Alternative Forms of Consensus Algorithms

In the context of formation control for a network of multiple vehicles, Fax and Murray [28] introduced the following version of a Laplacian-based system on a graph G with 0-1 weights:

$$\dot{x}_i = \frac{1}{|N_i|} \sum_{j \in N_i} (x_j - x_i) \qquad (22)$$

This is a special case of a consensus algorithm on a graph G^* with adjacency elements $a_{ij} = 1/|N_i| = 1/d_i$ for $j \in N_i$ and zero for $j \notin N_i$. According to this form, $d_i = \sum_{j \ne i} a_{ij} = 1$ for all i that means the degree matrix of G^* is $D^* = I$ and its adjacency matrix is $A^* = D^{-1}A$ provided that all nodes have nonzero degrees (e.g. for connected graphs/digraphs). In graph theory literature, A^* is called *normalized adjacency* matrix. Let Q be the key matrix in the dynamics of (22), i.e. $\dot{x} = -Qx$. Then, an alternative form of graph Laplacian is

$$Q = I - D^{-1}A. \qquad (23)$$

This is identical to the standard Laplacian of the weighted graph G^* due to $L^* = D^* - A^* = I - D^{-1}A$. The convergence analysis of this algorithm is identical to the consensus algorithm presented earlier. The Perron matrix associated with Q is in the form $P = I - \epsilon L^*$ with $0 < \epsilon < 1$. In explicit form, this gives the following iterative consensus algorithm

$$x(k + 1) = [(1 - \epsilon)I + \epsilon D^{-1}A]x(k).$$

The aforementioned algorithm for $\epsilon = 1$ takes a rather simple form $x(k+1) = D^{-1}Ax(k)$ that does not converge for digraphs such as cycles of length n. Therefore, this discretization with $\epsilon = 1$ is invalid. Interestingly, the Markov process

$$\pi(k + 1) = \pi(k)P \qquad (24)$$

with transition probability matrix $P = D^{-1}A$ is known as the process of *random walks on a graph* [49] in graph theory and computer science literature with close connections to gossip-based consensus algorithms [11].

Keep in mind that based on algorithm (22), if graph G is undirected (or balanced), the quantity

$$\alpha = (\sum_i d_i x_i)/(\sum_i d_i)$$

is invariant in time and a weighted-average consensus is asymptotically reached. The weighting $w_i = d_i/(\sum_i d_i)$ is specified by node degree $d_i = |N_i|$. Only for regular networks (i.e. $d_1 = d_2 = \cdots = d_n$), (22) solves an average-consensus problem. This is a rather restrictive condition because most networks are not regular.

Another popular algorithm proposed in [38] (also used in [59, 74]) is the following discrete-time consensus algorithm for undirected networks:

$$x_i(k+1) = \frac{1}{1+|N_i|}(x_i(k) + \sum_{j \in N_i} x_j(k)) \tag{25}$$

which can be expressed as

$$x(k+1) = (I+D)^{-1}(I+A)x(k).$$

Note that the stochastic Perron matrix $P = (I+D)^{-1}(I+A)$ is obtained from the following normalized Laplacian matrix with $\epsilon = 1$

$$Q_l = I - (I+D)^{-1}(I+A) \tag{26}$$

This Laplacian is a modification of (23) and has the drawback that it does not solve average-consensus problem for general undirected networks.

Now, we demonstrate that algorithm (25) is equivalent to (23) (and thus a special case of (7)). Let G be a graph with adjacency matrix A and no self-loops, i.e. $a_{ii} = 0, \forall i$. Then, the new adjacency matrix $A_l = I + A$ corresponds to a graph G_l that is obtained from G by adding n self-loops with unit weights ($a_{ii} = 1, \forall i$). As a result, the corresponding degree matrix of G_l is $D_l = I + D$. Thus, the normalized Laplacian of G_l in (26) is $Q_l = I - D_l^{-1}A_l$. In other words, the algorithm proposed by Jadbabaie *et al.* is identical to the algorithm of Fax and Murray for a graph with n self-loops. In both cases $\epsilon = 1$ is used to obtain the stochastic non-negative matrix P.

Remark 5. A undirected cycle is not a counterexample for discretization of $\dot{x} = -Q_l x$ with $\epsilon = 1$. Since the Perron matrix $P_l = (I+D)^{-1}(I+A)$ is symmetric and primitive.

Example 2. In this example, we clarify that why $P = D^{-1}A$ can be an unstable matrix for a connected graph G, whereas $P_l = (I+D)^{-1}(I+A)$ remains stable for the same exact graph. for doing so, let us consider a bipartite graph G with $n = 2m$ nodes and adjacency matrix

$$A = \begin{bmatrix} 0_m & J_m \\ J_m & 0_m \end{bmatrix} \tag{27}$$

where 0_m and J_m denote the $m \times m$ matrices of zeros and ones, respectively. Note that $D = mI_n$ and $P = D^{-1}A = \frac{1}{m}A$. On the other hand, the Perron matrix of G with n self-loops is

$$P_l = (I_n + D)^{-1}(I_n + A) = \frac{1}{m+1}\begin{bmatrix} I_m & J_m \\ J_m & I_m \end{bmatrix}$$

Let $v = \mathbf{1}_{2m}$ be the vector of ones with $2m$ elements and $w = \text{col}(\mathbf{1}_m, -\mathbf{1}_m)$. Both v and w are eigenvectors of P associated with eigenvalues 1 and -1, respectively, due to $Pv = v$ and $Pw = -w$. This proves that P is *not a primitive matrix* and the limit $\lim_{k \to \infty} P^k$ does not exist (since P has two eigenvalues with modulus 1).

In contrast, P_l does not suffer from this problem because of the n nonzero diagonal elements. Again, v is an eigenvector of P_l associated with the eigenvalue 1, but $P_l w = -(m-1)/(m+1)w$ and due to $(m-1)/(m+1) < 1$ for all $m \geq 1$, -1 is no longer an eigenvalue of P_l.

Table 2. Forms of Laplacians

Source	Laplacian L	Perron P	ϵ
[70, 76]	$D - A$	$I - \epsilon L$	$(0, \Delta^{-1})$
[28]	$I - D^{-1}A$	$D^{-1}A$	1
[38, 59, 74]	$I - (I + D)^{-1}(I + A)$	$(I + D)^{-1}(I + A)$	1

Table 2 summarizes three types of graph Laplacians used in systems and control theory. The alternative forms of Laplacians in the second and third rows of Table 2 are both special cases of $L = D - A$ that is widely used as the standard definition of Laplacian in algebraic graph theory [32].

The algorithms in all three cases are in two forms:

$$\dot{x} = -Lx, \tag{28}$$

$$x(k + 1) = Px(k). \tag{29}$$

Based on Example 2, the choice of the discrete-time consensus algorithm is not arbitrary. Only the first and the third row of Table 2 guarantee stability of a discrete-time linear system for all possible connected networks. The second type requires a further analysis to verify whether P is stable, or not.

2.6 Weighted-Average Consensus

The choice of the Laplacian for the continuous-time consensus depends on the specific application of interest. In cases that reaching an average-consensus is desired, only $L = D - A$ can be used. In case of *weighted-average consensus* with a desired weighting vector $\gamma = (\gamma_1, \ldots, \gamma_n)$, the following algorithms can be used

$$K\dot{x} = -Lx \tag{30}$$

with $K = \text{diag}(\gamma_1, \ldots, \gamma_n)$ and $L = D - A$. This is equivalent to a nodes with a *variable rate of integration* based on the protocol

$$\gamma_i \dot{x}_i = \sum_{j \in N_i} a_{ij}(x_j - x_i).$$

In the special case the weighting is proportional to the node degrees, or $K = D$, one obtains the second type of Laplacian in Table 2, or $\dot{x} = -D^{-1}Lx = -(I - D^{-1}A)x$.

2.7 Consensus under Communication Time-Delays

Suppose that agent i receives a message sent by its neighbor j after a time-delay of τ. This is equivalent to a network with a uniform one-hop communication time-delay. The following consensus algorithm

$$\dot{x}_i(t) = \sum_{j \in N_i} a_{ij}(x_j(t - \tau) - x_i(t - \tau)) \tag{31}$$

was proposed in [70] to reach an average-consensus for undirected graphs G.

Remark 6. Keep in mind that the algorithm

$$\dot{x}_i(t) = \sum_{j \in N_i} a_{ij}(x_j(t - \tau) - x_i(t)) \tag{32}$$

does not preserve the average $\bar{x}(t) = (1/n)\sum_i x_i(t)$ in time for a general graph. The same is true when the graph in (31) is a general digraph. It turns out that for balanced digraphs with 0-1 weights, $\bar{x}(t)$ is an invariant quantity along the solutions of (31).

The collective dynamics of the network can be expressed as

$$\dot{x}(t) = -Lx(t - \tau).$$

Rewriting this equation after taking Laplace transform of both sides, we get

$$X(s) = \frac{H(s)}{s}x(0) \tag{33}$$

with a proper MIMO transfer function $H(s) = (I_n + \frac{1}{s}\exp(-s\tau)L)^{-1}$. One can use Nyquist criterion to verify the stability of $H(s)$. A similar criterion for stability of formations was introduced by Fax and Murray [28]. The following theorem provides an upper bound on the time-delay such that stability of the network dynamics is maintained in presence of time-delays.

Theorem 4. *(Olfati-Saber and Murray, 2004) The algorithm in* (31) *asymptotically solves the average-consensus problem with a uniform one-hop time-delay τ for all initial states if and only if $0 \le \tau < \pi/2\lambda_n$.*

Proof. See the proof of Theorem 10 in [70].   □

Since $\lambda_n < 2\Delta$, a sufficient condition for convergence of the average-consensus algorithm in (31) is that $\tau < \pi/4\Delta$. In other words, there is a trade-off between having a large maximum degree and robustness to time-delays. Networks with hubs (having very large degrees) that are commonly known as *scale-free networks* [4] are fragile to time-delays. In contrast, *random graphs* [22] and *small-world networks* [92, 65] are fairly robust to time-delays since they do not have hubs. In conclusion, construction of engineering networks with nodes that have high degrees is not a good idea for reaching a consensus.

3 Consensus in Dynamic Networks

In many scenarios, networked systems can possess a *dynamic topology* that is time-varying due to node and link failures/creations, packet-loss [83, 33] , asynchronous consensus [34], state-dependence [56], formation reconfiguration [69], evolution [4], and flocking [77, 66].

Networked systems with a dynamic topology are commonly known as *switching networks*. A switching network can be modeled using a dynamic graph $G_{s(t)}$ parameterized with a switching signal $s(t) : \mathbb{R} \to J$ that takes its values in an index set $J = \{1, \ldots, m\}$. The consensus mechanism on a network with a variable topology becomes a linear switching system

$$\dot{x} = -L(G_k)x, \tag{34}$$

with the topology index $k = s(t) \in J$ and a Laplacian of the type $D - A$. The set of topologies of the network is $\Gamma = \{G_1, G_2, \ldots, G_m\}$. First, we assume at any time instance, the network topology is a balanced digraph (or undirected graph) that is strongly connected. Let us denote $\lambda_2((L + L^T)/2)$ by $\lambda_2(G_k)$ for a topology dependent Laplacian $L = L(G_k)$. The following result provides the analysis of average-consensus for dynamic networks with a performance guarantee.

Theorem 5. *(Olfati-Saber and Murray, 2004) Consider a network of agents applying the consensus algorithm in* (34) *with topologies $G_k \in \Gamma$. Suppose every graph in Γ is a balanced digraph that is strongly connected and let $\lambda_2^* = \min_{k \in J} \lambda_2(G_k)$. Then, for any arbitrary switching signal, the agents asymptotically reach an average-consensus for all initial states with a speed faster or equal to λ_2^*. Moreover, $\Phi(\delta) = \delta^T \delta$ is a common Lyapunov function for the collective dynamics of the network.*

Proof. See the proof of Theorem 9 in [70].   □

Note that Γ is a *finite set* with at most $n(n-1)$ elements and this allows the definition of λ_2^*. Moreover, the use of normal Laplacians does not render the average $\bar{x} = (1/n)\sum_i x_i$ invariant in time, unless all graphs in Γ are d-regular (all of their nodes have degree d). This is hardly the case for various applications.

The following result on consensus for switching networks does not require the necessity for connectivity in all time instances and is due to Jadbabaie *et al.* [38]. This weaker form of network connectivity is crucial in analysis of asynchronous consensus with performance guarantees (which is currently an open problem). We need to rephrase the next result for the purpose of compatibility with the notation used in this paper.

Consider the following discrete-time consensus algorithm

$$x_{k+1} = P_{s_k} x_k; \ t = 0, 1, 2, \ldots \tag{35}$$

with $s_k \in J$. Let $\mathcal{P} = \{P_1, \ldots, P_m\}$ denote the set of Perron matrices associated with a finite set of undirected graphs Γ with n self-loops. We say a switching network with the set of topologies Γ is *periodically connected* with a period $N > 1$ if the unions of all graphs over a sequence of intervals $[j, jN)$ for $j = 0, 1, 2, \ldots$ are connected graphs, i.e. $\mathcal{G}_j = \cup_{k=j}^{jN-1} G_{s_k}$ is connected for $j = 0, 1, 2, \ldots$.

Theorem 6. *(Jadbabaie, Lin, and Morse, 2003) Consider the system in* (35) *with $P_{s_k} \in \mathcal{P}$ for $k = 0, 1, 2, \ldots$. Assume the switching network is periodically connected. Then, $\lim_{k\to\infty} x_k = \alpha\mathbf{1}$, or an alignment is asymptotically reached.*

Proof. See the proof of Theorem 2 in [38]. □

The solution of (35) can be explicitly expressed as

$$x_t = (\prod_{k=0}^{t} P_{s_k})x_0 = \Lambda_t x_0$$

with $\Lambda_t = P_{s_t} \cdots P_{s_2} P_{s_1}$. the convergence of the consensus algorithm in (35) depends on whether the infinite product of non-negative stochastic matrices $P_{s_t} \cdots P_{s_2} P_{s_1}$ has a limit. The problem of convergence of infinite product of stochastic matrices has a long history and has been studied by several mathematicians including Wolfowitz [94]. The proof in [38] relies on Wolfowitz's lemma:

Lemma 5. *(Wolfowitz, 1963) Let $\mathcal{P} = \{P_1, P_2, \ldots, P_m\}$ be a finite set of primitive stochastic matrices such that for any sequence of matrices $P_{s_1}, P_{s_2}, \ldots, P_{s_k} \in \mathcal{P}$ with $k \geq 1$, the product $P_{s_k} \cdots P_{s_2} P_{s_1}$ is a primitive matrix. Then, there exists a row vector w such that*

$$\lim_{k\to\infty} P_{s_k} \cdots P_{s_2} P_{s_1} = \mathbf{1}w. \tag{36}$$

According to Wolfowitz's lemma, we get $\lim_{k\to\infty} x_k = \mathbf{1}(wx_0) = \alpha\mathbf{1}$ with $\alpha = wx_0$. The vector w depends on the switching sequence and cannot be determined *a priori*. Thus, an alignment is asymptotically reached and the group decision is an undetermined quantity in the convex hull of all initial states.

Remark 7. Since normal Perron matrices in the form $(I + D)^{-1}(I + A)$ are employed in [38], the agents (in general) do not reach an average-consensus. The use of Perron matrices in the form $I - \epsilon L$ with $0 < \epsilon < 1/(1 + \max_{k\in J} \Delta(G_k))$ resolves this problem.

Recently, an extension of Theorem 6 with connectivity of the union of graphs over an infinite interval has been introduced by Moreau [59] (also, an extension is presented in [74] for weighted graphs). Here, we rephrase a theorem due to Moreau and present it based our notation. First, let us define a less restrictive notion of connectivity of switching networks compared to periodic connectivity. Let Γ be a finite set of undirected graphs with n self-loops. We say a switching networks with topologies in Γ is *ultimately connected* if there exists an initial time k_0 such that over the infinite interval $[k_0, \infty)$ the graph $\mathcal{G} = \cup_{k=k_0}^{\infty} G_{s_k}$ with $s_k \in J$ is connected.

Theorem 7. *(Moreau, 2005) Consider an ultimately connected switching network with undirected topologies in Γ and dynamics (35). Assume $P_{s_k} \in \mathcal{P}$ where $\mathcal{P}$ is the set of normal Perron matrices associated with Γ. Then, a consensus is globally asymptotically reached.*

Proof. See the proof of Proposition 2 in [59].   □

Similarly, the algorithm analyzed in Proposition 2 of [59] does not solve the f-consensus problem. This can be resolved by using the first form of Perron matrices in Table 2. The proof in [59] uses a non-quadratic Lyapunov function and no performance measures for reaching a consensus is presented.

4 Cooperation in Networked Control Systems

This section provides a system-theoretic framework for addressing the problem of cooperative control of networked multi-vehicle systems using distributed controllers. On one hand, a multi-vehicle system represents a collection of decision-making agents that each have limited knowledge of both the environment and the state of the other agents. On the other hand, the vehicles can influence their own state and interact with their environment according to their dynamics which determines their behavior.

The design goal is to execute tasks cooperatively exercising both the *decision-making* and *control* capabilities of the vehicles. In real-life networked multi-vehicle systems, there are a number of limitations including limited sensing capabilities of the vehicles, network bandwidth limitations, as well as interruptions in communications due to packet-loss [83, 33] and physical disruptions to the communication devices of the vehicle.
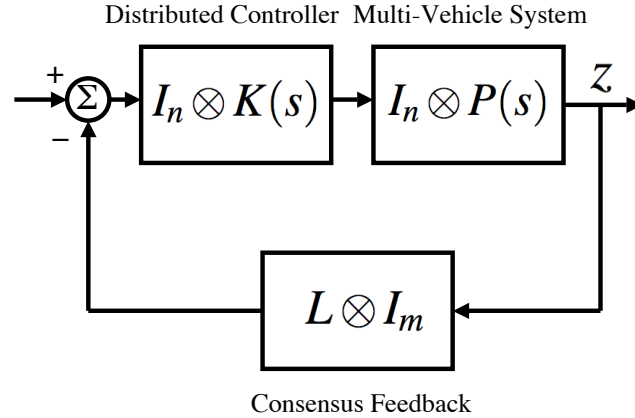

Fig. 3. The block diagram of cooperative and distributed formation control of networked multi-vehicle systems. The Kronecker product $\otimes$ is defined in equation (37).

The system framework we analyze is presented in a schematic form in Fig. 3. The *Kronecker product* $\otimes$ between two matrices $P = [p_{ij}]$ and $Q = [q_{ij}]$ is defined as

$$P \otimes Q = [p_{ij}Q]. \tag{37}$$

This is a block matrix with the ijth block of $p_{ij}Q$.

The dynamics of each vehicle, represented by $P(s)$, is decoupled from the dynamics of other vehicles in the network—thus, the system transfer function $I_n \otimes P(s)$. The output of $P(s)$ represents observable elements of the state of each vehicle. Similarly, the controller of each vehicle, represented by $K(s)$, is decoupled from the controller of others—thus, the controller transfer function $I_n \otimes K(s)$. The coupling occurs through *cooperation* via the *consensus feedback*. Since all vehicles apply the same controller, they form a cooperative team of vehicles with consensus feedback gain matrix $L \otimes I_m$. This cooperation requires sharing of information among vehicles, either through inter-agent sensing, or explicit communication of information.

4.1 Collective Dynamics of Multi-Vehicle Formations

Let us consider a group of n vehicles, whose (identical) linear dynamics are denoted by

$$\dot{x}_i = Ax_i + Bu_i, \tag{38}$$

where $x_i \in \mathbb{R}^m$, $u_i \in \mathbb{R}^p$ are the vehicle states and controls, and $i \in V = \{1, \ldots, n\}$ is the index for the vehicles in the group. Each vehicle receives the following measurements:

$$y_i = C_1 x_i \tag{39}$$
$$z_{ij} = C_2(x_i - x_j), \ j \in N_i, \tag{40}$$

Thus, $y_i \in \mathbb{R}^k$ represents internal state measurements, and $z_{ij} \in \mathbb{R}^l$ represents external state measurements relative to other vehicles. We assume that $N_i \neq \emptyset$, meaning that each vehicle can sense at least one other vehicle. Note that a single vehicle cannot drive all the z_{ij} terms to zero simultaneously; the errors must be fused into a single signal error measurement:

$$z_i = \frac{1}{|N_i|} \sum_{j \in N_i} z_{ij}, \tag{41}$$

where $|N_i|$ is the cardinality of the set N_i. We also define a distributed controller K which maps y_i, z_i to u_i and has internal states $v_i \in \mathbb{R}^s$, represented in state-space form by

$$\begin{aligned} \dot{v}_i &= Fv_i + G_1 y_i + G_2 z_i \\ u_i &= Hv_i + D_1 y_i + D_2 z_i. \end{aligned} \tag{42}$$

Now, we consider the collective system of all n vehicles. For dimensional compatibility, we use the Kronecker product to assemble the matrices governing the formation behavior. The collective dynamics of n vehicles can be represented as follows:

$$\begin{pmatrix} \dot{x} \\ \dot{v} \end{pmatrix} = \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix} \begin{pmatrix} x \\ v \end{pmatrix}. \tag{43}$$

where the M_{ij}'s are block matrices defined as a function of the normalized graph Laplacian L (i.e. the second type in Table 2) and other matrices as follows:

$$\begin{aligned} M_{11} &= I_n \otimes (A + BD_1 C_1) + (I_n \otimes BD_2 C_2)(L \otimes I_m), \\ M_{12} &= I_n \otimes BH, \\ M_{21} &= I_n \otimes G_1 C_1 + (I_n \otimes G_2 C_2)(L \otimes I_m), \\ M_{22} &= I_n \otimes F. \end{aligned}$$

4.2 Stability of Relative Dynamics of Formations

The main stability result on relative-position based formations of networked vehicles is due to Fax and Murray [28] and can be stated as follows:

Theorem 8. *(Fax and Murray, 2004) A local controller K stabilizes the formation dynamics in* (43) *if and only if it stabilizes all the n systems*

$$\begin{aligned}
\dot{x}_i &= Ax_i + Bu_i \\
y_i &= C_1 x_i \\
z_i &= \lambda_i C_2 x_i
\end{aligned} \qquad (44)$$

where $\{\lambda_i\}_{i=1}^n$ is the set of eigenvalues of the normalized graph Laplacian L.

Theorem 8 reveals that the stability of a formation of n identical vehicles can be verified by stability analysis of a single vehicle with the same dynamics and an output that is scaled by the eigenvalues of the (normalized) Laplacian of the network. Note that λ_i may be complex, leading to a complex-valued LTI system in the above formulation. This formalism lends itself to applications of tools from robust control theory [100].

The zero eigenvalue of L can be interpreted as the unobservability of absolute motion of the formation in the measurements z_i. A prudent design strategy is to close an inner loop around y_i such that the internal vehicle dynamics are stable, and then to close an outer loop around z_i which achieves desired formation performance. For the remainder of this section, we concern ourselves solely with the outer loop. Hence, we assume from now on that C_1 is empty and that A has no eigenvalues in the open right half plane. We do not wish to exclude eigenvalues along the $j\omega$ axis because they are characteristic of vehicle systems, representing the directions in which motion is possible. The controller K is also presumed to be stable. If K stabilizes the system in (44) for all λ_i other than the zero eigenvalue, we say that it stabilizes the *relative dynamics of a formation.*

Let us refer to the system from u_i to y_i as P, its transfer function as $P(s)$, and that of the controller from y_i to u_i as $K(s)$. For single-input single-output (SISO) systems, we can state a second version of Theorem 8 which is useful for stability and robustness analysis:

Theorem 9. *(Fax and Murray, 2004) Suppose P is a SISO system. Then K stabilizes the relative dynamics of a formation if and only if the net encirclement of $-1/\lambda_i$ by the Nyquist plot of $-K(s)P(s)$ is zero for all nonzero λ_i.*

The application of the above theorem is demonstrated in Section 5.2.

5 Simulations

In this section, we present the simulation results for three applications of consensus problems in networked systems.

5.1 Consensus in Complex Networks

In this experiment, we demonstrate the speed of convergence of consensus algorithm (7) for three different networks with $n = 100$ nodes in Fig. 4. The initial state is set to $x_i(0) = i$ for $i = 1, \ldots, 100$. In Figs. 4(a) and (c), the network has 300 links and on average each node communicates with $\bar{d} = 6$ neighbors. Apparently, the group with a small-world network topology reaches an average-consensus more than $\lambda_2(G_a)/\lambda_2(G_c) \approx 22$ times faster. To create a regular lattice with comparable algebraic connectivity, every node has to communicate with 20 other nodes on average to gain an algebraic connectivity $\lambda_2(G_e)/\lambda_2(G_a) \approx 1.2$ that is close to that of the small-world network. Of course, the regular network in Fig. 4 (e) has 3.33 times as many links as the small-world network. For further information on small-world networks, we refer the reader to [92, 63, 65].
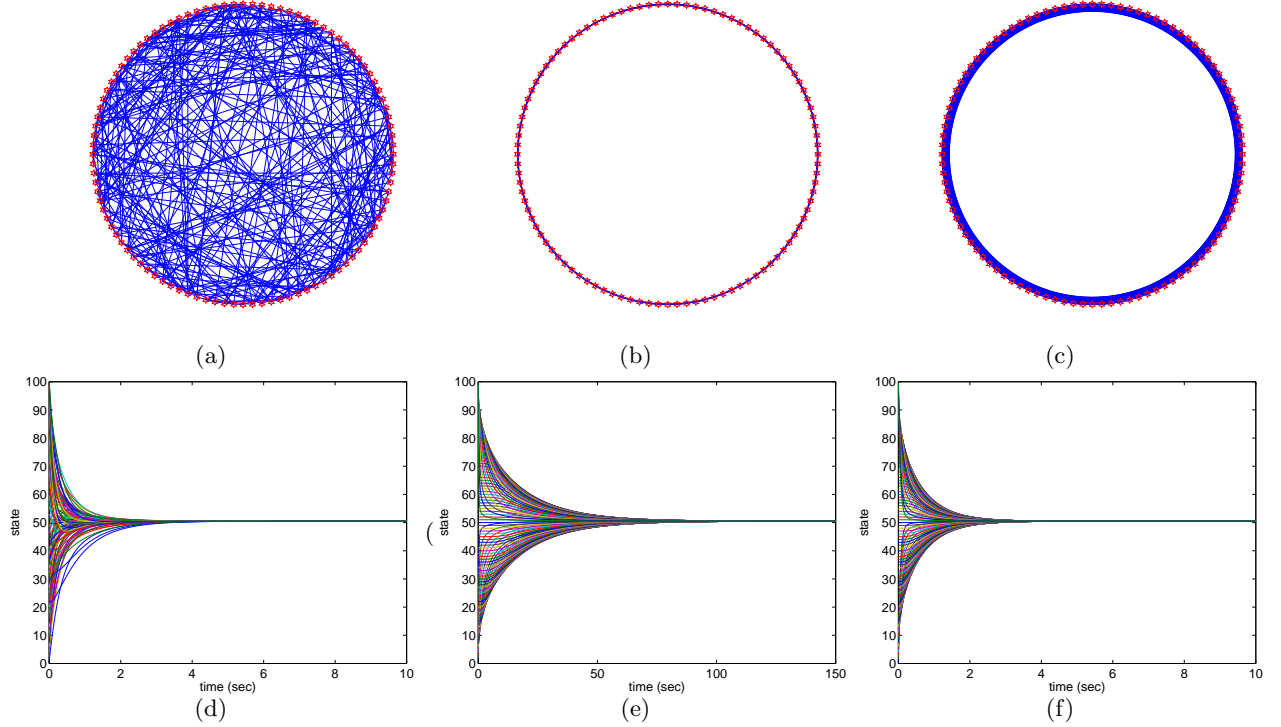
Fig. 4. (a) a small-world with 300 links, (b) a regular lattice with interconnections to $k = 2$ nearest neighbors and 300 links, (c) a regular lattice with interconnections to $k = 10$ nearest neighbors and 1000 links; (d),(e),(f) the state evolution corresponding to networks in (a), (b), and (c), respectively.

5.2 Multi-vehicle Formation Control

Consider a system of the form $P(s) = \frac{e^{-sT}}{s^2}$, modeling a second-order system with time-delay and suppose this system has been stabilized with a proportional-derivative (PD) controller. Fig. 5 shows a formation graph and the Nyquist plot of $K(s)P(s)$ with the location of Laplacian eigenvalues. The "o" locations correspond to the eigenvalues of the graph defined by the solid arcs in Fig. 5, and the '×' locations are for eigenvalues of the graph when the dashed arc is included as well. This example clearly shows the effect the formation has on stability margins. The standard Nyquist plot reveals a system with reasonable stability margins — about 8 dB and 45 degrees. When one accounts for the effects of the formation, however, one sees that for the 'o' formation, the stability margins are substantially degraded, and for the '×' formation, the system is in fact unstable. Interestingly, the formation is rendered unstable when additional information (its position relative to vehicle 6) is used by vehicle 1. This is primarily due to the fact that changing the topology of a network directly effects the location of eigenvalues of the Laplacian matrix. This example clarifies that the stability analysis of formations of networked vehicles with directed switching topology in presence of time-delays is by no means trivial.

6 Conclusions

A theoretical framework was provided for analysis of consensus algorithms for networked multi-agent systems with fixed or dynamic topology and directed information flow. The connections between consensus problems and several applications were discussed that include synchronization of coupled oscillators, flocking, formation control, fast consensus in small-world networks, Markov processes and gossip-based algorithms, load
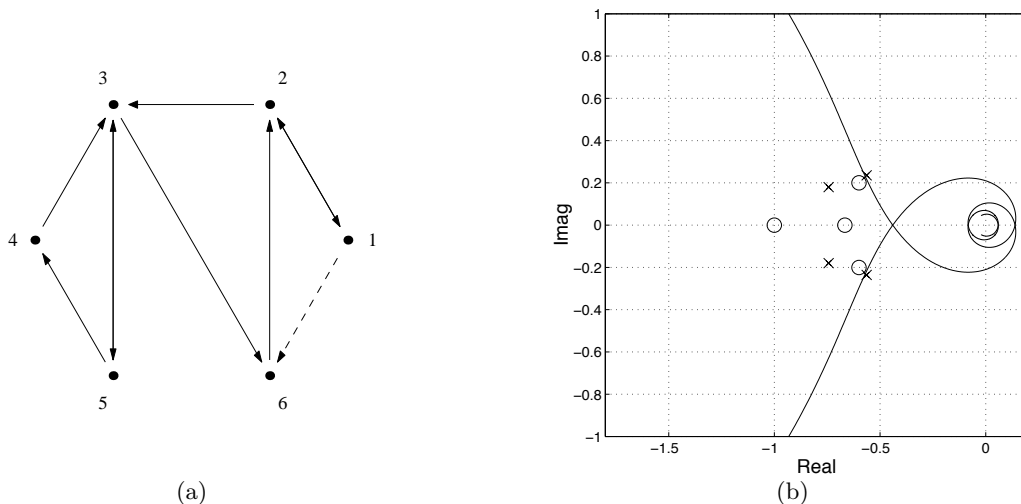
Fig. 5. (a) Interconnection graph of a multi-vehicle formation and (b) the Nyquist plot.

balancing in networks, rendezvous in space, distributed sensor fusion in sensor networks, and belief propagation. The role of "cooperation" in distributed coordination of networked autonomous systems was clarified and the effects of lack of cooperation was demonstrated by an example. It was demonstrated that notions such as graph Laplacians, non-negative stochastic matrices, and algebraic connectivity of graphs and digraphs play an instrumental role in analysis of consensus algorithms. We proved that algorithms introduced by Jadbabaie *et al.* and Fax and Murray are identical for graphs with n self-loops and are both special cases of the consensus algorithm of Olfati-Saber and Murray. The notion of Perron matrices was introduced as the discrete-time counterpart of graph Laplacians in consensus protocols. A number of fundamental spectral properties of Perron matrices were proved. This led to a unified framework for expression and analysis of consensus algorithms in both continuous-time and discrete-time. Simulation results for reaching a consensus in small-worlds versus lattice-type nearest-neighbor graphs and cooperative control of multi-vehicle formations were presented.

Acknowledgments

References

1. H Abelson, D. Allen, D. Coore, C. Hanson, G. Homsy, Jr. Knight, T.F., R. Nagpal, E. Rauch, G. J. Sussman, and R. Weiss. Amorphous computing. *Communications of the ACM*, 43(5):74–82, 2000.
2. M. Alighanbari and J. P. How. Decentralized task assignment for unmanned aerial vehicles. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 5668–5673, Dec. 2005.
3. H. Ando, Y. Oasa, I. Suzuki, and M. Yamashita. Distributed memoryless point convergence algorithm for mobile robots with limited visibility. *IEEE Trans. on Robotics and Automation*, 15(5):818–828, 1999.
4. A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.

5. D. Bauso, L. Giarré, and R. Pesenti. Distributed Consensus in Networks of Dynamic Agents. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 7054–7059, Dec. 2005.

6. J. A. Benediktsson and P. H. Swain. Consensus theoretic classification methods. *IEEE Trans. on Systems, Man and Cybernetics*, 22(4):688–704, 1992.

7. D. P. Bertsekas and J. Tsitsiklis. *Parallell and Distributed Computation*. Prentice-Hall, 1989.

8. P.-A. Bliman and G. Ferrari-Trecate. Average consensus problems in networks of agents with delayed communications. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 7066–7071, Dec. 2005.

9. V. Blondel, J. M. Hendrickx, A. Olshevsky, and J. N. Tsitsiklis. Convergence in multiagent coordination, consensus, and flocking. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 2996–3000, Dec. 2005.

10. V. Borkar and P. Varaiya. Asymptotic agreement in distributed estimation. *IEEE Trans. Automatic Control*, 27:650–655, 1982.

11. S. Boyd, A. Ghosh, and D. Prabhakar, B. Shah. Gossip algorithms: design, analysis and applications. *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '05)*, pages 1653–1664, March 2005.

12. M. Cao, D. A. Spielman, and A. S. Morse. A lower bound on convergence of a distributed network consensus algorithm. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 2356–2361, Dec. 2005.

13. J.-Y. Chen, G. Pandurangan, and D. Xu. Robust computation of aggregates in wireless sensor networks: distributed randomized algorithms and analysis. *Fourth International Symposium on Information Processing in Sensor Networks*, pages 348–355, April 2005.

14. N. Chopra and M. W. Spong. On synchronization of Kuramoto oscillators. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 3916–3922, Dec. 2005.

15. J. Cortés. Achieving coordination tasks in finite time via nonsmooth gradient flows. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 6376–6381, Dec. 2005.

16. J. Cortés, S. Martínez, and F. Bullo. Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions. *IEEE Trans. on Automatic Control*, July 2004. To appear.

17. J. Cortés, S. Martínez, T. Karatas, and F. Bullo. Coverage control for mobile sensing networks. *IEEE Trans. on Robotics and Automation*, 20(2):243–255, April 2004.

18. G. Cybenko. Dynamic load balancing for distributed memory multiprocessors. *Journal of Parallel and Distributed Computing*, 7(2):279–301, Oct. 1989.

19. M. H. DeGroot. Reaching a consensus. *Journal of American Statistical Association*, 69(345):118–121, 1974.

20. D. V. Dimarogonas and K. J. Kyriakopoulos. Formation control and collision avoidance for multi-agent systems and a connection between formation infeasibility and flocking behavior. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 84–89, Dec. 2005.

21. M. Egerstedt and X. Hu. Formation control with virtual leaders and reduced communications. *IEEE Trans. on Robotics and Automation*, 17(6):947–951, 2001.

22. P. Erdös and A. Rényi. On the evolution of random graphs. *Pub. Math. Inst. Hungarian Academy of Science*, 5:17–61, 1960.

23. T. Eren, P. N. Belhumeur, and A. S. Morse. Closing ranks in vehicle formations based on rigidity. *Proceedings of the 41st IEEE Conference on Decision and Control*, pages 2959–2964, Dec. 2002.

24. T. Eren, W. Whiteley, A. S. Morse, P. N. Belhumeur, and B. D. O. Anderson. Sensor and network topologies of formations with direction, bearing and angle information between agents. *Proceedings of the 42nd IEEE Conference on Decision and Control*, pages 3064–3069, Dec. 2003.

25. G. B. Ermentrout and N. Kopell. Frequency plateaus in a chain of weakly coupled oscillator. *SIAM J. Math. Anal.*, 15:215–237, 1984.

26. L. Fang, P. J. Antsaklis, and A. Tzimas. Asynchronous consensus protocols: preliminary results, simulations and open questions. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 2194–2199, Dec. 2005.

27. J. A. Fax. *Optimal and Cooperative Control of Vehicle Formations*. PhD thesis, Control and Dynamical Systems, California Institute of Technology, Pasadena, CA, 2001.

28. J. A. Fax and R. M. Murray. Information flow and cooperative control of vehicle formations. *IEEE Trans. on Automatic Control*, 49(9):1465–1476, Sep. 2004.

29. M. Fiedler. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, 23(98):298–305, 1973.
30. R. Freeman, P. Yang, and K. M. Lynch. Distributed estimation and control of swarm formation statistics. *Proceedings of the 2006 American Control conference*, June 2006.
31. D. Fudenberg and D. K. Levine. *The Theory of Learning in Games*. MIT Press, 1998.
32. C. Godsil and G. Royle. *Algebraic Graph Theory*, volume 207 of *Graduate Texts in Mathematics*. Springer, 2001.
33. V. Gupta, V. Hassibi, and R. M. Murray. On sensor fusion in the presence of packet-dropping communication channels. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 3547–3552, Dec. 2005.
34. Y. Hatano and M. Mesbahi. Agreement over random networks. *IEEE Trans. on Automatic Control*, 50(11):1867–1872, 2005.
35. R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 1987.
36. D. Hristu and K. Morgansen. Limited communication control. *Systems and Control Letters*, 37:193–205, July 1999.
37. J. Hu, M. Prandini, and C. Tomlin. Interesting conjugate points in formation constrained multi-agent coordination. *Proceedings of the 2005 American Control Conference*, pages 1871–1876, June 2005.
38. A. Jadbabaie, J. Lin, and A. S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Trans. on Automatic Control*, 48(6):988–1001, June 2003.
39. A. Jadbabaie, N. Motee, and M. Barahona. On the stability of the Kuramoto model of coupled nonlinear oscillators. *Proceedings of the 2004 American Control Conference*, June 2004.
40. M. Ji and M. Egerstedt. Connectedness preserving distributed coordination control over dynamic graphs. *Proceedings of the 2005 American Control Conference*, pages 93–98, June 2005.
41. D. Kempe, A. Dobra, and J. Gehrke. Gossip-based computation of aggregate information. *Proc. of the 44th Annual IEEE Symposium on Foundations of Computer Science (FOCS '03)*, 8(3):482–491, 2003.
42. T. Keviczky, Borelli, and G. J. F. Balas. A study on decentralized receding horizon control for decoupled systems. *Proceedings of the 2004 American Control Conference*, pages 4921–4926, 2004.
43. Y. Kim and M. Mesbahi. On maximizing the second smallest eigenvalue of state-dependent graph Laplacian. *IEEE Trans. on Automatic Control*, 51(1):116–120, 2006.
44. E. Klavins. Communication complexity of multi-robot systems. *The 5th Int. Workshop on Algorithmic Foundation of Robotics*, Dec. 2002.
45. Y. Kuramoto. *Chemical oscillators, waves, and turbulence*. Springer, Berlin, 1984.
46. J. Lin, A. S. Morse, and B. D. O. Anderson. The multi-agent rendezvous problem. *Proceedings of the 42nd IEEE Conference on Decision and Control*, pages 1508–1513, Dec. 2003.
47. Z. Lin, M. Brouke, and B. Francis. Local control strategies for groups of mobile autonomous agents. *IEEE Trans. on Automatic Control*, 49(4):622–629, 2004.
48. Z. Lin, B. Francis, and M. Maggiore. Necessary and sufficient graphical conditions for formation control of unicyles. *IEEE Trans. on Automatic Control*, 50(1):121–127, 2005.
49. L. Lovász. Random walks on graphs: a survey. In D. Milos, V. T. Sos, and T. Szony, editors, *Combinatorics, Paul Erdos is Eighty*, pages 353–398, Budapest, 1996. János Bolyai Mathematical Society.
50. A. Lubotzky, R. Phillips, and P. Sarnak. Ramanujan graphs. *Combinatorica*, 8(3):261–277, 1988.
51. N. A. Lynch. *Distributed Algorithms*. Morgan Kaufmann Publishers, Inc., 1997.
52. Martínez, F. Bullo, J. Cortés, and E. Frazzoli. On synchronous robotic networks Part I: models, tasks, and complexity notions. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 2847–2852, Dec. 2005.
53. Martínez, F. Bullo, J. Cortés, and E. Frazzoli. On synchronous robotic networks Part II: time complexity of the rendezvous and deployment algorithms. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 8313–8318, Dec. 2005.
54. M. Mehyar, D. Spanos, J. Pongsjapan, S. H. Low, and R. M. Murray. Distributed averaging on asynchronous communication networks. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 7446–7451, Dec. 2005.
55. R. Merris. Laplacian matrices of a graph: a survey. *Linear Algebra and its Applications*, 197:143–176, 1994.
56. M. Mesbahi. On state-dependent dynamic graphs and their controllability properties. *IEEE Trans. on Automatic Control*, 50(3):387–392, 2005.
57. R. E. Mirollo and S. H. Strogatz. Synchronization of pulse-coupled biological oscillators. *SIAM J. Appl. Math.*, 50:1645–1662, 1990.

58. B. Mohar. The Laplacian spectrum of graphs. In Y. Alavi, G. Chartrand, O. Ollermann, and A. Schwenk, editors, *Graph Theory, Combinatorics, and Applications*, pages 871–898, New York, 1991. John Wiley.

59. L. Moreau. Stability of multiagent systems with time-dependent communication links. *IEEE Trans. on Automatic Control*, 50(2):169–182, 2005.

60. N. Moshtagh, A. Jadbabaie, and K. Daniilidis. Distributed geodesic control laws for flocking of nonholonomic agents. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 2835–2841, Dec. 2005.

61. A. Muhammad and M. Egerstedt. Connectivity graphs as models of local interactions. *Journal of Applied Math. and Computation (preprint, submitted)*, 2004.

62. R. Nagpal. Programmable self-assembly using biologically-inpired multiagent control. *Proc. 1st Int'l Conf. Autonomous Agents and Multiagent Systems*, pages 418–425, 2002.

63. M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45:167–256, 2003.

64. R. Olfati-Saber. Distributed Kalman filter with embedded consensus filter. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 8179–8184, Dec. 2005.

65. R. Olfati-Saber. Ultrafast consensus in small-world networks. *Proceedings of the 2005 American Control Conference*, pages 2371–2378, June 2005.

66. R. Olfati-Saber. Flocking for Multi-Agent Dynamic Systems: Algorithms and Theory. *IEEE Trans. on Automatic Control*, 51(3), Mar. 2006.

67. R. Olfati-Saber, E. Franco, E. Frazzoli, and J. S. Shamma. Belief consensus and distributed hypothesis testing in sensor networks. *Workshop on Network Embedded Sensing and Control, Notre Dame University*, Oct. 2005.

68. R. Olfati-Saber and R. M. Murray. Distributed cooperative control of multiple vehicle formations using structural potential functions. *The 15th IFAC World Congress, Barcelona, Spain*, July 2002.

69. R. Olfati-Saber and R. M. Murray. Graph rigidity and distributed formation stabilization of multi-vehicle systems. *41st IEEE Conference on Decision and Control*, 3:2965–2971, 2002.

70. R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Trans. on Automatic Control*, 49(9):1520–1533, Sep. 2004.

71. R. Olfati-Saber and J. S. Shamma. Consensus filters for sensor networks and distributed sensor fusion. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 6698–6703, Dec. 2005.

72. A. Papachristodoulou and A. Jadbabaie. Synchronization in oscillator networks: switching topologies and non-homogeneous delays. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 5692–5697, Dec. 2005.

73. V. M. Preciado and G. C. Verghese. Synchronization in generalized Erd os-Rénye networks of nonlinear oscillators. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 4628–4633, Dec. 2005.

74. W. Ren and R. W. Beard. Consensus seeking in multiagent systems under dynamically changing interaction topologies. *IEEE Trans. on Automatic Control*, 50(5):655–661, 2005.

75. R. O. Saber, W. B. Dunbar, and R. M. Murray. Cooperative control of multi-vehicle systems using cost graphs and optimization. *Proceedings of the 2003 American Control Conference*, pages 2217–2222, June 2003.

76. R. O. Saber and R. M. Murray. Consensus protocols for networks of dynamic agents. *Proceeding of the 2003 American Control Conference*, 2:951–956, 2003.

77. R. O. Saber and R. M. Murray. Flocking with Obstacle Avoidance: Cooperation with Limited Communication in Mobile Networks. *42nd IEEE Conference on Decision and Control*, 2:2022–2028, 2003.

78. V. Saligrama, M. Alanyali, and O. Savas. Asynchronous distributed detection in sensor networks. *Preprint*, 2005.

79. J. Sandhu, M. Mesbahi, and T. Tsukamaki. Relative sensing networks: observability, estimation, and the control structure. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 6400–6405, Dec. 2005.

80. A. V. Savkin. Coordinated collective motion of groups of autonomous mobile robots: analysis of Vicsek's model. *IEEE Trans. on Automatic Control*, 49(6):981–982, June 2004.

81. R. Sepulchre, D. Paley, and N. Leonard. Collective motion and oscillator synchronization. *Proc. Block Island Workshop on Cooperative Control*, Jun 2003.

82. J. S. Shamma and G. Arslan. Dynamic fictitious play, dynamic gradient play, and distributed convergence to Nash equilibria. *IEEE Trans. on Automatic Control*, 50(3):312–327, Mar. 2005.

83. B. Sinopoli, L. Schenato, M. Franceschetti, K. Poola, M. I. Jordan, and S. S. Sastry. Kalman filtering with intermittent observations. *IEEE Trans. on Automatic Control*, 49(9):1453–1464, Sep. 2004.

84. M. Sipser and D. A. Spielman. Expander codes. *IEEE Trans. on Information Theory*, 42(6):1710–1772, Nov. 1996.

85. D. Spanos, R. Olfati-Saber, and R. M. Murray. Dynamic Consensus on Mobile Networks. *The 16th IFAC World Congress, Prague, Czech*, 2005.

86. D. P. Spanos, R. Olfati-Saber, and R. M. Murray. Approximate distributed Kalman filtering in sensor networks with quantifiable performance. *Fourth International Symposium on Information Processing in Sensor Networks*, pages 133–139, April 2005.

87. S. H. Strogatz. From Kuramoto to Crawford: exploring the onset of synchronization in populations of coupled oscillators. *Physica D*, 143:1–20, 2000.

88. H. G. Tanner, G. J. Pappas, and V. Kumar. Leader-to-formation stability. *IEEE Trans. on Robotics and Automation*, 20(3):443–455, June 2004.

89. R. Teo, D. Stipanović, and C. J. Tomlin. Decentralize spacing control of a string of multiple vehicles over lossy datalinks. *Proceedings of the 42nd IEEE Conference on Decision and Control*, pages 682–687, Dec. 2003.

90. J. N. Tsitsiklis and M. Athens. Convergence and asymptotic agreement in distributed decision problems. *IEEE Trans. on Automatic Control*, 29(8):690–696, 1984.

91. T. Vicsek, A. Cziroók, E. Ben-Jacob, I. Cohen, and O. Shochet. Novel type of phase transition in a system of self-deriven particles. *Phys. Rev. Letters*, 75(6):1226–1229, 1995.

92. D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, June 1998.

93. S. C. Weller and N. C. Mann. Assessing rater performance without a "gold standard" using consensus theory. *Medical Decision Making*, 17(1):71–79, 1997.

94. J. Wolfowitz. Products of indecomposable, aperiodic, stochastic matrices. *Proc. American Mathematical Socity*, 15:733–736, 1963.

95. W. Xi, X. Tan, and J. S. Baras. A stochastic algorithm for self-organization of autonomous swarms. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 765–770, Dec. 2005.

96. X. Xi and E. H. Abed. Formation control with virtual leaders and reduced communications. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 1854–1860, Dec. 2005.

97. L. Xiao and S. Boyd. Fast linear iterations for distributed averaging. *Systems & Control Letters*, 52:65–78, 2004.

98. L. Xiao, S. Boyd, and S. Lall. A scheme for asynchronuous distributed sensor fusion based on average consensus. *Fourth International Symposium on Information Processing in Sensor Networks*, April 2005.

99. M. M. Zavlanos and G. J. Pappas. Controlling connectivity of dynamic graphs. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05)*, pages 6388–6393, Dec. 2005.

100. K. Zhou and J. C. Doyle. *Essentials of robust control*. Prentice Hall, 1997.