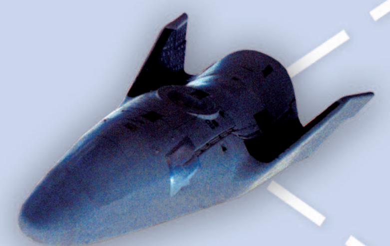
Information Consensus in Multivehicle Cooperative Control

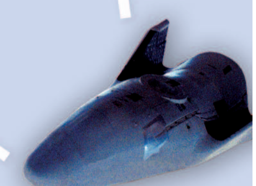
WEI REN, RANDAL W. BEARD, and ELLA M. ATKINS



COLLECTIVE GROUP BEHAVIOR THROUGH LOCAL INTERACTION

The abundance of embedded computational resources in autonomous vehicles enables enhanced operational effectiveness through cooperative teamwork in civilian and military applications. Compared to autonomous vehicles that perform solo missions, greater efficiency and operational capability can be realized from teams of autonomous vehicles operating in a coordinated fashion. Potential applications for multivehicle systems include space-based interferometers, combat, surveillance, and reconnaissance systems, hazardous material handling, and distributed reconfigurable sensor networks. To enable these applications, various cooperative control capabilities need to be developed, including formation control, rendezvous, attitude alignment, flocking, foraging, task and role assignment, payload transport, air traffic control, and cooperative search.

Execution of these capabilities requires that individual vehicles share a consistent view of the objectives and the world. For example, a cooperative rendezvous task requires that each vehicle know the rendezvous point. Information consensus guarantees that vehicles sharing information over a noisy time varying network topology have a consistent view of information that is critical to the coordination task [1]. The instantaneous value of that information is the information state. By necessity, consensus algorithms are designed to be distributed, assuming only neighbor-to-neighbor interaction between vehicles. Vehicles update the value of their information state based on the information states of their neighbors. The goal is to design an update law so that the information states of all of the vehicles in the network converge to a common value. Examples of the information state include a local representation of the center and shape of a formation, the rendezvous time, the length of a perimeter being monitored, the direction of motion for a multivehicle swarm, and the probability that a military target has been destroyed. Consensus algorithms have applications in rendezvous [2]–[4], formation control [5]–[9], flocking [10]–[16], attitude alignment [17]–[19], and sensor networks [20]–[23].



The purpose of this article is to provide a tutorial overview of information consensus in multivehicle cooperative control. Theoretical results regarding consensus-seeking under both time invariant and dynamically changing communication topologies are summarized. Several specific applications of consensus algorithms to multivehicle coordination are described.

A Tutorial on Graph Theory

It is natural to model information exchange among vehicles by means of directed or undirected graphs. A *directed graph* is a pair $(\mathcal{N}, \mathcal{E})$, where $\mathcal{N} = \{1, \ldots, n\}$ is a finite nonempty *node* set and $\mathcal{E} \subset \mathcal{N} \times \mathcal{N}$ is an *edge* set of ordered pairs of nodes, called *edges*. The edge $(i, j) \in \mathcal{E}$ denotes that vehicle j can obtain information from vehicle i, but not necessarily vice versa. Self edges $(i, i) \in \mathcal{E}$ are allowed. For the edge (i, j), i is the *parent node* and j is the *child node*. In contrast to a directed graph, the pairs of nodes in an *undirected graph* are unordered, where the edge (i, j) denotes that vehicles i and j can obtain information from each other. Note that an undirected graph can be viewed as a special case of a directed graph, where an edge (i, j) in the undirected graph corresponds to edges (i, j) and (j, i) in the directed graph. The union of a collection of graphs is a graph whose node and edge sets are the unions of the node and edge sets of the graphs in the collection.

A *directed path* is a sequence of edges in a directed graph of the form $(i_1, i_2), (i_2, i_3), \ldots$. An *undirected path* in an undirected graph is defined analogously. In a directed graph, a *cycle* is a directed path that starts and ends at the same node. The self edge (i, i) denotes a cycle of length 1. A directed graph is *strongly connected* if there is a directed path from every node to every other node. An undirected graph is *connected* if there is an undirected path between every pair of distinct nodes. A *rooted directed tree* is a directed graph in which every node has exactly one parent except for one node, called the *root*, which has no parent and which has a directed path to every other node. Note that a rooted directed tree has no cycle since every edge is oriented away from the root. In the case of undirected graphs, a *tree* is a graph in which every pair of nodes is connected by exactly one undirected path.

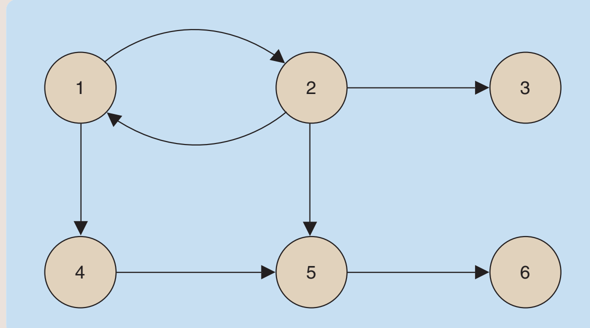


FIGURE S1 Communication graph among six vehicles. An arrow from node i to node j indicates that vehicle j receives information from vehicle i. This directed graph contains two rooted directed spanning trees with root nodes 1 and 2, but is not strongly connected since each node 3, 4, 5, and 6 does not have directed paths to all of the other nodes.

A *subgraph* $(\mathcal{N}_1, \mathcal{E}_1)$ of $(\mathcal{N}, \mathcal{E})$ is a graph such that $\mathcal{N}_1 \subset \mathcal{N}$ and $\mathcal{E}_1 \subset \mathcal{E} \bigcap (\mathcal{N}_1 \times \mathcal{N}_1)$. A *rooted directed spanning tree* $(\mathcal{N}_1, \mathcal{E}_1)$ of the directed graph $(\mathcal{N}, \mathcal{E})$ is a subgraph of $(\mathcal{N}, \mathcal{E})$ such that $(\mathcal{N}_1, \mathcal{E}_1)$ is a rooted directed tree and $\mathcal{N}_1 = \mathcal{N}$. An *undirected spanning tree* of an undirected graph is defined analogously. The graph $(\mathcal{N}, \mathcal{E})$ *has* or *contains* a rooted directed spanning tree if a rooted directed spanning tree is a subgraph of $(\mathcal{N}, \mathcal{E})$. Note that the directed graph $(\mathcal{N}, \mathcal{E})$ has a rooted directed spanning tree if and only if $(\mathcal{N}, \mathcal{E})$ has at least one node with a directed path to all of the other nodes. In the case of undirected graphs, the existence of an undirected spanning tree is equivalent to being connected. However, in the case of directed graphs, the existence of a rooted directed

CONSENSUS ALGORITHMS

The basic idea of a consensus algorithm is to impose similar dynamics on the information states of each vehicle. If the communication network among vehicles allows continuous communication or if the communication bandwidth is sufficiently large, then the information state update of each vehicle is modeled using a differential equation. On the other hand, if the communication data arrive in discrete packets, then the information state update is modeled using a difference equation. This section overviews consensus algorithms in which a scalar information state is updated by each vehicle using a first-order differential equation.

The team's communication topology can be represented by a directed graph (see "A Tutorial on Graph Theory"). For example, Figure 1 shows three different communication topologies for three vehicles. The communication topology may be time varying due to vehicle motion or communication dropouts. For example, communication dropouts might occur when a UAV banks away from its neighbor or flies through an urban canyon. The most common continuous consensus algorithm [1], [5], [24]–[26] is given by

$$\dot{x}_i(t) = -\sum_{j=1}^{n} a_{ij}(t)(x_i(t) - x_j(t)), \quad i = 1, \ldots, n, \tag{1}$$

where $a_{ij}(t)$ is the (i, j) entry of the adjacency matrix of the associated communication graph at time t, which is defined in the sidebar "A Tutorial on Graph Theory," and x_i is the information state of the ith vehicle. Setting $a_{ij} = 0$ denotes the fact that vehicle i cannot receive information from vehicle j. A consequence of (1) is that the information state $x_i(t)$ of vehicle i is driven toward the information states of its neighbors. The critical convergence question is, when do the information states of all of the vehicles converge to a common value?

While (1) ensures that the information states of the team come into agreement, it does not dictate a specified spanning tree is a weaker condition than being strongly connected. Figure S1 shows a directed graph that contains more than one rooted directed spanning tree, but is not strongly connected. Nodes 1 and 2 are both roots of rooted directed spanning trees since they both have a directed path to all of the other nodes. However, the graph is not strongly connected since each node 3, 4, 5, and 6 does not have directed paths to all of the other nodes.

The *adjacency* matrix $A = [a_{ij}] \in \mathbb{R}^{n\times n}$ of a directed graph with node set $\mathcal{N} = \{1, \ldots, n\}$ is defined such that a_{ij} is a positive weight if $(j, i) \in \mathcal{E}$, while $a_{ij} = 0$ if $(j, i) \notin \mathcal{E}$. Note that all graphs are weighted. If the weights are not relevant, then a_{ij} is set equal to 1 for all $(j, i) \in \mathcal{E}$. Self edges, where $a_{ii} > 0$, are allowed. A graph is *balanced* if $\sum_{j=1}^{n} a_{ij} = \sum_{j=1}^{n} a_{ji}$ for all i. For an undirected graph, A is symmetric, and thus every undirected graph is balanced.

Define the Laplacian matrix $L = [\ell_{ij}] \in \mathbb{R}^{n\times n}$ of a directed graph as $\ell_{ii} = \sum_{j\neq i} a_{ij}$ and $\ell_{ij} = -a_{ij}$ for all $i \neq j$. Note that if $(j, i) \notin \mathcal{E}$ then $\ell_{ij} = -a_{ij} = 0$. The Laplacian matrix satisfies

$$\ell_{ij} \le 0, \qquad i \neq j, \tag{S1}$$

$$\sum_{j=1}^{n} \ell_{ij} = 0, \qquad i = 1, \ldots, n. \tag{S2}$$

For an undirected graph, L is symmetric. However, for a directed graph, L is not necessarily symmetric. In both the undirected and directed cases, since L has zero row sums, 0 is an eigenvalue of L with the associated eigenvector $\mathbf{1} \triangleq [1, \ldots, 1]^T$, the $n \times 1$ column vector of ones. Note that L is diagonally dominant and has nonnegative diagonal entries. If follows from Gershgorin's disc theorem [S1, p. 344] that, for an undirected graph, all of the nonzero eigenvalues of L are positive (L is positive semidefinite), whereas, for a directed graph, all of the nonzero eigenvalues of L have positive real parts. Therefore, all of the nonzero eigenvalues of $-L$ have negative real parts. For an undirected graph, 0 is a simple eigenvalue of L if and only if the undirected graph is connected [S2, p. 147]. For a directed graph, 0 is a simple eigenvalue of L if the directed graph is strongly connected [5, Proposition 3], although the converse does not hold. For an undirected graph, let $\lambda_i(L)$ be the ith smallest eigenvalue of L with $\lambda_1(L) \le \lambda_2(L) \le \cdots \le \lambda_n(L)$ so that $\lambda_1(L) = 0$. For an undirected graph, $\lambda_2(L)$ is the *algebraic connectivity*, which is positive if and only if the undirected graph is connected [S2, p. 147]. The algebraic connectivity quantifies the convergence rate of consensus algorithms [48].

Given a matrix $S = [s_{ij}] \in \mathbb{R}^{n\times n}$, the *directed graph of S*, denoted by $\Gamma(S)$, is the directed graph with node set $\mathcal{N} = \{1, \ldots, n\}$ such that there is an edge in $\Gamma(S)$ from j to i if and only if $s_{ij} \neq 0$ [S1, page 357]. In other words, the entries of the adjacency matrix satisfy $a_{ij} > 0$ if $s_{ij} \neq 0$ and $a_{ij} = 0$ if $s_{ij} = 0$.

[S1] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge University Press, 1985.

[S2] R. Merris, "Laplacian matrices of graphs: A survey," *Linear Algebra and its Applications*, vol. 197–198, pp. 143–176, 1994.

[S3] J.N. Tsitsiklis, D.P. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *IEEE Trans. Automat. Contr.*, vol. 31, no. 9, pp. 803–812, 1986.

common value. For example, consider a cooperative rendezvous problem where a team of vehicles is tasked to simultaneously arrive at a specified location known to all of the vehicles. Since the rendezvous time is not given and may need to be adjusted in response to pop-up threats or other environmental disturbances, the team needs to come to consensus on the rendezvous time. To do this, each vehicle first creates an information variable x_i that represents the ith vehicle's understanding of the rendezvous time. To initialize its information state, each vehicle determines a time at which it is able to rendezvous with the team and sets $x_i(0)$ to this value. Each team member then communicates with its neighbors and negotiates a team arrival time using the consensus algorithm (1). Onboard controllers then maneuver each vehicle to rendezvous at the negotiated arrival time. When environmental conditions change, individual vehicles may reset their information state and thus cause the negotiation process to resume.

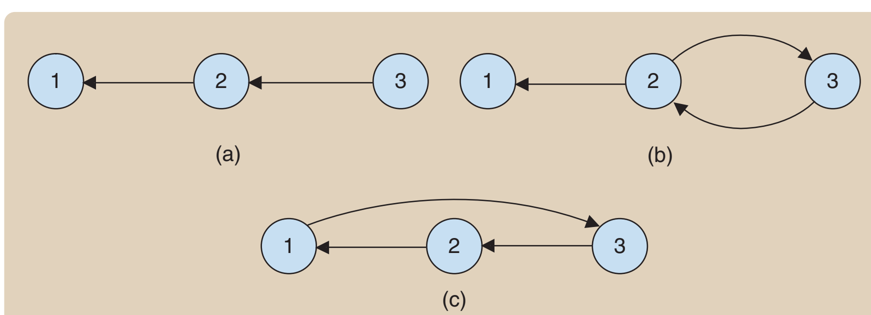


FIGURE 1 Three different communication topologies for three vehicles. Case (c) is strongly connected since there is a directed path between every pair of nodes. However, (a) and (b) are not strongly connected.

Note that (1) does not permit specification of a desired information state. We show in "Equilibrium State Under a Time-invariant Communication Topology" that if the communication topology is fixed and the gains a_{ij} are time invariant, then the common asymptotic value is a linear combination of the initial information states. In general, it is possible to guarantee only that the common value is a convex combination of the initial information states.

Discrete-Time Consensus Algorithms

When communication between vehicles occurs at discrete instants of time, the information state is updated using a difference equation. The most common discrete-time consensus algorithm has the form [S3], [24], [26], [31]

$$x_i[k+1] = \sum_{j=1}^{n} a_{ij}[k] x_j[k], \quad i = 1, \ldots, n, \tag{S3}$$

where k denotes a communication event, $a_{ij}[k]$ is the (i, j) entry of the adjacency matrix of the directed graph that represents the communication topology, with the additional assumption that A is row stochastic and $a_{ii}[k] > 0$ for all $i = 1, \ldots, n$. Intuitively, the information state of each vehicle is updated as the weighted average of its current state and the current states of its neighbors. Note that a vehicle maintains its current information state if it does not exchange information with other vehicles at that event instant. The discrete-time consensus algorithm (S3) is written in matrix form as $x[k+1] = D[k]x[k]$, where $D[k] = [a_{ij}[k]]$ is a row-stochastic matrix. Similar to the continuous case, consensus is *achieved* if, for all $x_i[0]$ and for all $i, j = 1, \ldots, n, |x_i[k] - x_j[k]| \to 0$ as $k \to \infty$.

For the discrete-time consensus algorithm (S3), Gershgorin's disc theorem implies that all of the eigenvalues of D are either in the open unit disk or at 1. If 1 is simple, then $\lim_{k\to\infty} D^k \to \mathbf{1}\nu^T$ as $k \to \infty$ [S1, page 498], where ν is a nonnegative column left eigenvector of D associated with the eigenvalue 1 and satisfies $\nu^T\mathbf{1} = 1$. As a result, $x[k] = D^k x[0] \to \mathbf{1}\nu^T x[0]$ as $k \to \infty$, which implies that, for all i, $x_i[k] \to \nu^T x[0]$ as $k \to \infty$, and thus $|x_i[k] - x_j[k]| \to 0$ as $k \to \infty$.

The Perron-Frobenius theorem (see "A Tutorial on Matrix Theory") implies that $1 = \rho(A)$ is a simple eigenvalue of the row stochastic matrix A if the directed graph $\Gamma(A)$ is strongly connected, or equivalently, if A is irreducible. As in the continuous-time case, this condition is sufficient but not necessary. Furthermore, for the row-stochastic matrix D, $\Gamma(D)$ contains a rooted directed spanning tree if and only if $\lambda = 1$ is a simple eigenvalue of D and is the only eigenvalue of modulus one [26]. As a result, under a time-invariant communication topology, (S3) achieves consensus if and only if either the directed communication topology contains a rooted directed spanning tree or the undirected communication topology is connected [26].

The consensus algorithm (1) is written in matrix form as

$$\dot{x}(t) = -L(t)x(t),$$

where $x = [x_1, \ldots, x_n]^T$ is the information state and $L(t) = [\ell_{ij}(t)] \in \mathbb{R}^{n\times n}$ is the Laplacian of the underlying communication graph (see "A Tutorial on Graph Theory" for additional properties of L). Consensus is *achieved* by a team of vehicles if, for all $x_i(0)$ and all $i, j = 1, \ldots, n$, $|x_i(t) - x_j(t)| \to 0$ as $t \to \infty$. The sidebar "Discrete-Time Consensus Algorithms" addresses the case in which the evolution of the information state is modeled by a difference equation.

CONVERGENCE ANALYSIS OF CONSENSUS ALGORITHMS

Convergence Analysis for Time-Invariant Communication Topologies

In this section, we investigate conditions under which the states of the consensus algorithm (1) converge when the communication topology is time invariant, that is, the Laplacian matrix L is constant. As noted in "A Tutorial on Graph Theory," zero is always an eigenvalue of $-L$, and all of the nonzero eigenvalues of $-L$ have negative real parts. As also noted in the sidebar "A Tutorial on Graph Theory," the column vector $\mathbf{1}$ of ones is an eigenvector of the zero eigenvalue, which implies that span$\{\mathbf{1}\}$ is contained in the kernel of L. It follows that if zero is a simple eigenvalue of L, then $x(t) \to \bar{x}\mathbf{1}$, where $\bar{x}$ is a scalar constant, which implies that $|x_i(t) - x_j(t)| \to 0$ as $t \to \infty$ for all $i, j = 1, \ldots, n$. Convergence analysis therefore focuses on conditions that ensure that zero is a simple eigenvalue of L, since otherwise the kernel of L includes elements that are not in span$\{\mathbf{1}\}$, in which case consensus is not guaranteed.

If the directed graph of L is strongly connected (see "A Tutorial on Graph Theory" for definitions), then zero is a simple eigenvalue of L [5, Proposition 3]. However, this condition is not necessary. For example, consider the Laplacian matrices

$$L_1 = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1.5 & -1.5 \\ 0 & 0 & 0 \end{bmatrix}, \quad L_2 = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1.5 & -1.5 \\ 0 & -2 & 2 \end{bmatrix}, \quad L_3 = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1.5 & -1.5 \\ -2 & 0 & 2 \end{bmatrix} \tag{2}$$

of the directed graphs shown in Figure 1. Although all of the Laplacians in (2) have a simple zero eigenvalue, cases (a) and (b) in Figure 1 are not strongly connected. The common feature is that L_1, L_2, and L_3 all contain a rooted directed

The basic idea of a consensus algorithm is to impose similar dynamics on the information states of each vehicle.

spanning tree. As shown in [7], [8], and [27], zero is a simple eigenvalue of L if and only if the associated directed graph contains a rooted directed spanning tree. This result implies that (1) achieves consensus if and only if the directed communication topology contains a rooted directed spanning tree or the undirected communication topology is connected.

Equilibrium State Under a Time-invariant Communication Topology

We now investigate the consensus equilibrium for the special case in which the communication topology is fixed and the gains a_{ij} are constant. When the directed communication topology contains a rooted directed spanning tree, it follows from [27] that $\lim_{t\to\infty} e^{-Lt} \to \mathbf{1}\nu^T$, where ν is an $n \times 1$ nonnegative column vector satisfying $\sum_{j=1}^{n} \nu_j = 1$. As a result, for each $i = 1, \ldots, n$, $x_i(t) \to \sum_{j=1}^{n} \nu_j x_j(0)$ as $t \to \infty$. In fact, ν is a nonnegative left eigenvector of L corresponding to the zero eigenvalue. That is, the equilibrium state is a weighted average of the initial information states in the network. However, some of the components of ν may be zero, implying that the information states of some of the vehicles do not contribute to the equilibrium.

To illustrate this phenomenon, consider the Laplacians given in (2). It can be verified that, for L_1, $x(t) \to x_3(0)\mathbf{1}$, for L_2, $x(t) \to (0.5714x_2(0) + 0.4286x_3(0))\mathbf{1}$, and, for L_3, $x(t) \to (0.4615x_1(0) + 0.3077x_2(0) + 0.2308x_3(0))\mathbf{1}$. Note that with L_1, the initial information states of vehicles 1 and 2 do not affect the equilibrium. With L_2, the initial information state of vehicle 1 does not affect the equilibrium. However, with L_3, all of the vehicle's initial information states affect the equilibrium. Observing the directed graphs shown in Figure 1, we can see that, for L_1, vehicle 3 is the only agent that can pass information to all of the other vehicles on the team, either directly or indirectly. Similarly, for L_2, both vehicles 2 and 3 can pass information to the entire team, whereas, for L_3, all of the vehicles can pass information to the entire team.

Next, define the nonnegative matrix $M = \max_i \ell_{ii} I_n - L$. Since ν is the nonnegative column left eigenvector of L corresponding to the zero eigenvalue, ν is also the nonnegative left eigenvector of M corresponding to the eigenvalue $\max_i \ell_{ii}$ of M. From Gershgorin's disc theorem it follows that $\rho(M) = \max_i \ell_{ii}$. If the directed communication graph is strongly connected, so is the directed graph of M, which also implies that M is irreducible (see "A Tutorial on Matrix Theory"). By the Perron-Frobenius theorem, if M is irreducible, then ν is positive (see "A Tutorial on Matrix Theory"). Therefore, when the directed communication topology is strongly connected, all of the initial information states contribute to the consensus equilibrium since $\nu_i \neq 0$ for all i. Furthermore, if $\nu_i = 1/n$ for all i, then the consensus equilibrium is the average of initial information states, a condition called *average consensus* [1]. If the directed communication topology is both strongly connected and balanced, then $\mathbf{1}$ is a left eigenvector of L associated with the simple zero eigenvalue. Therefore, as shown in [1], average consensus is achieved if and only if the directed communication topology is both strongly connected

A Tutorial on Matrix Theory

The matrix $A \in \mathbb{R}^{n\times n}$ is *reducible* if either (i) $n = 1$ and $A = 0$, or (ii) $n \geq 2$ and there exists a permutation matrix $P \in \mathbb{R}^{n\times n}$ such that $P^T AP$ is in block upper triangular form. A matrix is *irreducible* if it is not reducible. The matrix A is irreducible if and only if $\Gamma(A)$ is strongly connected [S1, p. 362].

A vector or matrix is *nonnegative* (respectively, *positive*) if all of its entries are nonnegative (respectively, positive). The Perron-Frobenius theorem [S1, p. 508] states that if A is irreducible, then $\rho(A) > 0$ is a simple eigenvalue of A associated with a positive eigenvector, where $\rho(A)$ denotes the spectral radius of A. A square nonnegative matrix is *primitive* if it is irreducible and has exactly one eigenvalue of maximum modulus, which is necessarily positive. A square nonnegative matrix is *row stochastic* if all of its row sums are 1 [S1, p. 526]. Every row-stochastic matrix has 1 as an eigenvalue with associated $n \times 1$ eigenvector $\mathbf{1}$. The spectral radius of a row-stochastic matrix is 1 since 1 is an eigenvalue and Gershgorin's disc theorem implies that all of the eigenvalues are contained in the closed unit disk. The row-stochastic matrix A is *indecomposable and aperiodic* (SIA) if $\lim_{k\to\infty} A^k = \mathbf{1}\nu^T$, where ν is a column vector [29]. Two row-stochastic matrices are of the same *type* if they have zero entries and positive entries in the same locations [29].

Let $A \in \mathbb{R}^{n\times n}$ be nonnegative. If A is primitive, then $\lim_{k\to\infty}[\rho(A)^{-1} A^k] \to w\nu^T$, where $Aw = \rho(A)w$, $A^T\nu = \rho(A)\nu$, $w > 0$, $\nu > 0$, and $w^T\nu = 1$ [S1, p. 516]. In fact, A is primitive if and only if there exists a positive integer m such that A^m is positive [S1, p. 516]. Therefore, since the spectral radius of a row stochastic matrix is 1, if A is row-stochastic and primitive, then $\lim_{k\to\infty} A^k = \mathbf{1}\nu^T$, where $\nu > 0$ is a column vector satisfying $\mathbf{1}^T\nu = 1$.

and balanced. It can be shown that, in the case of undirected communication, average consensus is achieved if and only if the topology is connected [1].

To illustrate these ideas, Figure 2 shows time histories of the information states for two different updates strategies. Figure 2(a) shows the information states for $\dot{x} = -L_3 x$, where L_3 is given in (2). Since the directed graph of L_3 is strongly connected, all of the vehicle's initial conditions contribute to the equilibrium state. However, the equilibrium is not an average consensus since the directed graph is not balanced. In contrast, Figure 2(b) shows the time histories of the information states for $\dot{x} = -\text{diag}\{w\}L_3 x$, where w is the positive column left eigenvector of L_3 satisfying $w^T \mathbf{1} = 1$ and $\text{diag}\{w\}$ is the diagonal matrix whose diagonal entries are given by w. It can be shown that the directed graph $\Gamma(\text{diag}\{w\}L_3)$ is strongly connected and balanced, resulting in average consensus.

In contrast, when the directed communication topology contains a rooted directed spanning tree, the consensus equilibrium is equal to the weighted average of the initial conditions of those vehicles that have a directed path to all of the other vehicles [27]. Requiring a rooted directed spanning tree is less stringent than requiring a strongly connected and balanced graph. However, as shown above, the consensus equilibrium is a function of only the initial information states of those vehicles that have a directed path to all of the other vehicles.

Convergence Analysis for Dynamic Communication Topologies

Information-exchange topologies are often dynamic. For example, communication links among vehicles might be unreliable due to multipath effects and other disturbances. Alternatively, if information is exchanged by means of line-of-sight sensors, the neighbors visible to a vehicle might change over time as when a UAV banks away from its neighbor. Therefore, in this section we investigate conditions under which the consensus algorithms converge under random switching of the communication topologies.

One approach to analyzing switching topologies is to use algebraic graph theory, which associates each graph topology with an algebraic structure of corresponding matrices. Since (1) is linear, its solution can be written as $x(t) = \Phi(t, 0)x(0)$, where $\Phi(t, 0)$ is the transition matrix corresponding to $-L(t)$. In fact, $\Phi(t, 0)$ is a row-stochastic matrix with positive diagonal entries for all $t > 0$ [28]. Consensus is achieved if $\lim_{t\to\infty} \Phi(t, 0) \to \mathbf{1}\mu^T$, where μ is a column vector. It is typical to assume that the communication topology is piecewise constant over finite lengths of time, called the *dwell times*, and that the dwell times are bounded below by a positive constant [24]. In this case, $L(t)$ is piecewise constant with dwell times $\tau_j = t_{j+1} - t_j$, where $t_1, t_2, \ldots$ are the switching instants, and thus consensus is achieved if $\lim_{j\to\infty} e^{-L(t_j)\tau_j} e^{-L(t_{j-1})\tau_{j-1}} \ldots e^{-L(t_0)\tau_0} = \mathbf{1}\mu^T$. Since $e^{-L(t_j)(t-t_j)}$ is a row-stochastic matrix, convergence analysis involves to the study of infinite products of stochastic matrices.

A classical result given in [29] (see also [28]) demonstrates the convergence property of infinite products of SIA matrices (see sidebar "A Tutorial on Matrix Theory"). Specifically, let $\mathcal{S} = \{S_1, S_2, \ldots, S_k\}$ be a finite set of SIA matrices with the property that every finite product $S_{i_j} S_{i_{j-1}} \cdots S_{i_1}$ is SIA. Then, for each infinite sequence $S_{i_1}, S_{i_2}, \ldots$ there exists a column vector ν such that $\lim_{j\to\infty} S_{i_j} S_{i_{j-1}} \cdots S_{i_1} = \mathbf{1}\nu^T$. Since the number of potential communication topologies is finite, the set of matrices $\{S_j \triangleq e^{-L(t_j)(t_{j+1}-t_j)}\}_{j=1}^{\infty}$ is finite if the allowable dwell times $\tau_j = t_{j+1} - t_j$ are drawn from a finite set. Reference [24] shows that these matrices are SIA and uses this result to show that the heading angles of a swarm of vehicles achieve consensus using the nearest-neighbor rules of [30], which is a special case of the discrete consensus algorithm (S3), if there exists an infinite sequence of contiguous, uniformly bounded time intervals, having one of a finite number of different lengths, with the property that across each interval, the union (see "A Tutorial on Graph Theory") of the undirected communication graphs is connected.

Consider, on the other hand, the more realistic assumption that the dwell times are drawn from an infinite but

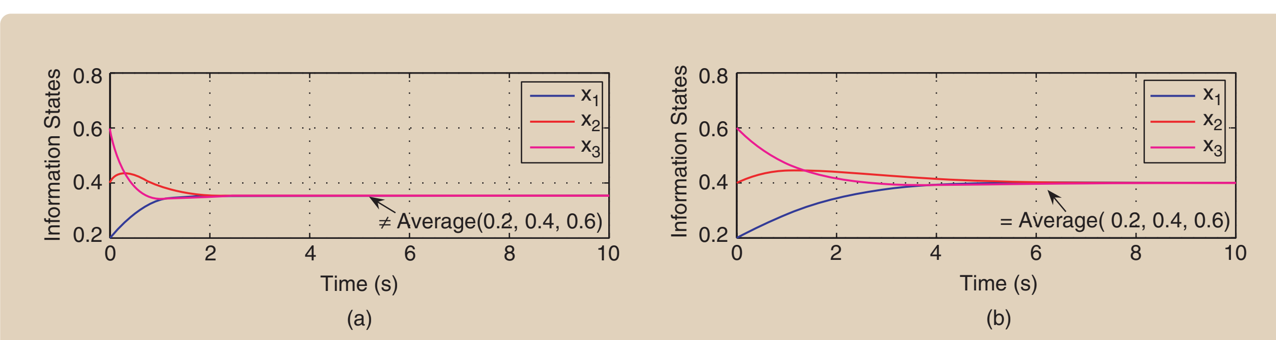


FIGURE 2 Consensus for three vehicles. Plots (a) and (b) correspond to $\dot{x} = -L_3 x$ and $\dot{x} = -\text{diag}\{w\}L_3 x$, respectively. Since 0.4 is the average of the initial states (0.2, 0.4, 0.6), average consensus is achieved in (b), where the graph is strongly connected and balanced, but not in (a), where the graph is only strongly connected.

The consensus equilibrium is a function of only the initial information states of those vehicles that have a directed path to all of the other vehicles.

bounded set or $L(t)$ is piecewise continuous and its nonzero entries are uniformly lower and upper bounded. In this case let $\mathcal{S} = \{S_1, S_2, \dots\}$ be an infinite set of $n \times n$ SIA matrices, let N_t be the number of different types (see "A Tutorial on Graph Theory") of all of the $n \times n$ SIA matrices, and define the matrix function $\lambda(P) = 1 - \min_{i_1, i_2} \sum_j \min(p_{i_1 j}, p_{i_2 j})$. Then, $\lim_{j\to\infty} S_{i_j} S_{i_{j-1}} \cdots S_{i_1} = \mathbf{1}\nu^T$ if there exists a constant $d \in [0, 1)$ such that, for every $W \triangleq S_{k_1} S_{k_2} \cdots S_{k_{N_t+1}}$, it follows that $\lambda(W) \le d$ [29]. It can be shown that this condition is satisfied if there exists an infinite sequence of contiguous, uniformly bounded time intervals, with the property that across each interval, the union of the communication graphs has a rooted directed spanning tree [26], [28].

Lyapunov Analysis of Consensus Algorithms

Nonlinear analysis tools can also be used to study consensus algorithms [31]. For the discrete consensus algorithm (S3), a set-valued function V is defined as $V(x_1, \dots, x_n) = (\mathrm{conv}\{x_1, \dots, x_n\})^n$, where $\mathrm{conv}\{x_1, \dots, x_n\}$ denotes the convex hull of $\{x_1, \dots, x_n\}$, and $X^n \triangleq X \times \cdots \times X$. It is shown in [31] that $V(t_2) \subseteq V(t_1)$ for all $t_2 \ge t_1$, and that $x(t)$ approaches an element of the set span$\{\mathbf{1}\}$, which implies that consensus is reached. Using set-valued Lyapunov theory, [31] shows that the discrete-time consensus algorithm (S3) is uniformly globally attractive with respect to the collection of equilibrium solutions span$\{\mathbf{1}\}$ if and only if there exists $K \ge 0$ such that the union of the communication graphs has a rooted directed spanning tree across each interval of length Kh, where h is the sample time.

For the continuous consensus algorithm (1), [32] considers the Lyapunov candidate $V(x) = \max\{x_1, \dots, x_n\} - \min\{x_1, \dots, x_n\}$. It is shown in [32] that the equilibrium set span$\{\mathbf{1}\}$ is uniformly exponentially stable if there is an interval length $T > 0$ such that, for all t, the directed graph of $-\int_t^{t+T} L(s)ds$ has a rooted directed spanning tree.

As an alternative analysis method, [33] applies nonlinear contraction theory to synchronization and schooling applications, which are related to information consensus. In particular, (1) is analyzed under undirected switching communication topologies, and a convergence result identical to the result given in [24] is derived.

Information consensus is also studied from a stochastic point of view in [34]–[36], which consider a random network, in which the existence of an information channel between a pair of vehicles at each time is probabilistic and independent of other channels, resulting in a time-varying undirected communication topology. For example, the adjacency matrix $A = [a_{ij}] \in \mathbb{R}^{n\times n}$ for an undirected random graph is defined as $a_{ii}(p) = 0$, $a_{ij}(p) = 1$ with probability p, and $a_{ij} = 0$ with probability $1 - p$ for all $i \ne j$. In [34], consensus over an undirected random network is addressed by means of notions from stochastic stability.

Communication Delays and Asynchronous Consensus

When information is exchanged among vehicles through communication, time delays associated with both message transmission and processing after receipt must be considered. Let σ_{ij} denote the time delay for information communicated from vehicle j to reach vehicle i. In this case, (1) is modified as

$$\dot{x}_i = \sum_{j=1}^{n} a_{ij}(t)[x_j(t - \sigma_{ij}) - x_i(t - \sigma_{ij})].$$

In the simplest case, where $\sigma_{ij} = \sigma$ and the communication topology is fixed, undirected, and connected, average consensus is achieved if and only if $0 \le \sigma < (\pi/2\lambda_{\max}(L))$ [1], where L is the Laplacian of the communication graph. See [37], [38] for extensions.

Alternatively, consider the case in which the time delay affects only the information state that is being transmitted so that (1) is modified as

$$\dot{x}_i = \sum_{j=1}^{n} a_{ij}(t)[x_j(t - \sigma_{ij}) - x_i(t)].$$

When $\sigma_{ij} = \sigma$ and the communication topology is directed and switching, the consensus result for switching topologies remains valid for an arbitrary time delay σ [32].

For the discrete consensus algorithm (S3), it is shown in [39] that if consensus is reached under a time-invariant undirected communication topology, then the presence of communication delays does not affect consensus. In addition, the result in [31] is extended to take into account bounded time delays in [40]. Furthermore, [41] shows sufficient conditions for consensus under dynamically changing communication topologies and bounded time-varying communication delays.

More generally, in an asynchronous consensus framework [42]–[46], each vehicle exchanges information asynchronously and updates its state with possibly outdated information from its local neighbors. As a result, heterogenous agents, time-varying communication delays, and packet dropout must all be taken into account in the same asynchronous consensus framework. Reference [46] categorizes several consensus

> **One approach to analyzing switching topologies is to use algebraic graph theory.**

results in the literature according to synchronism, connectivity, and direction of information flow.

SYNTHESIS AND EXTENSIONS OF CONSENSUS ALGORITHMS

Consensus Synthesis

In some applications, consensus algorithms must satisfy given requirements or optimize performance criteria. For example, when a UAV or micro-air vehicle (MAV) swarm consists of hundreds or thousands of vehicles, it might be desirable to solve the fastest distributed linear averaging (FDLA) problem, which is defined as follows [47]. Let $W = [W_{ij}] \in \mathbb{R}^{n\times n}$ be such that $W_{ij} = 0$ if information is not exchanged between vehicle i and vehicle j. Given $x[k+1] = Wx[k]$, find W to minimize

$$r_{asym}(W) = \sup_{x[0]\neq\bar{x}} \lim_{k\to\infty} \left(\frac{\|x[k]-\bar{x}\|}{\|x[0]-\bar{x}\|}\right)^{1/k}$$

subject to the condition that $\lim_{t\to\infty} W^k = (1/n)\mathbf{1}\mathbf{1}^T$, where $\bar{x} = (1/n)\mathbf{1}\mathbf{1}^T x[0]$. In other words, the FDLA problem is to find the weight matrix W that guarantees the fastest convergence to the average consensus value. In contrast to the discrete consensus algorithm (S3), the weights W_{ij} can be negative [47]. With the additional constraint $W_{ij} = W_{ji}$, the FDLA problem reduces to a numerically solvable semidefinite program [47]. A related problem is considered in [48], where an iterative, semidefinite-programming-based approach is developed to maximize the algebraic connectivity of the Laplacian of undirected graphs (see "A Tutorial on Graph Theory") with the motivation that the algebraic connectivity of the Laplacian characterizes the convergence rate of the consensus algorithm.

Another problem is considered in [49], which focuses on designing consensus algorithms in which the information state is updated according to $\dot{x}_i = u_i$, and the information available to the ith agent is given by $y_i = G_i x$, where $x = [x_1, \ldots, x_n]^T$, $y_i \in \mathbb{R}^{m_i}$, and $G_i \in \mathbb{R}^{m_i\times n}$. The information control variable is designed in the form of $u_i = k_i y_i + z_i$, where k_i is a row vector with m_i components and z_i is a scalar.

More generally, consider an interconnected network of n vehicles whose information states are updated according to $\dot{x}_i = \sum_{j=1}^n A_{ij}x_j + B_{1i}w_i + B_{2i}u_i$, $i = 1, \ldots, n$, where $x_i \in \mathbb{R}^n$ denotes the information state, $w_i \in \mathbb{R}^m$ denotes disturbances, and $u_i \in \mathbb{R}^r$ denotes the information control input with $i = 1, \ldots, n$. Letting x, w, and u be column vectors with components x_i, w_i, and u_i, respectively, the dynamics of x are denoted by $\dot{x} = Ax + B_1 w + B_2 u$. Reference [50] focuses on synthesizing a decentralized state feedback control law that guarantees consensus for the closed-loop system without disturbances as well as synthesizing a state-feedback controller that achieves not only consensus but optimal $\mathcal{H}_2$ performance for disturbance attenuation.

Extensions of Consensus Algorithms

The consensus algorithm (1) is extended in various ways in the literature. For example, in [51] an external input is incorporated in (1) so that the information state tracks a time-varying input. In [52], necessary and sufficient conditions are derived so that a collection of systems is controlled by a team leader. An approach based on nonsmooth gradient flows is developed in [53] to guarantee that average consensus is reached in finite time.

The single-integrator consensus algorithm given by (1) is also extended to double-integrator dynamics in [54] and [55] to more naturally model the evolution of physical phenomena, such as a coaxial rotorcraft MAV that can be controlled through gentle maneuvers with a decoupled double-integrator model. For double-integrator dynamics, the consensus algorithm is given by

$$\ddot{x}_i = -\sum_{j=1}^n a_{ij}(t)[(x_i - x_j) + \gamma(\dot{x}_i - \dot{x}_j)],$$

where $\gamma > 0$ denotes the coupling strength between the state derivatives, and both x_i and $\dot{x}_i$ are transmitted between team members. It is shown in [54] that both the communication topology and coupling strength γ affect consensus-seeking in the general case of directed information exchange. To achieve consensus, the directed communication topology must have a rooted directed spanning tree and γ must be sufficiently large.

Related to consensus algorithms are synchronization phenomena arising in systems of coupled nonlinear oscillators. The classical Kuramoto model [56] consists of n coupled oscillators with dynamics given by

$$\dot{\theta}_i = \omega_i + \frac{k}{n}\sum_{j=1}^n \sin(\theta_j - \theta_i), \qquad (3)$$

where θ_i and ω_i are, respectively, the phase and natural frequency of the ith oscillator, and k is the coupling strength. Note that the model (3) assumes full connectivity of the network. The model (3) is generalized in [57] to nearest-neighbor information exchange as

$$\dot{\theta}_i = \omega_i + \frac{k}{n}\sum_{j=1}^n a_{ij}(t)\sin(\theta_j - \theta_i).$$

Connections between phase models of coupled oscillators and kinematic models of self-propelled particle groups are studied in [58]. Analysis and design tools are developed to stabilize the collective motions. Stability of the generalized Kuramoto coupled nonlinear oscillator model is studied in [57], where it is proven that, for couplings above a critical value, all oscillators synchronize given identical and uncertain natural frequencies. Extensions of [57] to a tighter lower bound on the coupling strength are given in [59] for the traditional Kuramoto model with full connectivity. The result in [57] is also extended to account for heterogenous time delays and switching topologies in [60].

Synchronization of coupled oscillators with other nonlinear dynamics are also studied in the literature. As an example, consider a network of n vehicles with information dynamics given by

$$\dot{x}_i = f(x_i, t) + \sum_{j=1}^{n} a_{ij}(t)(x_j - x_i), \tag{4}$$

where $x = [x_1, \ldots, x_n]^T$. In [33] partial contraction theory is applied to derive conditions under which consensus is reached for vehicles with dynamics (4). As another example, [61] studies a dynamical network of n nonlinear oscillators, where the state equation for each node is given by

$$\dot{x}_i = f(x_i) + \gamma \sum_{j=1}^{n} a_{ij}(t)(x_j - x_i),$$

where $x_i \in \mathbb{R}^m$ and $\gamma > 0$ denotes the global coupling strength parameter. It is shown in [61] that the algebraic connectivity of the network Laplacian matrix plays a central role in synchronization.

DESIGN OF COORDINATION STRATEGIES BY MEANS OF CONSENSUS ALGORITHMS

In this section, we briefly describe a few applications of consensus algorithms to multivehicle coordination problems.

Rendezvous Problem

The rendezvous problem requires that a group of vehicles in a network rendezvous at a time or a location determined through team negotiation. Consensus algorithms can be used to perform the negotiation in a way that is robust to environmental disturbances such as nonuniform wind for a team of UAVs. The rendezvous problem for a group of mobile autonomous vehicles is studied in [2] and [3], where synchronous and asynchronous cases are considered. In [2] and [3], agents execute a sequence of stop-and-go maneuvers to rendezvous in a distributed manner without communication between neighbors. A stop-and-go maneuver takes place within a time interval consisting of a sensing period during which neighbors' positions are determined, as well as a maneuvering period during which vehicles move in response to the position of their neighbors.

Figure 3 shows a simple coordination framework for multivehicle rendezvous, where a consensus manager applies distributed consensus algorithms to guarantee that all vehicles reach consensus on a rendezvous objective such as a rendezvous time or rendezvous location. Based on the output of the consensus manager, each vehicle uses a local control law to drive itself to achieve the rendezvous time and/or location. An application of Figure 3 is described in [62], where multiple UAVs are controlled to converge simultaneously on the boundary of a radar detection area to maximize the element of surprise. Team-wide consensus is reached on time-over-target, requiring each vehicle to adjust its velocity to ensure synchronous arrival.

Formation Stabilization

The formation stabilization problem requires that vehicles collectively maintain a prescribed geometric shape. This problem is relatively straightforward in the centralized case, where all team members know the desired shape and location of the formation. On the other hand, in the decentralized formation stabilization problem each vehicle knows the desired formation shape but the location of the formation needs to be negotiated among team members. The information state for this problem includes the center of the formation. Each vehicle initializes its information state by proposing a formation center that does not require it to maneuver into formation. The consensus algorithm is then employed by the team of vehicles to negotiate a formation center known to all members of the team.

In [5], an information flow filter is used to improve stability margins and formation accuracy through propagation of the formation center to all of the vehicles. Formation stabilization for multiple unicycles is studied in [7] using a consensus algorithm to achieve point, line, and general formation patterns. In addition, the simplified pursuit strategy for wheeled-vehicle formations in [63] can be considered a special case of the continuous consensus algorithm (1), where the communication topology is a unidirectional ring.

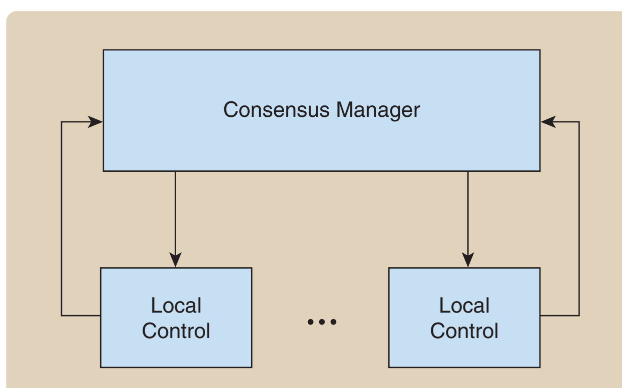


FIGURE 3 A simple coordination framework for multivehicle rendezvous. The consensus manager applies distributed consensus algorithms to guarantee that the team reaches consensus on a rendezvous objective. Based on the output of the consensus manager, each vehicle applies a local control law to achieve the rendezvous objective.

Furthermore, feedback control laws are derived in [8] using relative information between neighboring vehicles to stabilize vehicle formations.

Formation Maneuvering and Flocking

Consensus algorithms can be applied to execute decentralized formation maneuvers. For example, in [6], a class of formation maneuvers is studied where the desired position of each robot, $h_i^d(t)$, is either communicated to the team by a centralized entity or is preprogrammed on each robot. The robots are to maintain a prespecified formation shape even during transients and in response to environmental disturbances. In other words, when one robot slows down or maneuvers to avoid an obstacle, the other robots must maneuver to maintain the formation shape. The inter-vehicle communication network is limited and requires a decentralized approach to maintaining the formation. The mobile robot dynamic model is feedback linearized as the double-integrator system $\ddot{h}_i = u_i$, where h_i denotes the location of a point on the i^{th} robot that is not on the wheel axis, and u_i denotes the control input. The decentralized formation control law is given in [6] as

$$\begin{aligned} u_i = & -K_g\tilde{h}_i - D_g\dot{h}_i - K_f\sum_{j=1}^{n} a_{ij}(\tilde{h}_i - \tilde{h}_j) \\ & - D_f\sum_{j=1}^{n} a_{ij}(\dot{h}_i - \dot{h}_j), \end{aligned} \tag{5}$$

where K_g and K_f are symmetric positive definite, D_g and D_f are symmetric positive semidefinite, and $\tilde{h}_i \triangleq h_i - h_i^d$. In the control law (5), the first two terms guarantee that h_i approaches h_j^d, while the second two terms guarantee that the pairs $\tilde{h}_i, \tilde{h}_j$ and $\dot{h}_i, \dot{h}_j$ reach consensus. If consensus can be reached for each $\tilde{h}_j$, the desired formation shape is guaranteed to be preserved during maneuvers.

A similar approach can be applied to the rigid body attitude dynamics

$$\begin{aligned} \dot{\hat{q}}_i &= -\frac{1}{2}\omega_i \times \hat{q}_i + \frac{1}{2}\bar{q}_i\omega_i, \quad \dot{\bar{q}}_i = -\frac{1}{2}\omega_i \cdot \hat{q}_i, \\ J_i\dot{\omega}_i &= -\omega_i \times (J_i\omega_i) + T_i, \end{aligned}$$

where, for the ith rigid body, $\hat{q}_i \in \mathbb{R}^3$, $\bar{q}_i \in \mathbb{R}$, and $q_i = [\hat{q}_i^T, \bar{q}_i]^T \in \mathbb{R}^4$ is the unit quaternion, that is, the Euler parameters, $\omega_i \in \mathbb{R}^3$ is the angular velocity, and $J_i \in \mathbb{R}^{3\times3}$ and $T_i \in \mathbb{R}^3$ are, respectively, the inertia tensor and the control torque. Defining vec $([\hat{q}, \bar{q}]^T) = \hat{q}$ as the operator that extracts the vector part of a quaternion, the control torque is given by [17]–[19]

$$\begin{aligned} T_i = & -k_G\text{vec}\left(q_i^{d*}q_i\right) - D_G\omega_i - k_S\sum_{j=1}^{n} a_{ij}\text{vec}\,(q_j^*q_i) \\ & - D_S\sum_{j=1}^{n} a_{ij}(\omega_i - \omega_j), \end{aligned} \tag{6}$$

where $k_G > 0$ and $k_S \geq 0$ are scalars, D_G is symmetric positive definite, D_S is symmetric positive semidefinite, q^* is the quaternion inverse [64, p. 465], and q^d is the centrally commanded quaternion. The first two terms in (6) align the rigid body with the prespecified desired orientation q_i^d. The second two terms in (6) are consensus terms that cause the team to maintain attitude alignment during the transients and in response to environmental disturbances [19].

Using biologically observed motions of flocks of birds, [65] defines three rules of flocking and applies them to generate realistic computer animations. The three rules of flocking are collision avoidance, velocity matching, and flock centering. Together these rules maintain the flock in close proximity without collision. Reference [65] motivates the use of similar rules for multivehicle robotic systems [10]–[12]. As an example, consider the vehicle dynamics

$$\dot{r}_i = v_i, \qquad \dot{v}_i = u_i,$$

where r_i and v_i are the position and velocity of vehicle i, respectively, and u_i denotes its input. In [10], the control input u_i is defined as

$$u_i = -\frac{\partial V(r)}{\partial r_i} + \sum_{j=1}^{n} a_{ij}(r)(v_j - v_i) + f_i^\gamma, \tag{7}$$

where the first term is the gradient of a collective potential function $V(r)$, the second term drives the system toward velocity consensus, and the third term incorporates navigational feedback. In (7), the first term guarantees flock centering and collision avoidance among the vehicles, the second term guarantees velocity matching among the vehicles, and the third term achieves a group objective. Equation (7) has been validated for flocking with undirected communication topologies.

CONCLUSIONS

This article has provided a tutorial on consensus strategies and has reviewed recent results from the literature. Current research in information consensus primarily assumes that the consensus equilibrium is a weighted average (see [66] for generalization to a weighted power mean) of the initial information states and therefore constant. This assumption might not be appropriate when each vehicle's information state evolves over time, as occurs in formation control problems, where the formation evolves in two- or three-dimensional space. In addition, recall that (1) ensures only that the information states converge to a common value but does not allow specification of a particular value. While this paradigm is useful for applications such cooperative rendezvous where there is not a single correct value, there are many applications where there is a desired, or correct, information state. For example, a distributed sensor network may be tasked to determine the location of an intruder. Consensus algorithms need to be extended to handle distributed sensor inputs that drive the information state. In this case the convergence issues include both convergence to a common value, as well as convergence of the common state to its correct value.

Experimental implementation of consensus algorithms is a key element of future research. Issues such as disturbance rejection, time delay, communication or sensor noise, and model uncertainties need to be addressed before consensus algorithms find widespread use in cooperative control applications. Recent experimental results [67], [68] provide initial validation of consensus theory applied to mobile robot flocking and cyclic pursuit. Further infusion of consensus algorithms into hardware platforms tasked with realistic missions is feasible and is of paramount importance to enable robust coordinated control for the vast array of emerging networked systems.

REFERENCES

[1] 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, 2004.

[2] J. Lin, A.S. Morse, and B.D.O. Anderson, "The multi-agent rendezvous problem," in *Proc. IEEE Conf. Decision Control*, Maui, Hawaii, 2003, pp. 1508–1513.

[3] J. Lin, A.S. Morse, and B.D.O. Anderson, "The multi-agent rendezvous problem—the asynchronous case," in *Proc. IEEE Conf. Decision Control*, Paradise Island, Bahamas, Dec. 2004, pp. 1926–1931.

[4] S. Martinez, J. Cortes, and F. Bullo, "On robust rendezvous for mobile autonomous agents," in *Proc. IFAC World Congr.*, Prague, Czech Republic, 2005, paper We-A18-TO/3.

[5] J.A. Fax and R.M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE Trans. Automat. Contr.*, vol. 49, no. 9, pp. 1465–1476, 2004.

[6] J.R. Lawton, R.W. Beard, and B. Young, "A decentralized approach to formation maneuvers," *IEEE Trans. Robot. Automat.*, vol. 19, no. 6, pp. 933–941, 2003.

[7] Z. Lin, B. Francis, and M. Maggiore, "Necessary and sufficient graphical conditions for formation control of unicycles," *IEEE Trans. Automat. Contr.*, vol. 50, no. 1, pp. 121–127, 2005.

[8] J.S. Caughman, G. Lafferriere, J.J.P. Veerman, and A. Williams, "Decentralized control of vehicle formations," *Syst. Control Lett.*, vol. 54, no. 9, pp. 899–910, 2005.

[9] W. Ren, "Consensus based formation control strategies for multi-vehicle systems," in *Proc. American Control Conf.*, Minneapolis, MN, June 2006, pp. 4237–4242.

[10] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Trans. Automat. Contr.*, vol. 51, no. 3, pp. 401–420, 2006.

[11] H.G. Tanner, A. Jadbabaie, and G.J. Pappas, "Stable flocking of mobile agents, part i: Fixed topology," in *Proc. IEEE Conf. Decision Control*, Maui, Hawaii, 2003, pp. 2010–2015.

[12] H.G. Tanner, A. Jadbabaie, and G.J. Pappas, "Stable flocking of mobile agents, part ii: Dynamic topology," in *Proc. IEEE Conf. Decision Control*, Maui, Hawaii, 2003, pp. 2016–2021.

[13] J.J.P. Veerman, G. Lafferriere, J.S. Caughman, and A. Williams, "Flocks and formations," *J. Stat. Physics*, vol. 121, pp. 901–936, Dec. 2005.

[14] D.V. Dimarogonas, S.G. Loizou, K.J. Kyriakopoulos, and M.M. Zavlanos, "A feedback stabilization and collision avoidance scheme for multiple independent non-point agents," *Automatica*, vol. 42, no. 2, pp. 229–243, 2006.

[15] N. Moshtagh, A. Jadbabaie, and K. Daniilidis, "Distributed geodesic control laws for flocking of nonholonomic agents," in *Proc. IEEE Conf. Decision Control, European Control Conf.*, Seville, Spain, Dec. 2005, pp. 2835–2840.

[16] D. Lee and M.W. Spong, "Stable flocking of multiple inertial agents on balanced graphs," in *Proc. American Control Conf.*, Minneapolis, MN, June 2006, pp. 2136–2141.

[17] J.R. Lawton and R.W. Beard, "Synchronized multiple spacecraft rotations," *Automatica*, vol. 38, no. 8, pp. 1359–1364, 2002.

[18] W. Ren and R.W. Beard, "Decentralized scheme for spacecraft formation flying via the virtual structure approach," *AIAA J. Guidance, Control, Dynamics*, vol. 27, no. 1, pp. 73–82, 2004.

[19] W. Ren, "Distributed attitude consensus among multiple networked spacecraft," in *Proc. American Control Conf.*, Minneapolis, MN, June 2006, pp. 1760–1765.

[20] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proc. Int. Conf. Information Processing in Sensor Networks*, Los Angeles, CA, Apr. 2005, pp. 63–70.

[21] D.P. Spanos and R.M. Murray, "Distributed sensor fusion using dynamic consensus," in *Proc. IFAC World Congr.*, Prague, Czech Republic, 2005, Paper We-E04-TP/3.

[22] R. Olfati-Saber and J.S. Shamma, "Consensus filters for sensor networks and distributed sensor fusion," in *Proc. IEEE Conf. Decision Control, European Control Conf.*, Seville, Spain, Dec. 2005, pp. 6698–6703.

[23] R.A. Freeman, P. Yang, and K.M. Lynch, "Distributed estimation and control of swarm formation statistics," in *Proc. American Control Conf.*, Minneapolis, MN, June 2006, pp. 749–755.

[24] 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, 2003.

[25] Z. Lin, M. Broucke, and B. Francis, "Local control strategies for groups of mobile autonomous agents," *IEEE Trans. Automat. Contr.*, vol. 49, no. 4, pp. 622–629, 2004.

[26] 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, 2005.

[27] W. Ren, R.W. Beard, and T.W. McLain, "Coordination variables and consensus building in multiple vehicle systems," in *Cooperative Control: A Post-Workshop Volume 2003 Block Island Workshop on Cooperative Control*, V. Kumar, N.E. Leonard, and A.S. Morse, Eds. New York: Springer-Verlag, 2005, pp. 171–188.

[28] W. Ren, R.W. Beard, and D.B. Kingston, "Multi-agent Kalman consensus with relative uncertainty," in *Proc. American Control Conf.*, Portland, OR, 2005, pp. 1865–1870.

[29] J. Wolfowitz, "Products of indecomposable, aperiodic, stochastic matrices," in *Proc. American Mathematical Society*, 1963, vol. 15, pp. 733–736.

[30] T. Vicsek, A. Czirok, E. Ben-Jacob, I. Cohen, and O. Schochet, "Novel type of phase transitions in a system of self-driven particles," *Phys. Rev. Lett.*, vol. 75, no. 6, pp. 1226–1229, 1995.

[31] L. Moreau, "Stability of multi-agent systems with time-dependent communication links," *IEEE Trans. Automat. Contr.*, vol. 50, no. 2, pp. 169–182, 2005.

[32] L. Moreau, "Stability of continuous-time distributed consensus algorithms," in *Proc. IEEE Conf. Decision Control*, Paradise Island, Bahamas, 2004, pp. 3998–4003.

[33] J.-J.E. Slotine and W. Wang, "A study of synchronization and group cooperation using partial contraction theory," in *Cooperative Control: A Post-Workshop Volume 2003 Block Island Workshop on Cooperative Control*, V. Kumar, N. E. Leonard, and A. S. Morse, Eds. New York: Springer-Verlag, 2005, pp. 207–228.

[34] Y. Hatano and M. Mesbahi, "Agreement over random networks," *IEEE Trans. Automat. Contr.*, vol. 50, no. 11, pp. 1867–1872, 2005.

[35] Y. Hatano, A.K. Das, and M. Mesbahi, "Agreement in presence of noise: Pseudogradients on random geometric networks," in *Proc. IEEE Conf. Decision Control, European Control Conf.*, Seville, Spain, Dec. 2005, pp. 6382–6387.

[36] C.W. Wu, "Synchronization and convergence of linear dynamics in random directed networks," *IEEE Trans. Automat. Contr.*, vol. 51, no. 7, pp. 1207–1210, 2006.

[37] P.-A. Bliman and G. Ferrari-Trecate, "Average consensus problems in networks of agents with delayed communications," in *Proc. IEEE Conf. Decision Control, European Control Conf.*, Seville, Spain, Dec. 2005, pp. 7066–7071.

[38] D. Lee and M.W. Spong, "Agreement with non-uniform information delays," in *Proc. American Control Conf.*, Minneapolis, MN, June 2006, pp. 756–761.

[39] H.G. Tanner and D.K. Christodoulakis, "State synchronization in local-interaction networks is robust with respect to time delays," in *Proc. IEEE Conf. Decision Control, European Control Conf.*, Seville, Spain, Dec. 2005, pp. 4945–4950.

[40] D. Angeli and P.-A. Bliman, "Extension of a result by Moreau on stability of leaderless multi-agent systems," in *Proc. IEEE Conf. Decision Control,*

European Control Conf., Seville, Spain, Dec. 2005, pp. 759–764.
[41] F. Xiao and L. Wang, "State consensus for multi-agent systems with switching topologies and time-varying delays," *Int. J. Control*, vol. 79, no. 10, pp. 1277–1284, 2006.
[42] L. Fang and P.J. Antsaklis, "Information consensus of asynchronous discrete-time multi-agent systems," in *Proc. American Control Conf.*, Portland, OR, 2005, pp. 1883–1888.
[43] M. Cao, A.S. Morse, and B.D.O. Anderson, "Coordination of an asynchronous multi-agent system via averaging," in *Proc. IFAC World Congr.*, Prague, Czech Republic, 2005, Paper Mo-A09-TO/5.
[44] V.D. Blondel, J.M. Hendrickx, A. Olshevsky, and J.N. Tsisiklis, "Convergence in multiagent coordination, consensus, and flocking," in *Proc. IEEE Conf. Decision Control, European Control Conf.*, Seville, Spain, Dec. 2005, pp. 2996–3000.
[45] M. Mehyar, D. Spanos, J. Pongsajapan, S.H. Low, and R.M. Murray, "Distributed averaging on asynchronous communication networks," in *Proc. IEEE Conf. Decision Control, European Control Conf.,* Seville, Spain, Dec. 2005, pp. 7446–7451.
[46] L. Fang and P.J. Antsaklis, "On communication requirements for muliagent consensus seeking," in *Networked Embedded Sensing and Control,* P.J. Antsaklis and P. Tabuada, Eds. (Lecture Notes in Control and Information Sciences, vol. 331). Berlin: Springer-Verlag, 2006, pp. 53–67.
[47] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Syst. Control Lett.*, vol. 53, no. 1, pp. 65–78, 2004.
[48] Y. Kim and M. Mesbahi, "On maximizing the second smallest eigenvalue of a state-dependent graph Laplacian," *IEEE Trans. Automat. Contr.*, vol. 51, no. 1, pp. 116–120, 2006.
[49] S. Roy, A. Saberi, and K. Herlugson, "A control-theoretic perspective on the design of distributed agreement protocols," in *Proc. American Control Conf.*, Portland, OR, 2005, pp. 1672–1679.
[50] G.A. de Castro and F. Paganini, "Convex synthesis of controllers for consensus," in *Proc. American Control Conf.*, Boston, MA, 2004, pp. 4933–4938.
[51] D.P. Spanos, R. Olfati-Saber, and R.M. Murray, "Dynamic consensus on mobile networks," in *Proc. IFAC World Congr.*, Prague, Czech Republic, 2005, Paper We-A18-TO/1.
[52] H.G. Tanner, "On the controllability of nearest neighbor interconnections," in *Proc. IEEE Conf. Decision Control*, Paradise Island, Bahamas, Dec. 2004, pp. 2467–2472.
[53] J. Cortes, "Achieving coordination tasks in finite time via nonsmooth gradient flows," in *Proc. IEEE Conf. Decision Control, European Control Conf.,* Seville, Spain, Dec. 2005, pp. 6376–6381.
[54] W. Ren and E. Atkins, "Second-order consensus protocols in multiple vehicle systems with local interactions," in *Proc. AIAA Guidance, Navigation, Control Conf.*, San Francisco, CA, 2005, Paper AIAA-2005-6238.
[55] G. Xie and L. Wang, "Consensus control for a class of networks of dynamic agents: Fixed topology," in *Proc. IEEE Conf. Decision Control, European Control Conf.,* Seville, Spain, Dec. 2005, pp. 96–101.
[56] Y. Kuramoto, *Chemical Oscillations, Waves and Turbulence*. Berlin: Springer-Verlag, 1984.
[57] A. Jadbabaie, N. Motee, and M. Barahona, "On the stability of the Kuramoto model of coupled nonlinear oscillators," in *Proc. American Control Conf.*, Boston, MA, 2004, pp. 4296–4301.
[58] R. Sepulchre, D. Paley, and N. Leonard, "Collective motion and oscillator synchronization," in *Cooperative Control: A Post-Workshop Volume 2003 Block Island Workshop on Cooperative Control,* V. Kumar, N. E. Leonard, and A. S. Morse, Eds. New York: Springer-Verlag, 2005, pp. 189–206.
[59] N. Chopra and M.W. Spong, "On synchronization of Kuramoto oscillators," in *Proc. IEEE Conf. Decision Control, European Control Conf.,* Seville, Spain, Dec. 2005, pp. 3916–3922.
[60] A. Papachristodoulou and A. Jadbabaie, "Synchronization in oscillator networks: Switching topologies and non-homogeneous delays," in *Proc. IEEE Conf. Decision Control, European Control Conf.,* Seville, Spain, Dec. 2005, pp. 5692–5697.
[61] V.M. Preciado and G.C. Verghese, "Synchronization in generalized Erdos-Renyi networks of nonlinear oscillators," in *Proc. IEEE Conf. Decision Control, European Control Conf.,* Seville, Spain, Dec. 2005, pp. 4628–4633.
[62] D.B. Kingston, W. Ren, and R.W. Beard, "Consensus algorithms are input-to-state stable," in *Proc. American Control Conf.*, Portland, OR, 2005, pp. 1686–1690.
[63] J.A. Marshall, M.E. Broucke, and B.A. Francis, "Formations of vehicles in cyclic pursuit," *IEEE Trans. Automat. Contr*., vol. 49, no. 11, pp. 1963–1974, 2004.
[64] M.D. Shuster, "A survey of attitude representations," *J. Astronautical Sci.*, vol. 41, pp. 439–517, Oct.–Dec. 1993.
[65] C.W. Reynolds, "Flocks, herds, and schools: A distributed behavioral model," *Comput. Graph.*, vol. 21, no. 4, pp. 25–34, 1987.
[66] D. Bauso, L. Giarre, and R. Pesenti, "Nonlinear protocols for optimal distributed consensus in networks of dynamic agents," *Syst. Control Lett.*, vol. 55, no. 11, pp. 918–928, 2006.
[67] A. Regmi, R. Sandoval, R. Byrne, H. Tanner, and C. Abdallah, "Experimental implementation of flocking algorithms in wheeled mobile robots," in *Proc. American Control Conf.,* Portland, OR, 2005, pp. 4917–4922.
[68] J.A. Marshall, T. Fung. M.E. Broucke, G.M.T. D'Eleuterio, and B.A. Francis, "Experiments in multirobot coordination,," *Robot. Autonom. Syst.*, vol. 54, no. 3, pp. 265–275, 2006.

AUTHOR INFORMATION

Wei Ren (wren@engineering.usu.edu) is an assistant professor in the Department of Electrical and Computer Engineering at Utah State University. He received his Ph.D. degree in electrical engineering from Brigham Young University, Provo, UT in 2004. From October 2004 to July 2005, he was a research associate in the Department of Aerospace Engineering at the University of Maryland, College Park, MD. His research focuses on cooperative control for multivehicle systems and autonomous control of unmanned vehicles. He is a member of the IEEE Control Systems Society and AIAA. He can be contacted at the Department of Electrical and Computer Engineering, Utah State University, Logan, UT 84322 USA.

Randal W. Beard received the B.S. degree in electrical engineering from the University of Utah, Salt Lake City in 1991, the M.S. degree in electrical engineering in 1993, the M.S. degree in mathematics in 1994, and the Ph.D. degree in electrical engineering in 1995, all from Rensselaer Polytechnic Institute, Troy, NY. Since 1996, he has been with the Electrical and Computer Engineering Department at Brigham Young University, Provo, UT, where he is currently an associate professor. In 1997 and 1998, he was a summer faculty fellow at the Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA. In 2006 and 2007, he was a visiting research fellow at the Air Force Research Laboratory, Munitions Directorate, Eglin AFB, FL. His primary research focus is autonomous control of miniature air vehicles and multivehicle coordination and control. He is currently an associate editor for the *IEEE Control Systems Magazine* and the *Journal of Intelligent and Robotic Systems*.

Ella M. Atkins is an associate professor in the Aerospace Engineering Department at the University of Michigan. She holds B.S. and M.S. degrees in aeronautics and astronautics from MIT and M.S. and Ph.D. degrees in computer science and engineering from the University of Michigan. From 1999 to 2006, she was an assistant professor in the Aerospace Engineering Department at the University of Maryland. She is the author of more than 60 archival and conference publications and serves as an associate editor for the *AIAA Journal of Aerospace Computing, Information, and Communication (JACIC)*. She is chair of the AIAA Intelligent Systems Technical Committee, an associate fellow of AIAA, a member of the IEEE and AAAI, and a private pilot.