

Experimental Validation of Consensus Algorithms for Multivehicle Cooperative Control

Wei Ren, *Member, IEEE*, Haiyang Chao, *Student Member, IEEE*, William Bourgeois, *Student Member, IEEE*, Nathan Sorensen, *Student Member, IEEE*, and YangQuan Chen, *Senior Member, IEEE*

Abstract—In this brief, consensus algorithms are experimentally implemented and validated on a mobile actuator and sensor network platform under directed, possibly switching interaction topologies to explore issues and challenges in distributed multivehicle cooperative control. Distributed consensus algorithms are applied to two target applications including rendezvous and axial alignment. In the rendezvous application, multiple mobile robots simultaneously arrive at a common *a priori* unknown target location determined through team negotiation. In the axial alignment application, multiple mobile robots collectively align their final positions along a line. The experimental results show the effectiveness and robustness of the consensus algorithms even in the presence of platform physical limitations, packet loss, information delay, etc. These experimental results validate the corresponding theoretical results.

Index Terms—Axial alignment, cooperative control, consensus algorithm, mobile actuator and sensor network, multivehicle systems, rendezvous.

I. INTRODUCTION

AUTONOMOUS vehicle systems are expected to find potential applications in military operations, search and rescue, environment monitoring, commercial cleaning, material handling, and homeland security. While single vehicles performing solo missions have yielded some benefits, greater benefits will come from the cooperation of teams of vehicles.

In cooperative control systems, a centralized coordination scheme often relies on the assumption that each member of the team has the ability to communicate to a central location or share information via a fully connected network. As a result, the centralized scheme does not scale well with the number of vehicles. A central location may also result in a catastrophic failure for the overall system due to its single point of failure. In addition, real-world communication topologies are usually not fully connected. In many cases, they depend on the relative position of the vehicles and on other environmental factors and are, therefore, dynamically changing in time. In addition, wireless communication channels are subject to multipath, fading, and dropout. Therefore, decentralized schemes are superior to centralized schemes in terms of robustness and scalability.

Manuscript received January 18, 2007; revised March 31, 2007. Manuscript received in final form July 25, 2007. Recommended by Associate Editor A. Tsourdos. The work of W. Ren was supported in part by the National Science Foundation CAREER Award ECCS-0748287 and the Community/University Research Initiative (2007–2008).

The authors are with the Department of Electrical and Computer Engineering, Utah State University, Logan, UT 84322-4120 USA (e-mail: wren@engineering.usu.edu).

Digital Object Identifier 10.1109/TCST.2007.912239

As an inherently distributed strategy that only requires local neighbor-to-neighbor interaction among vehicles, information consensus has received significant attention in the cooperative control community recently. The basic idea for information consensus is that each vehicle updates its information state based on the information states of its local, possibly time-varying neighbors in such a way that the final information state of each vehicle converges to a common value. This basic idea can be extended to a variety of different scenarios that incorporate group behaviors and dynamics.

Theoretical aspects of consensus algorithms have recently been studied extensively in the literature using algebraic graph theory (e.g., [1]–[7]) and nonlinear mathematical tools (e.g., [8]–[10]). Optimality issues in consensus algorithms are also considered in the literature (e.g., [11]). In addition, information consensus is studied in the context of random networks [12] and asynchronous communication [13]. The consensus algorithms might be applied in the context of cooperative estimation [14]–[16].

Consensus algorithms have applications in rendezvous [17]–[20], formation control [21]–[23], flocking [24]–[26], attitude alignment [27], [28], decentralized task assignment [29], sensor fusion [30]–[32], etc.,. However, in the current literature, most of the research activities in information consensus have focused on theoretical aspects, and most of the applications are demonstrated by means of simulations. Recent efforts in experimental implementation of multirobot flocking and cyclic pursuit are reported, respectively, in [33] and [34], where flocking assumes undirected interaction and the interaction topology for cyclic pursuit forms a unidirectional ring. Infusion of general consensus algorithms into hardware platforms tasked with realistic missions is feasible and systematic experimental implementation and validation of consensus algorithms under a general (possibly directed switching) interaction topologies play an important role in enabling robust coordinated control for the vast array of emerging networked systems.

The main purpose of the current brief is to experimentally implement and validate consensus algorithms under directed, possibly switching interaction topologies to explore issues and challenges in distributed multivehicle cooperative control. The experimental results are expected to provide a preliminary effort toward bridging part of the gap between theory and application in the research of consensus algorithms. In particular, distributed consensus algorithms are applied to two target applications including rendezvous and axial alignment. In the rendezvous application, multiple robots are required to simultaneously arrive at a common *a priori* unknown target location determined through team negotiation. The rendezvous case is directly relevant to unmanned aerial vehicle (UAV) cooperative

timing missions, where multiple UAVs are controlled to converge on the boundary of a radar detection area simultaneously to maximize the element of surprise. In the axial alignment application, multiple robots are required to be evenly distributed on a line with given separation distance through team negotiation. The axial alignment case is directly relevant to sensor deployment and satellite attitude alignment applications. The two target applications are validated on a low-cost mobile actuator and sensor network platform. The experimental results show the effectiveness and robustness of consensus algorithms even in the presence of platform physical limitations, packet loss, information delay, etc. These experimental results validate the corresponding theoretical results.

II. BACKGROUND AND PRELIMINARIES

It is natural to model interaction among vehicles by directed/undirected graphs. A directed graph consists of a pair $(\mathcal{N}, \mathcal{E})$, where \mathcal{N} is a finite nonempty set of nodes and $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ is a set of ordered pairs of nodes, called edges. An edge (i, j) in a directed graph denotes that vehicle j can obtain information from vehicle i , but not necessarily vice versa. In contrast, the pairs of nodes in an undirected graph are unordered, where an edge (i, j) denotes that vehicles i and j can obtain information from one another. Note that an undirected graph can be considered 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. If there is an edge from node i to node j in a directed graph, then i is the parent node and j is the child node.

A directed path is a sequence of edges in a directed graph of the form $(i_1, i_2), (i_2, i_3), \dots$, where $i_j \in \mathcal{N}$. An undirected path in an undirected graph is defined analogously. 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 a path between every distinct pair of nodes. A directed tree is a directed graph, where every node has exactly one parent except for one node, called the root, which has no parent, and the root has a directed path to every other node. In the case of undirected graphs, a tree is a graph in which any two nodes are connected by exactly one path.

A (rooted) directed spanning tree of a directed graph is a directed tree formed by graph edges that connect all of the nodes of the directed graph. A directed graph has or contains a directed spanning tree if there exists a directed spanning tree as a subset of the directed graph. Note that the condition that a directed graph has a directed spanning tree is equivalent to the case where there exists at least one node having a directed path to all of the other nodes. In the case of undirected graphs, having an undirected spanning tree is equivalent to being connected. However, in the case of directed graphs, having a directed spanning tree is a weaker condition than being strongly connected. The union of a group of graphs is a graph with nodes given by the union of the node sets and edges given by the union of the edge sets of the group of graphs.

Suppose that there are n vehicles in the team. The adjacency matrix $A = [a_{ij}] \in \mathbb{R}^{n \times n}$ of a weighted directed graph is defined as $a_{ii} = 0$ and $a_{ij} > 0$ if $(j, i) \in \mathcal{E}$, where $i \neq j$. The

adjacency matrix of a weighted undirected graph is defined analogously except that $a_{ij} = a_{ji}$, $\forall i \neq j$, since $(j, i) \in \mathcal{E}$ implies $(i, j) \in \mathcal{E}$. Let matrix $L = [l_{ij}] \in \mathbb{R}^{n \times n}$ be defined as $l_{ii} = \sum_{j \neq i} a_{ij}$ and $l_{ij} = -a_{ij}$, where $i \neq j$. The matrix L satisfies the following conditions:

$$l_{ij} \leq 0, \quad i \neq j \quad \sum_{j=1}^n l_{ij} = 0, \quad i = 1, \dots, n. \quad (1)$$

For an undirected graph, L is called the Laplacian matrix, which has the property that it is symmetric positive semidefinite. However, L for a directed graph does not have this property. In both the undirected and directed cases, 0 is an eigenvalue of L with the associated eigenvector $\mathbf{1}$, where $\mathbf{1}$ is a column vector of all ones. In the case of undirected graphs, all of the nonzero eigenvalues of L are positive. In the case of directed graphs, all of the nonzero eigenvalues of L have positive real parts from Gershgorin disc theorem [35]. In the case of undirected graphs, 0 is a simple eigenvalue of L if and only if the undirected graph is connected [36]. In addition, the second smallest eigenvalue of L is known as the algebraic connectivity of the undirected graph. In the case of directed graphs, 0 is a simple eigenvalue of L if and only if the directed graph contains a directed spanning tree [37].

III. CONSENSUS ALGORITHMS

Consider vehicles with single-integrator dynamics given by

$$\dot{\xi}_i = u_i, \quad i = 1, \dots, n \quad (2)$$

where $\xi_i \in \mathbb{R}^m$ is the information state of the i th vehicle, and $u_i \in \mathbb{R}^m$ is the control input. A consensus algorithm is proposed in [1]–[4] and [21] as

$$u_i = - \sum_{j \in \mathcal{J}_i(t)} k_{ij}(t)(\xi_i - \xi_j), \quad i = 1, \dots, n \quad (3)$$

where $\mathcal{J}_i(t)$ represents the set of vehicles whose information states are available to vehicle i at time t , and $k_{ij}(t)$ is a positive weighting factor at time t . The objective of (3) is to drive the information state of each vehicle toward the states of its local neighbors. For (3), consensus is *reached* asymptotically among the n vehicles if $\xi_i(t) \rightarrow \xi_j(t)$, $\forall i \neq j$, as $t \rightarrow \infty$ for all $\xi_i(0)$.

Let $L(t) = [l_{ij}(t)] \in \mathbb{R}^{n \times n}$ be defined as

$$\begin{aligned} l_{ij}(t) &= -k_{ij}(t), & j \in \mathcal{J}_i(t) \setminus \{i\} \\ l_{ij}(t) &= 0, & j \notin \mathcal{J}_i(t) \\ l_{ii}(t) &= \sum_{j \in \mathcal{J}_i(t) \setminus \{i\}} k_{ij}(t). \end{aligned} \quad (4)$$

Note that $L(t)$ satisfies (1). Also note that with (3), (2) can be written in matrix form as $\dot{\xi} = -(L(t) \otimes I_m)\xi$, where $\xi = [\xi_1^T, \dots, \xi_n^T]^T$, \otimes denotes the Kronecker product, and I_m is the $m \times m$ identity matrix.

Note that information in this brief is defined differently than in [38]. When applying the consensus algorithm (3) in practice, ξ_i might be an estimate of the system state. In addition, in practice, there may exist a perturbation term in (3) due to uncertainties and external disturbances. As a result, the consensus algorithm (3) becomes

$$u_i = - \sum_{j \in \mathcal{J}_i(t)} k_{ij}(t)(\xi_i - \xi_j) + w_i, \quad i = 1, \dots, n \quad (5)$$

where $w_i \in \mathbb{R}^m$ denotes the perturbation term.

The following two lemmas hold for (5) under time-invariant and switching interaction topologies, respectively.

Lemma 3.1: Suppose that the interaction topology is time invariant and k_{ij} are constant. Under the condition that $w_i = w_j$, $\forall i \neq j$, consensus is reached asymptotically with (5) if and only if the directed interaction topology contains a directed spanning tree. Furthermore, if $w_i = 0$, $\forall i$, and the directed interaction topology contains a directed spanning tree, then $\xi_i \rightarrow \sum_{j=1}^n \nu_j \xi_j(0)$, $\forall i$, where $\nu = [\nu_1, \dots, \nu_n]^T$ is a nonnegative left eigenvector of L , given by (4), associated with the zero eigenvalue, and ν satisfies the condition that $\nu_i > 0$ if vehicle i has a directed path to every other vehicle and $\nu_i = 0$, otherwise, and $\sum_{j=1}^n \nu_j = 1$. In addition, under the condition that the directed interaction topology contains a directed spanning tree, if $\|w_i - w_j\|$ is uniformly bounded, so is $\|\xi_i - \xi_j\|$, $\forall i \neq j$.

Proof: The first two arguments follow [37] directly. For the third argument, let $\xi_{ij} \triangleq \xi_i - \xi_j$ and $w_{ij} \triangleq w_i - w_j$. Also let $\tilde{\xi} = [\xi_{12}^T, \xi_{13}^T, \dots, \xi_{1n}^T]^T$ and $\tilde{w} = [w_{12}^T, w_{13}^T, \dots, w_{1n}^T]^T$. Note that $\xi_{ij} = \xi_{1j} - \xi_{1i}$. With (5), (2) can be written as

$$\dot{\tilde{\xi}} = (\Gamma \otimes I_m) \tilde{\xi} + \tilde{w} \quad (6)$$

where $\Gamma \in \mathbb{R}^{(n-1) \times (n-1)}$ is constant. Note that the first argument of the lemma shows that under the condition that $w_i = w_j$, $\forall i \neq j$, and the directed interaction topology contains a directed spanning tree, consensus is reached asymptotically, which implies that $\tilde{\xi}(t) \rightarrow 0$ asymptotically as $t \rightarrow \infty$. It thus follows that (6) is asymptotically stable under the same condition by noting that $w_i = w_j$, $\forall i \neq j$, implies $\tilde{w} = 0$. Because (6) is a linear time-invariant system, asymptotical stability implies bounded input bounded state. That is, if \tilde{w} is uniformly bounded, so is $\tilde{\xi}$. Equivalently, if $\|w_i - w_j\|$ is uniformly bounded, so is $\|\xi_i - \xi_j\|$, $\forall i \neq j$.

Lemma 3.2: Suppose that $k_{ij}(t) \in [\underline{k}, \bar{k}]$, where \underline{k} and \bar{k} are positive constants. Also suppose that the directed interaction topologies are switching but there exist infinitely many consecutive uniformly bounded time intervals such that the union of the directed interaction topologies across each time interval has a directed spanning tree. With (5), consensus is reached asymptotically under the condition $w_i = 0$. Furthermore, if $\|w_i - w_j\|$ is uniformly bounded, so is $\|\xi_i - \xi_j\|$, $\forall i \neq j$.

Proof: The first argument follows [4] directly. For the second argument, assume that $\xi_i \in \mathbb{R}$ and $w_i \in \mathbb{R}$ for simplicity. However, the proof holds for $\xi_i \in \mathbb{R}^m$ and $w_i \in \mathbb{R}^m$ by introducing the Kronecker product. Let $\xi_{ij} \triangleq \xi_i - \xi_j$, $w_{ij} \triangleq w_i - w_j$, $\tilde{\xi} = [\xi_{12}, \xi_{13}, \dots, \xi_{1n}]^T$, and $\tilde{w} = [w_{12}, w_{13}, \dots, w_{1n}]^T$. With (5), (2) can be written as

$$\dot{\tilde{\xi}} = \Gamma(t) \tilde{\xi} + \tilde{w} \quad (7)$$

where $\Gamma(t) \in \mathbb{R}^{(n-1) \times (n-1)}$ is switching with time. The first argument of the lemma shows that $\tilde{\xi}(t) \rightarrow 0$ uniformly in t_0 as $t \rightarrow \infty$ when $w_i = 0$. However, unlike the proof of Lemma 3.1, for a linear time-varying system like (7), asymptotical stability does not necessarily imply bounded input bounded state. The next statement will show that (7) is in fact *uniformly* asymptotically stable under the condition that $w_i = 0$. With (3), (2) can be written in matrix form as

$$\dot{\xi} = -L(t)\xi \quad (8)$$

where $\xi = [\xi_1, \dots, \xi_n]^T$ and $L(t)$ given by (4) is switching with time. The solution to (8) is given by $\xi(t) = \Phi(t, 0)\xi(0)$, where the transition matrix $\Phi(t, 0) \in \mathbb{R}^{n \times n}$ is a row stochastic matrix as shown in [39, Lemma 3.3]. As a result, it follows that $\xi_i(t) = \sum_{j=1}^n \beta_{ij}(t)\xi_j(0)$, where $\beta_{ij}(t) \geq 0$ and $\sum_{j=1}^n \beta_{ij}(t) = 1$. Therefore, it follows that $\max_j \xi_j(t) \leq \max_j \xi_j(0)$ and $\min_j \xi_j(t) \geq \min_j \xi_j(0)$, which in turn implies that

$$\max_{j \neq 1} |\xi_{1j}(t)| \leq \max_j \xi_j(0) - \min_j \xi_j(0). \quad (9)$$

Let $\underline{j} = \operatorname{argmin}_j \xi_j(0)$ and $\bar{j} = \operatorname{argmax}_j \xi_j(0)$. Note that

$$\begin{aligned} \max_j \xi_j(0) - \min_j \xi_j(0) &= |\xi_{\bar{j}\underline{j}}(0)| \\ &= |\xi_{1\underline{j}}(0) - \xi_{1\bar{j}}(0)| \\ &\leq |\xi_{1\underline{j}}(0)| + |\xi_{1\bar{j}}(0)| \leq 2 \max_{j \neq 1} |\xi_{1j}(0)|. \end{aligned} \quad (10)$$

Combining (9) and (10) gives $\max_{j \neq 1} |\xi_{1j}(t)| \leq 2 \max_{j \neq 1} |\xi_{1j}(0)|$, which implies $\|\tilde{\xi}(t)\|_\infty \leq 2\|\tilde{\xi}(0)\|_\infty$, $\forall t$, when $w_i = 0$. Therefore, (7) is uniformly stable when $w_i = 0$. Combing the previous arguments, it follows that (7) is uniformly asymptotically stable when $w_i = 0$, which in turn implies that (7) is uniformly exponentially stable [40, Th. 6.13]. According to [40, Lemma 12.4], it follows that if $\|\tilde{w}\|$ is uniformly bounded, so is $\|\tilde{\xi}\|$. Equivalently, it follows that if $\|w_i - w_j\|$ is uniformly bounded, so is $\|\xi_i - \xi_j\|$, $\forall i \neq j$.

Note that (3) represents the fundamental form of a consensus algorithm. The algorithm can be extended to achieve different convergence results. For example, (3) can be extended to guarantee that the differences of the information states converge to desired values, i.e., $\xi_i - \xi_j \rightarrow \Delta_{ij}(t)$, where $\Delta_{ij}(t)$ denotes the desired (time-varying) separation between ξ_i and ξ_j . The following algorithm is applied for relative separations:

$$u_i = \dot{\delta}_i - \sum_{j \in \mathcal{J}_i(t)} k_{ij}(t)[(\xi_i - \xi_j) - (\delta_i - \delta_j)], \quad i = 1, \dots, n \quad (11)$$

where $\delta_i - \delta_j$, $\forall i \neq j$, denotes the desired separation between the information states. Note that with appropriate choices of δ_ℓ , $\ell = 1, \dots, n$, the differences of the information states are guaranteed to converge to desired values. The algorithm (11) has applications in formation control, where the team forms a certain formation shape by maintaining relative positions among vehicles. Also note that (3) corresponds to the case where $\Delta_{ij} = 0$, $\forall i \neq j$.

The following corollary holds for relative separations.

Corollary: Under the conditions that k_{ij} are constant and the interaction topology is time invariant, $\xi_i - \xi_j \rightarrow \delta_i - \delta_j$ asymptotically with (11) if and only if the directed interaction topology has a directed spanning tree. In addition, under directed switching interaction topologies, $\xi_i - \xi_j \rightarrow \delta_i - \delta_j$ asymptotically under the conditions of Lemma 3.2.

Proof: With (11), (2) can be written as $\dot{\hat{\xi}}_i = -\sum_{j \in \mathcal{J}_i(t)} k_{ij}(t)(\hat{\xi}_i - \hat{\xi}_j)$, where $\hat{\xi}_i \triangleq \xi_i - \delta_i$, $i = 1, \dots, n$. The rest of the proof then follows the fact that $\hat{\xi}_i \rightarrow \hat{\xi}_j$ is equivalent to $\xi_i - \xi_j \rightarrow \delta_i - \delta_j$.

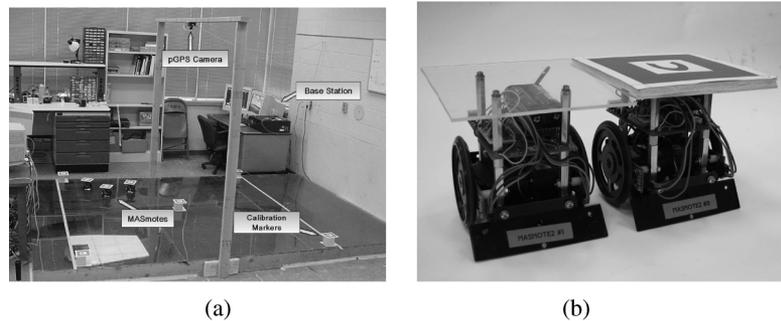


Fig. 1. MASnet experimental platform. (a) MASmote testbed. (b) MASmote robot hardware.

Note that Lemmas 3.1 and 3.2 have shown that the consensus algorithm (3) is robust to perturbations. Also note that similar proofs to those in Lemmas 3.1 and 3.2 can also show that (11) is robust to perturbations. The robustness of the consensus algorithms (3) and (11) to asynchronous communication and time delays is shown in theory in [13] (see, also, [41] for other references). The objective of the next section is to experimentally validate these theoretical results.

IV. EXPERIMENTAL VALIDATION OF CONSENSUS ALGORITHMS FOR COOPERATIVE CONTROL

In this section, the consensus algorithms are experimentally validated via two cooperative control applications including rendezvous and axial alignment. The experimental platform, implementation of the two applications, and experimental results will be described in detail.

A. Experimental Platform

The mobile actuator and sensor network (MASnet) platform in the Center for Self-Organizing and Intelligent Systems (CSOIS), Utah State University, Logan, combines wireless sensor networks with mobility (see [42] and references therein). That is, a large number of robots can serve both as actuators and sensors. Although each robot has limited sensing, computation, and communication ability, they can coordinate with each other as a team to achieve challenging cooperative control tasks such as formation keeping and environment monitoring.

The MASnet platform comprises MASmotes, an overhead camera, and a base station personal computer (PC) as shown in Fig. 1. MASmotes are actually two-wheeled differentially steered robots that can carry sensors and actuators wireless networked via Micaz from Crossbow. The functionality of MASmotes includes intermote and mote to base station communication, data collecting, pulsewidth modulation (PWM) signal generation, and encoder counting. An overhead charge-coupled device (CCD) camera is used to identify each robot and determine its position and orientation [i.e., pseudo-global positioning system (GPS) information]. Images from the camera are processed by the base station. The functionality of the base station includes image processing, serial to programming board communication, pseudo-GPS information broadcasting, and decision making. The base station communicates with a gateway mote mounted on a programming board through a serial port. The gateway mote then communicates with all of the MASmotes

over a 2.4-GHz wireless mesh network. Note that the gateway mote serves as a gateway between wireless communication and serial port communication, and its only purpose is to forward all messages between the serial port and the radio-frequency (RF) port. Through communication the base station can send commands and pseudo-GPS information to each MASmote. All of the MASmotes can also communicate with each other over the 2.4-GHz wireless mesh network.

B. Implementation of Two Target Applications on MASnet Platform

Because both intermote and mote to base station communication are available, the MASnet platform can be used to experimentally test both centralized and decentralized cooperative control schemes. For a centralized scheme, each MASmote is only responsible for its low-level motor control while the base station, served as a centralized station, broadcasts pseudo-GPS information to each MASmote robot, implements cooperative control algorithms, and sends control commands based on the information gathered from the whole team. For a decentralized cooperative control scheme, each MASmote implements its own cooperative control algorithm based on the pseudo-GPS information provided by the base station.

In the experiments, all of the control algorithms are implemented on the MASmotes, and each MASmote only uses the pseudo-GPS information of its own and its local neighbors even if the pseudo-GPS information of all of the MASmotes is provided by the base station. By doing so, distributed cooperative control algorithms involving only local neighbor-to-neighbor interaction via communication or sensing can be tested for multivehicle systems. The feature of local interaction is important in applications where communication or sensing topologies are usually not fully connected, vehicles only have limited communication range and bandwidth, power consumption of the team may be constrained, and the stealth of the team may need to be increased. The two target applications are studied in the experiments including rendezvous and axial alignment. In both applications, only local neighbor-to-neighbor interaction is allowed.

One challenge of applying the consensus algorithms to the mobile robot platform comes from the fact that the robots are nonholonomic while the consensus algorithms assume single-integrator dynamics. Controlling the positions of the robots overcomes this issue with the compromise that the robot orientation information is lost when the robots reach their desired positions.

A *consensus controller* is applied to update the desired position of each MASmote robot at each time instant when the robot receives the position and orientation information of its own and its local neighbors from the pseudo-GPS. The update period for the consensus controller depends on the pseudo-GPS information update period, which is between 0.1 and 0.2 s on average. A low-level PID *position controller* is also applied to guarantee that each robot moves to its (time-varying) desired position (see [42] and references therein for the position controller). The position controller for each robot requires its current position and orientation as well as its desired position provided by its consensus controller. Each robot uses its encoders for position and orientation measurement in between the pseudo-GPS updates. However, a pseudo-GPS update will overwrite the encoder-based position and orientation measurement due to the inaccuracy of the encoders in the low-cost platform.

Let $r_i = [x_i, y_i]^T$ and $r_i^d = [x_i^d, y_i^d]^T$ denote, respectively, the actual and desired position of robot i . For rendezvous, motivated by (3), one strategy is to update r_i^d as

$$\dot{r}_i^d = - \sum_{j \in \mathcal{J}_i(t)} (r_i^d - r_j^d). \quad (12)$$

Another strategy is to update r_i^d as

$$\dot{r}_i^d = - \sum_{j \in \mathcal{J}_i(t)} (r_i - r_j). \quad (13)$$

Note that the desired team rendezvous with (12) is unaffected by robot tracking performance, while the desired team rendezvous with (13) is dependent on the actual positions of the robots. The two strategies may be appropriate for different contexts depending on the application scenarios. In the experiments, a discrete-time version of (13) is used. In essence, the algorithm (13) updates r_i^d to be the average of its current position and the current positions of its local neighbors at each sample time.

Note that (13) can be written as

$$\dot{r}_i^d = - \sum_{j \in \mathcal{J}_i(t)} (r_i^d - r_j^d) + \sum_{j \in \mathcal{J}_i(t)} (e_i - e_j)$$

where $e_i = r_i^d - r_i$ denotes the i th robot's tracking error. With the low-level position controller, $e_i(t)$ is bounded for all t and $e_i(t)$ approaches zero as $t \rightarrow \infty$, $i = 1, \dots, n$, which implies that $\sum_{j \in \mathcal{J}_i(t)} (e_i - e_j)$ is bounded for all t and approaches zero as $t \rightarrow \infty$. Therefore, it follows that with (13) $\|r_i^d(t) - r_j^d(t)\|$ is bounded for all t and approaches zero as $t \rightarrow \infty$ under the conditions of Lemmas 3.1 and 3.2. In other words, the cascade system composed of the consensus controller and the position controller is stable, i.e., $r_i^d(t) \rightarrow r_j^d(t)$ and $r_i(t) \rightarrow r_j(t)$, $\forall i \neq j$, asymptotically as $t \rightarrow \infty$.

For axial alignment, the following algorithm is applied to update r_i^d as:

$$\dot{r}_i^d = - \sum_{j \in \mathcal{J}_i(t)} [(r_i - r_j) - (\delta_i - \delta_j)] \quad (14)$$

where $\delta_i = [\delta_{ix}, \delta_{iy}]^T$ has been chosen to guarantee that the robots align on a horizontal line with a separation distance of 24

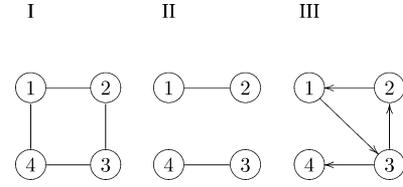


Fig. 2. Interaction topologies among four robots for the rendezvous application.

cm along the x -axis between two adjacent neighboring robots. The stability of (14) can be analyzed in a similar way to (13) by following Corollary 3.1.

C. Experimental Results

In this section, experimental results are shown for rendezvous and axial alignment on the MASnet platform.

1) *Rendezvous*: For the rendezvous application, rendezvous of four MASmote robots is studied under directed, time-invariant, and dynamic interaction topologies, respectively. The motivation for studying dynamic interaction topologies comes from the following observation. In real-world applications, the interaction topology among vehicles will likely be dynamic. For instance, communication links between vehicles may be unreliable due to disturbances, or they may be subject to communication range limitations. Alternatively, if information is exchanged via direct sensing, the visible neighbors of a vehicle will likely change over time.

Fig. 2 shows three different time-invariant interaction topologies associated with cases I–III. In particular, case I corresponds to an undirected connected graph, case II corresponds to an undirected graph with separated subgroups, and case III corresponds to a general directed graph containing a directed spanning tree.

Fig. 3 shows the experimental results of the rendezvous captured by the overhead camera for cases I–III, where the circles denote the initial positions of the robots, and the colored dots denote the actual trajectories of the robots identified by the overhead camera. If the vision system fails to identify a robot during a certain time period, no pseudo-GPS update is available to the robot (equivalent to a pseudo-GPS packet loss), and therefore, no colored dot is placed on the screen during that time period. The trajectories of the robots are captured between $t = 0$ and $t = t_f$ s.

Fig. 3 shows that the four robots rendezvous in all cases except in case II where only a subgroup rendezvous, which is consistent with the first argument of Lemma 3.1 since only the directed interaction graph of case II does not contain a directed spanning tree. In the experiments, the computationally complex task of processing the image, finding MASmote markers, and extracting position and orientation information introduces a delay of 0.1–0.2 s between image capture and position and orientation information broadcast. In addition, the hardware and software limitations of the vision system cause frequent pseudo-GPS packet loss (roughly 2%–5%) and an average position measurement error of 1.32 cm. When a robot is lost from the view of the overhead camera due to glare or other issues, the robot's position and orientation information is no

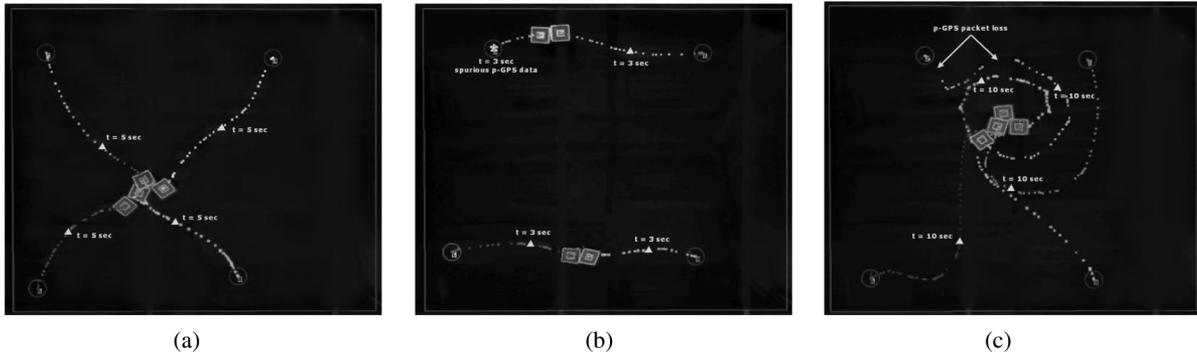


Fig. 3. Experimental results of rendezvous for cases I-III. (a) Case I ($t_f = 11.58$ s). (b) Case II ($t_f = 6.93$ s). (c) Case III ($t_f = 18.89$ s).

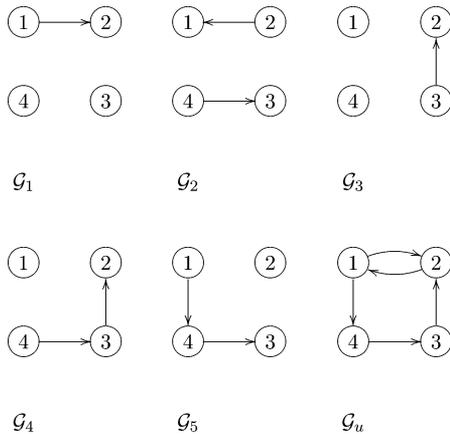


Fig. 4. Switching information topologies $\mathcal{G}_1 - \mathcal{G}_5$ and their union \mathcal{G}_u for rendezvous.

longer available to itself and its local neighbors, implying a pseudo-GPS packet loss. Specifically, in case III, the upper left robot does not receive pseudo-GPS update over several sample periods as shown in Fig. 3(c). Despite the existence of time delay and pseudo-GPS packet loss, the rendezvous experiments achieve results as predicted by the theory described in Section III. With the consensus controller (13), the desired position for a robot simply remains constant during the time period when the position of its own or its local neighbors is not available. That is, the consensus algorithm (13) remains stable with time delay or pseudo-GPS loss although the algorithm will converge slower, which in turn results in slower rendezvous for the team. With the low-level position controller, when there is no pseudo-GPS update, a robot relies on encoder data to move toward the desired position calculated at the previous sample period, which provides a way to compensate for time delay and pseudo-GPS loss.

In theory, if each robot has the same tracking performance, the final rendezvous with (13) should be the average of the initial positions of those robots that have a directed path to all of the other robots. In the experiments, due to the discrepancy among the robots and inaccuracy caused by the vision system, the final rendezvous is a weighted average of the initial positions of those robots that have a directed path to all of the other robots as shown in Fig. 3. The final rendezvous in case I is a weighted

average of all four robots' initial positions. In contrast, the final rendezvous in case III is a weighted average of the initial positions of robots 1, 2, and 3. This experimental result can be explained by noting that, in case I, each robot has a directed path to every other robot, and in case III, every robot except robot 4 has a directed path to all of the other robots. In particular, in case II, the final rendezvous for the upper two robots is not the center of the two robots' initial positions but closer to the starting position of the upper left robot. It is observed in the experiment that the upper left robot receives spurious orientation information during the beginning period of the experiment, which causes the robot to spin at its initial position without moving toward its desired position during the first several seconds. However, the upper right robot is able to move toward the upper left robot to achieve rendezvous. The example demonstrates the robustness of (13) even in the presence of incorrect output of the vision system for a certain period of time. The robustness of (13) is due to the fact that the desired position for each robot is dynamically determined according to its current position and the current positions of its local neighbors.

To test rendezvous in the case of switching topologies, assume that the interaction topologies for the four robots switch randomly from the set $\bar{\mathcal{G}}_s = \{\mathcal{G}_1, \mathcal{G}_2, \mathcal{G}_3, \mathcal{G}_4, \mathcal{G}_5\}$ as shown in Fig. 4. Note that each directed graph in $\bar{\mathcal{G}}_s$ does not have a directed spanning tree but that the union of these directed graphs denoted by \mathcal{G}_u does have a directed spanning tree as shown in Fig. 4. As the switching among graphs in $\bar{\mathcal{G}}_s$ is random, the condition for consensus in Lemma 3.2 is generically satisfied.

Fig. 5(a) shows the experimental result of rendezvous when the interaction topologies switch randomly from $\bar{\mathcal{G}}_s$ with switching periods randomly chosen between 2.75 and 8 s, while Fig. 5(b) shows the experimental result under the time-invariant interaction topology \mathcal{G}_u . Note that the four robots rendezvous even when the directed topologies switch randomly with time, which validates the theory in Lemma 3.2 and demonstrates the robustness of the consensus algorithm to switching topologies as long as the minimum connectivity condition in Lemma 3.2 is satisfied. By comparing Fig. 5(a) and (b), it can be seen that convergence in the case of switching topologies is slower than in the time-invariant case due to the fact that the robots simply stop when they do not receive information from their local neighbors. Also, the switching topologies result in sudden drastic changes in robot directions as shown in Fig. 5(a). In

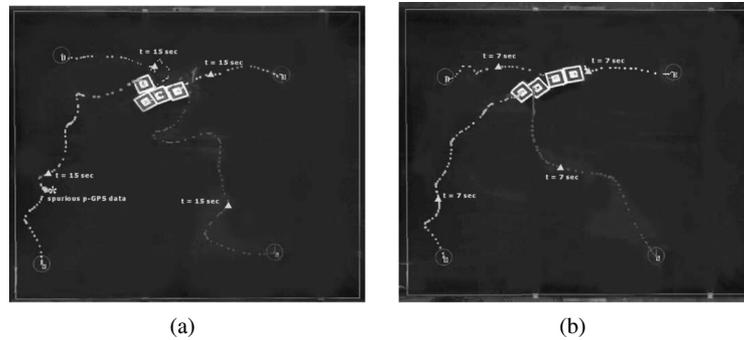


Fig. 5. Experimental results of rendezvous with topologies randomly switching from $\bar{\mathcal{G}}_s$ versus a time-invariant topology \mathcal{G}_u . (a) Rendezvous with topologies randomly switching from $\bar{\mathcal{G}}_s = \{\mathcal{G}_1, \dots, \mathcal{G}_5\}$ ($t_f = 45$ s). (b) Rendezvous with a time-invariant topology \mathcal{G}_u ($t_f = 24$ s).

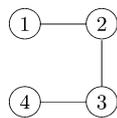


Fig. 6. Interaction topology for axial alignment.

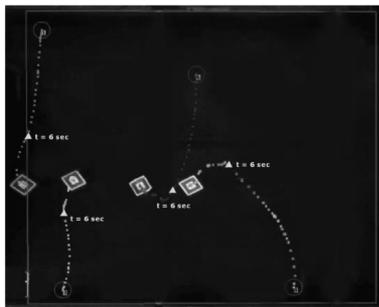


Fig. 7. Experimental result of axial alignment.

addition, in Fig. 5(a), when the lower left robot makes the second turn, due to spurious pseudo-GPS data for several sample periods, the robot starts to wander around as shown by the concentrated dots. However, the inherent stability of the cascade system composed of the consensus controller and the position controller causes the robot to recover once the vision system sends out correct position and orientation data.

2) *Axial Alignment*: For the axial alignment application, the case where four robots are evenly distributed along a straight line is studied under a time-invariant interaction topology.

Fig. 6 shows the undirected interaction topology among the four robots. Fig. 7 shows the experimental result of the axial alignment. In theory, although each robot starts at arbitrary initial positions, their final positions should be evenly distributed along a horizontal line with a separation distance of 24 cm. Due to hardware and software limitations of the vision system, some robots move in the wrong direction for certain periods of time until the vision system correctly identifies their orientations. In addition, there exists a position tracking error with the low-level position controller due to inaccurate position and orientation measurement. Even with these limitations, the overall experiment still achieves the goal of axial alignment with an average

error around 3 cm and further demonstrates the robustness of the consensus algorithm.

V. CONCLUSION AND FUTURE WORK

Consensus algorithms have been applied to two cooperative control problems including rendezvous and axial alignment. The experimental results of both applications on the MASnet platform have demonstrated the effectiveness and robustness of the consensus algorithms to distributed cooperative control. Consensus algorithms provide a promising method for distributed multivehicle cooperative control even in the presence of robot physical limitations, packet loss, information delay, etc.

Despite the success of the experiments, there are limitations. One limitation is that the robots are very dependent on the vision system for position and orientation measurement over long distance due to inaccuracy of the encoders in the low-cost platform. When the vision system produces incorrect measurement or fails to identify a robot during a certain period of time, a robot may move toward a wrong direction. As the time or frequency of vision failure increases, the overall team performance degrades dramatically. Another limitation is the pseudo-GPS update delay. When the robots are moving slowly, the pseudo-GPS update delay has little effect but at full speed the difference between the actual and the broadcast position and orientation can be quite large. This is most noticeable when a robot is rotating. As a result, there is a need to improve the vision system and encoder accuracy and develop a prediction model for position and orientation estimate in the future. The combination of pseudo-GPS data, encoder data, and mathematical model estimates is expected to improve the accuracy of position and orientation measurement. Future work will also introduce a collision avoidance mechanism for formation maneuvering of the robots.¹

REFERENCES

- [1] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1520–1533, Sep. 2004.

¹The videos of the experiments can be found at <http://www.neng.usu.edu/ece/faculty/wren/masnet/Masnet-consensus.htm>

- [2] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. Autom. Control*, vol. 48, no. 6, pp. 988–1001, Jun. 2003.
- [3] Z. Lin, M. Broucke, and B. Francis, "Local control strategies for groups of mobile autonomous agents," *IEEE Trans. Autom. Control*, vol. 49, no. 4, pp. 622–629, Apr. 2004.
- [4] W. Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Trans. Autom. Control*, vol. 50, no. 5, pp. 655–661, May 2005.
- [5] W. Ren, "Multi-vehicle consensus with a time-varying reference state," *Syst. Control Lett.*, vol. 56, no. 7–8, pp. 474–483, Jul. 2007.
- [6] Y. Hong, L. Gao, D. Cheng, and J. Hu, "Lyapunov-based approach to multiagent systems with switching jointly connected interconnection," *IEEE Trans. Autom. Control*, vol. 52, no. 5, pp. 943–948, May 2007.
- [7] W. Ren and E. M. Atkins, "Distributed multi-vehicle coordinated control via local information exchange," *Int. J. Robust Nonlinear Control*, vol. 17, no. 10–11, pp. 1002–1033, Jul. 2007.
- [8] L. Moreau, "Stability of multi-agent systems with time-dependent communication links," *IEEE Trans. Autom. Control*, vol. 50, no. 2, pp. 169–182, Feb. 2005.
- [9] J.-J. E. Slotine and W. Wang, "A study of synchronization and group cooperation using partial contraction theory," in *Cooperative Control*, ser. Lecture Notes in Control and Information Sciences, V. Kumar, N. E. Leonard, and A. S. Morse, Eds. Berlin, Germany: Springer-Verlag, 2005, vol. 309, pp. 207–228.
- [10] D. Bauso, L. Giarre, and R. Pesenti, "Non-linear protocols for optimal distributed consensus in networks of dynamic agents," *Syst. Control Lett.*, vol. 55, no. 11, pp. 918–928, Nov. 2006.
- [11] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Syst. Control Lett.*, vol. 53, no. 1, pp. 65–78, 2004.
- [12] Y. Hatano and M. Mesbahi, "Agreement over random networks," *IEEE Trans. Autom. Control*, vol. 50, no. 11, pp. 1867–1872, Nov. 2005.
- [13] L. Fang and P. J. Antsaklis, "Information consensus of asynchronous discrete-time multi-agent systems," in *Proc. Amer. Control Conf.*, Portland, OR, Jun. 2005, pp. 1883–1888.
- [14] F. Bourgault, T. Furukawa, and H. F. Durrani-Whyte, "Coordinated decentralized search for a lost target in a Bayesian world," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Las Vegas, NV, Oct. 2003, pp. 48–53.
- [15] H. F. Durrant-Whyte and M. Stevens, "Data fusion in decentralized sensing networks," presented at the 4th Int. Conf. Inf. Fusion, Montreal, QC, Canada, 2001.
- [16] T. H. McLoughlin and M. Campbell, "Scalable GNC architecture and sensor scheduling for large spacecraft networks," *J. Guid. Control Dyn.*, vol. 30, no. 2, pp. 289–300, Mar.–Apr. 2007.
- [17] 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.
- [18] J. Cortes, S. Martinez, and F. Bullo, "Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions," *IEEE Trans. Autom. Control*, vol. 51, no. 8, pp. 1289–1298, Aug. 2006.
- [19] S. L. Smith, M. E. Broucke, and B. A. Francis, "Curve shortening and the rendezvous problem for mobile autonomous robots," *IEEE Trans. Autom. Control*, vol. 52, no. 6, pp. 1154–1159, Jun. 2007.
- [20] D. V. Dimarogonas and K. J. Kyriakopoulos, "On the rendezvous problem for multiple nonholonomic agents," *IEEE Trans. Autom. Control*, vol. 52, no. 5, pp. 916–922, May 2007.
- [21] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1465–1476, Sep. 2004.
- [22] G. Lafferriere, A. Williams, J. Caughman, and J. J. P. Veerman, "Decentralized control of vehicle formations," *Syst. Control Lett.*, vol. 54, no. 9, pp. 899–910, 2005.
- [23] W. Ren, "Consensus strategies for cooperative control of vehicle formations," *IET Control Theory Appl.*, vol. 1, no. 2, pp. 505–512, Mar. 2007.
- [24] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Trans. Autom. Control*, vol. 51, no. 3, pp. 401–420, Mar. 2006.
- [25] 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.
- [26] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, "Flocking in fixed and switching networks," *IEEE Trans. Autom. Control*, vol. 52, no. 5, pp. 863–868, May 2007.
- [27] W. Ren, "Distributed attitude alignment in spacecraft formation flying," *Int. J. Adapt. Control Signal Process.*, vol. 21, no. 2–3, pp. 95–113, Mar.–Apr. 2007.
- [28] W. Ren, "Formation keeping and attitude alignment for multiple spacecraft through local interactions," *J. Guid. Control Dyn.*, vol. 30, no. 2, pp. 633–638, Mar.–Apr. 2007.
- [29] M. Alighanbari and J. How, "Decentralized task assignment for unmanned air vehicles," in *Proc. IEEE Conf. Decision Control*, Seville, Spain, Dec. 2005, pp. 5668–5673.
- [30] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proc. Int. Conf. Inf. Proc. Sensor Netw.*, Los Angeles, CA, Apr. 2005, pp. 63–70.
- [31] D. P. Spanos and R. M. Murray, "Distributed sensor fusion using dynamic consensus," in *Int. Fed. Autom. Control World Congr.*, Prague, Czech Republic, 2005, paper code We-E04–TP/3.
- [32] R. Olfati-Saber and J. S. Shamma, "Consensus filters for sensor networks and distributed sensor fusion," in *Proc. IEEE Conf. Decision Control*, Seville, Spain, Dec. 2005, pp. 6698–6703.
- [33] A. Regmi, R. Sandoval, R. Byrne, H. Tanner, and C. Abdallah, "Experimental implementation of flocking algorithms in wheeled mobile robots," in *Proc. Amer. Control Conf.*, Portland, OR, 2005, pp. 4917–4922.
- [34] 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.
- [35] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 1985.
- [36] F. R. K. Chung, *Spectral Graph Theory*. Providence, RI: American Mathematical Society, 1997.
- [37] W. Ren, R. W. Beard, and T. W. McLain, "Coordination variables and consensus building in multiple vehicle systems," in *Cooperative Control*, ser. Lecture Notes in Control and Information Sciences, V. Kumar, N. E. Leonard, and A. S. Morse, Eds. Berlin, Germany: Springer-Verlag, 2005, vol. 309, pp. 171–188.
- [38] J. Manyika and H. Durrant-Whyte, *Data Fusion and Sensor Management: An Information-Theoretic Approach*. Englewood Cliffs, NJ: Prentice-Hall, 1994.
- [39] W. Ren, R. W. Beard, and D. B. Kingston, "Multi-agent Kalman consensus with relative uncertainty," in *Proc. Amer. Control Conf.*, Portland, OR, Jun. 2005, pp. 1865–1870.
- [40] W. J. Rugh, *Linear System Theory*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1996.
- [41] W. Ren, R. W. Beard, and E. M. Atkins, "Information consensus in multivehicle cooperative control: Collective group behavior through local interaction," *IEEE Control Syst. Mag.*, vol. 27, no. 2, pp. 71–82, Apr. 2007.
- [42] L. Ma, P. Chen, Z. Song, Z. Wang, and Y. Chen, "Pattern formation experiments in mobile actuator and sensor network (Mas-net)," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Edmonton, AB, Canada, Aug. 2005, pp. 3658–3663.