INVITED PAPER

Decentralized Cooperative Aerial Surveillance Using Fixed-Wing Miniature UAVs

Teams of unmanned air vehicles (UAVs) can cooperate when their common objectives are defined and each vehicle has the information needed to cooperate, even when there are communications difficulties.

By Randal W. Beard, Timothy W. McLain, Derek B. Nelson, Derek Kingston, and David Johanson

ABSTRACT | Numerous applications require aerial surveillance. Civilian applications include monitoring forest fires, oil fields, and pipelines and tracking wildlife. Applications to homeland security include border patrol and monitoring the perimeter of nuclear power plants. Military applications are numerous. The current approach to these applications is to use a single manned vehicle for surveillance. However, manned vehicles are typically large and expensive. In addition, hazardous environments and operator fatigue can potentially threaten the life of the pilot. Therefore, there is a critical need for automating aerial surveillance using unmanned air vehicles (UAVs). This paper gives an overview of a cooperative control strategy for aerial surveillance that has been successfully flight tested on small (48-in wingspan) UAVs. Our approach to cooperative control problems can be summarized in four steps: 1) the definition of a cooperation constraint and cooperation objective; 2) the definition of a coordination variable as the minimal amount of information needed to effect cooperation; 3) the design of a centralized cooperation strategy; and 4) the use of consensus schemes to transform the centralized strategy into a decentralized algorithm. The effectiveness of the solution will be shown using both high-fidelity simulation and actual flight tests.

KEYWORDS | Aerial robotics; autonomous systems; consensus; cooperative control; surveillance; unmanned air vehicles

I. INTRODUCTION

Group cooperative behavior implies that individuals in the group share a common objective and act according to the mutual interest of the group. Effective cooperation often requires that individuals coordinate their actions. Coordination can take many forms ranging from staying out of each others' way to directly assisting another individual. In general, group cooperation is facilitated by coordinating the actions of individuals. However, each individual may not necessarily need to directly coordinate with every other individual in the group to effect group cooperative behavior. For example, fish engaged in schooling behavior only react to other fish that are in close physically proximity. We will term this type of coordination local coordination. At the other extreme is global coordination, where an individual coordinates its action with every other individual in the group. Due to communication constraints and computational feasibility, we are primarily interested in group cooperation problems where the coordination occurs locally. One of the interesting challenges in robotics is to design coordination strategies so that local coordination will result in group cooperation.

While there have been numerous publications detailing specialized approaches to cooperation problems, general design methodologies are only beginning to emerge. For the most part, these methodologies assume a group of

Manuscript received June 1, 2005; revised June 1, 2006. This work was supported in part by the Air Force Office of Scientific Research under Awards FA9550-04-1-0209 and FA9550-04-C-0032, in part by the National Aeronautics and Space Agency under Contract NNA-04-AC-51C, and in part by the Utah State Center of Excellence under Contract 05-0621.

R. W. Beard, **D. Kingston**, and **D. Johanson** are with the Department of Electrical and Computer Engineering, Brigham Young University, Provo, UT 84602 USA (e-mail: beard@ee.byu.edu; kingston@byu.edu; davejohansen@gmail.com).

T. W. McLain is with the Department of Mechanical Engineering, Brigham Young University, Provo, UT 84602 USA (e-mail: mclain@byu.edu).

D. B. Nelson is with Northrop Grumman Corporation, San Diego, CA 92127 USA (e-mail: derek.nelson@nqc.com).

Digital Object Identifier: 10.1109/JPROC.2006.876930

homogeneous robots with local coordination. The objective of this paper is describe a design philosophy for cooperation problems that we have successfully applied to a variety of cooperation problems involving unmanned air vehicles (UAVs). Our approach allows a range of coordination strategies ranging from local to global coordination. We do not claim that our approach will be appropriate for all cooperation problems. In fact, we expect that it will be many more years before the general principles underlying cooperative systems will be fully understood. However, we hope that our approach contributes toward that goal.

The essence of our approach is explained in the following steps.

Step 1) **Cooperation Objective and Constraints**: The first step is to analytically define the cooperation objective. Cooperation can often be identified when certain relationships between state variables are satisfied. These relationships are called the cooperation constraints.

Step 2) **Coordination Variable and Coordination Function**: The next step is to identify the essential information that each vehicle must know to coordinate with the team. This information is called the coordination variable. It will often be the case that cooperation can be achieved through a variety of individual actions. To facilitate the selection of the individual actions that best contribute to the cooperation objective, we quantify the relationship between the coordination variable and the cooperation objective and call this function the coordination function.

Step 3) **Centralized Cooperation Scheme**: The next step is to derive a cooperation strategy for minimizing the team objective function assuming that each member of the team has global knowledge of the coordination variable and the coordination functions of each member of the team.

Step 4) **Consensus Building**: In a decentralized situation where communication links are noisy and not persistent, and where the communication topology is dynamically changing and is unknown to each team member, the centralized solution will fail. The final step of our approach is to implement a consensus-building algorithm that ensures that each member of the team has consistent coordination information despite the inadequacies of the communication network.

While we expect this approach to be applicable to ground vehicle scenarios, we have only investigated applications involving fixed-wing aerial vehicles. Therefore, in this paper we will focus the discussion exclusively on fixed-wing UAVs. There has been extensive work in cooperative control for teams of ground robots; however, since this special issue contains other papers that describe approaches targeted at ground robots, we will focus on describing the approaches that have been developed for fixed-wing UAVs.

Cooperation between UAVs has it own set of unique challenges. For example, unlike ground robots, there is generally very little physical coupling except in the obvious cases of close formation flight. The most significant challenge that is unique to UAVs is three-dimensional (3-D) flight with immediate implications on path planning algorithms. Another characteristic unique to fixed-wing UAVs is that forward motion is required. Therefore, stop-and-wait path deconfliction algorithms (e.g., [1]) are not applicable. In addition, small fixed-wing UAVs are highly susceptible to wind. Therefore, cooperation strategies must incorporate feedback at the highest levels to account for objective failure modes.

However, cooperation problems for ground robots and UAVs share a number of similarities. For example, both ground and aerial robots have strict communication constraints: team members must be in close physical proximity to communicate, bandwidth is limited, and the communication topology may change unpredictably with time. Both ground and aerial robots must deal with collision avoidance constraints. Ground robots are necessarily concerned with maneuvering around each other in confined spatial environments. On the other hand, aerial robots usually have more room to maneuver but collisions are typically catastrophic. Another similarity is that decentralized cooperation strategies are generally required for both ground and aerial robots. In addition, cooperation strategies must be robust to the failure of individual team members.

Studies specific to cooperative control of UAVs have recently appeared in the literature. Extensive efforts have been directed toward close formation flight. Early studies reported in [2] and [3] reported significant potential fuel savings that could be gained by close formation flight. In [4], the physical equations that describe a fixed-wing aircraft flying in the vortex of the leader are described and a control system based on the linearized model is developed. The approach is extended to nonlinear aerodynamic coupling terms in [5]. Standard inner/outer loop designs are extended to close-formation flight in [6]. A behavioral approach to aircraft formation flight is given in [7]. In [8], differential flatness is used to generate group formation maneuvers. The effects of communication constraints on close formation flight are studied in [9]. Rigorous conditions for stable formation flight with limited communication are developed in [10] and [11].

Multiple UAV cooperative timing problems have also received significant attention. One version of this problem is where multiple UAVs are required to converge on the boundary of a radar detection area to maximize the element of surprise [12]–[16]. Cooperative timing problems also arise in refueling scenarios, fire and hazardous material monitoring [17], moving area of regard problems, and continuous surveillance problems [18].

Cooperative timing problems are sensitive to the assignment and ordering of tasks. One approach for handling cooperative timing is to apply timing constraints to the task assignment problem. In [19]–[21], mixed-integer linear programming (MILP) is used to solve tightly coupled task assignment problems with timing constraints. The advantage to this approach is that it yields the optimal solution for a given problem. The primary disadvantages are the complexity of problem formulation and the computational burden involved. Pruning strategies for simplifying the MILP problem have been proposed to enable near real-time solutions.

Although path planning for single UAVs has been an active area of research for some time (e.g., see [22]–[26]), cooperative path planning approaches for UAVs have only recently begun to appear. In [27], a decentralized optimization method based on a bargaining algorithm is developed and applied to a multiple aircraft coordination problem. A hybrid hierarchical control architecture is used for air traffic control in [28] and [29].

In this paper, we describe our approach to cooperative control which is based on two main ideas. The first is the notion of coordination variables and coordination functions, which were introduced in [13] and [30]. In essence, the coordination variable is the minimum amount of information that needs to be exchanged between two agents to effect coordination. Although it is known by different names, the notion of a coordination variable is found in many other works on cooperative control. For example, [31] and [32] introduce an "action reference" which, if known by each vehicle, facilitates formation keeping. In leader-following applications [33], [34], the states of the leader constitute the coordination variable, since the actions of the other vehicles in the formation are completely specified once the leader states are known. In [35]–[37], the notion of a virtual structure is used to derive formation control strategies. The motion of each vehicle is causally dependent on the dynamic states of the virtual structure; therefore, the states of the virtual structure are the coordination variable. In [38], a team of autonomous underwater vehicles (AUVs) is controlled to "swarm" around a desired mean location of the team with a specified standard deviation. The action of each vehicle is dependent on the location of its nearest neighbor and the desired mean and standard deviation. This information is the coordination variable. Coordination variables may also be discrete in nature. For example, in [14] and [20], cooperative task allocation is addressed. Individual vehicle behavior is dependent on the task allocation vector which becomes the coordination variable. Similarly, in [39], the coordination variable is the dynamic role assignment in a robot soccer scenario.

The second main idea in our approach to cooperative control is the notion of consensus seeking. Since coordination may be required between two agents that do not directly communicate, distributed consensus algorithms are required to ensure that the agents share similar coordination variables [40]. There is a growing body of literature on distributed consensus seeking. For example, consensus problems have recently been addressed in [41]–[48], to name a few. In [42], sufficient conditions are given for consensus of the heading angles of a group of agents under undirected switching interaction topologies. In [43], average consensus problems are solved for a network of integrators using directed graphs. In [46] and [47], an algebraic graph approach is used to show necessary and/or sufficient conditions for consensus of information under time-invariant and switching-interaction topologies, respectively. In [44], a set-valued Lyapunov function approach is used to consider consensus problems with unidirectional time-dependent communication links. A Kalman filter approach to consensus seeking that which accommodates agent confidence is described in [49].

The action of each vehicle is dependent on the location of its nearest neighbor and the desired mean and standard deviation. This information is the coordination variable.

The remainder of the paper is organized as follows. In Section II, we describe different types of coupling and coordination that occur in cooperative control problems. In Section III, we give an overview of our approach to cooperative control problems and illustrate the approach with a simple academic example. A high-level description of how that approach is applied to several cooperative UAV scenarios is given in Section IV. Section V describes our experimental testbed and discusses some of the practical challenges involved in flight testing cooperative UAV experiments. Section VI gives a detailed description of the application of our approach to cooperative aerial surveillance using miniature UAVs and presents simulation and hardware results. Finally, Section VII contains conclusions and some final thoughts.

II. COUPLING IN COOPERATIVE CONTROL PROBLEMS

One of the primary challenges in developing generalized strategies for cooperative control is the identification of broad classes of problems that are amenable to well-defined straightforward approaches. One way to classify cooperative control problems is by the level and type of coupling involved. For example, flying a UAV in the wake of another UAV to minimize the induced drag requires tight physical coupling between the UAVs. On the other hand, a team of UAVs tasked to cooperatively search a particular region is coupled primarily through the cooperation algorithms employed by the individual UAVs. We believe that our approach is particularly suited to algorithmically coupled problems. In this section, we describe different types of coupling that occur in cooperative control problems.

With respect to the cooperative control of multiagent systems, the degree and form of the coupling between the agents composing the system is of paramount importance to the nature and level of the cooperation that can be achieved. Cooperation implies some degree of coupling, if only through the cooperation objective or constraints involved. Generally speaking, the greater the degree of the coupling, the more challenging it is to formulate effective cooperative solutions.

Cooperative control problems can typically be formulated with a cooperation objective, or cooperation constraints, or both. A cooperation objective is typically optimized to increase the level of cooperation. Cooperation constraints, when satisfied, can be used to define the occurrence of cooperation.

A. Objective Coupling

Objective coupling describes the least restrictive form of coupling. It occurs when an agent's decisions affect only its costs and outcomes and do not influence another agent's costs and outcomes directly. Each agent's decisions affects the cooperation objective and the feasibility of cooperation constraints. Objective coupling requires agents to coordinate to ensure that constraints are satisfied and that the objective is optimized. An example of objective coupling is the UAV cooperative timing problem described in [30]. Suppose that two vehicles are to navigate independently through a threat field with the objective of simultaneously flying over a prescribed destination. The cooperation constraint requires the vehicles to arrive at the same instance of time, while the cooperation objective might be to minimize the collective power required for the mission. The trajectory taken by one vehicle (its decisions) does not affect the trajectory taken by the other vehicle directly, but it does affect the other vehicle through the cooperation constraint and objective. As this example illustrates, some degree of coupling is necessary for cooperation to occur; however, the coupling is strictly through the mission objective.

B. Local Coupling

Local coupling describes a more restrictive type of coupling in cooperative systems. As with objective coupling, each agent's decisions influence the cooperation objective and the feasibility of the cooperation constraints. Under local coupling, however, an agent's decisions affect not only its own costs and outcomes but also the decisions (and hence the costs and outcomes) of its local neighbors. A simple example of local or nearest neighbor coupling is shown in the cooperative search problem described in [50], where N UAVs are assigned the task of cooperatively searching an area of interest. To provide some structure to the search task, the vehicles are required to maintain a loose row formation. To avoid collisions, the UAVs are not allowed to overlap laterally. To maintain communication, the lateral spacing of the vehicles must be kept less than the communication range. The lateral spacing constraints can be viewed as cooperation constraints. The cooperation objective is to visit as many targets as possible. The decision by one vehicle to alter its trajectory to visit a sequence of targets will directly affect the decisions of its neighbors and the costs and benefits associated with their decisions. The key difference between objective coupling and local coupling is that for objective coupling the cost accrued for any agent is a function of that agent's decisions only. For local coupling, the cost accrued for any agent is a function of its own decisions as well as the decisions of its local neighbors.

C. Full Coupling

Fully coupled systems involve agents whose decisions affect the costs and outcomes of all other members of the team and thus their decisions, i.e., what a single agent chooses to do is influenced by what all other agents on the team are doing. Coupling exists through the cooperation objective and cooperation constraint as before, but in this most restrictive form of coupling the decisions of the individual agents are coupled directly. An example of full coupling is the wide area search munition problem described in [51]. In this problem, a team of autonomous flying munitions are tasked to search a region and identify potential targets. Once a target is identified, it must be classified by multiple passes over the target using one or more munitions. Upon identification and verification, the target is attacked removing one of the munitions from the team. The target must then be revisited for the purpose of battle damage assessment. If the minimum turning radius of the munition is large in relation to the search area, then each of these tasks will likely be performed by a different vehicle. The coupling in this scenario is complex and requires that each member of the team knows the intentions and flight paths of all the other members of the team.

D. Dynamic Coupling

When the vehicles are coupled through physical interactions, we call it dynamic coupling. The coupling in these systems can be either local or full coupling. For example, in close-formation flight of aircraft, the aerodynamic coupling that exists is local in that it affects those aircraft in the immediate wake of a leading aircraft. Those

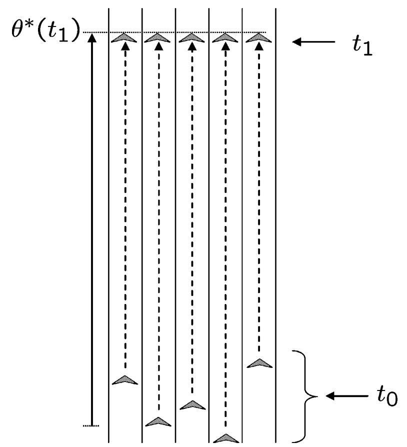

Fig. 1. ***Team of air vehicles are tasked to proceed along uniform front at constant ground speed.***

aircraft on the outer edges of the formation are not affected by those in the center of the formation.

Although the physics of dynamic coupling can be quite complex, there is one significant advantage when the actions of one agent are captured directly in the equations of motion describing other agents. These systems can be treated as one large system by combining their equations of motion. In this way, these large systems are amenable to control theoretic approaches. With one model to characterize the behavior and interaction of multiple agents, conventional single-agent approaches can be applied to achieve cooperation.

III. APPROACH TO DISTRIBUTED COOPERATIVE CONTROL PROBLEMS

In this section, we give an overview of our approach to cooperative control problems and illustrate with a simple example. The approach has been applied to problems with objective coupling [30], [49], loose coupling [50], and dynamic coupling [52]. We will illustrate the main ideas through the use of a simple academic example.

Example Problem: Suppose that five air vehicles are assigned virtual lanes as shown in Fig. 1. Assume that the lateral position of the vehicles is maintained inside the lanes by an onboard autopilot and that the longitudinal position y_i of each vehicle is governed by the dynamics $y_i = u_i$. Assume that the vehicles are initially at different longitudinal positions in the lane. The cooperation goal is to maneuver the vehicles so that they proceed along a uniform front at a constant known ground speed v, as shown in Fig. 1.

A. Cooperation Constraints and Objectives

The first step in our approach is to identify and quantify the cooperation constraint and the cooperation objective. The cooperation constraint is a formal definition of the team goal and indicates exact conditions when cooperation is achieved. More precisely, if x_i is the situation state of the ith vehicle and u_i is the decision variable, then the cooperation constraint is a positive definite mapping $J_{\text{constraint}}(x_1, u_1, x_2, u_2, \ldots x_N, u_N)$ that is identically zero when cooperation is achieved. In our example problem, if N is the number of agents, then a possible cooperation constraint is the mapping

$$J_{\text{constraint}} = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(y_i - y_j)^2 \tag{1}$$

which is identically zero when $y_i = y_j$. In the example problem, note that if the team is closely but not precisely aligned, then the team may still be considered to be in cooperation. When $J_{\text{constraint}} < \epsilon$, we say that the team has achieved ϵ cooperation.

For a given world situation state $\mathcal{X} = \{x_1, \ldots, x_N\}$, there may be many different decision variables $\mathcal{U} = \{u_1, \ldots, u_N\}$ that achieve ϵ cooperation. In addition, there may be auxiliary objectives that we would like to minimize. For instance, in the example problem we may also want to minimize overall fuel expenditure. To capture these auxiliary objectives we introduce a positive definite function $J_{\text{objective}}(\mathcal{X}, \mathcal{U})$ that quantifies these objectives. This function is called the coordination objective [30]. In our example problem, a possible coordination objective is given by the linear quadratic regulator equation [53]

$$J_{\text{objective}} = \sum_{i=1}^{N}\int_{t}^{\infty} q\big(y_i(\tau) - \theta^*(\tau)\big)^2 + r\big(u_i(\tau) - v\big)^2 d\tau$$

where q and r are positive constants, and where θ^* is the position of the uniform front.

It is often the case that the primary goal is to effect cooperation in a team without an auxiliary cooperation objective. In that case, there is not a cooperation objective and the algorithms focus on satisfying the cooperation constraint.

B. Coordination Variables and Coordination Functions

The second step in our approach is to determine the information that must be shared to achieve cooperation and to organize that into a single vector called the coordination variable. We will let θ^* denote the coordination variable. The philosophy is to distill the essentials of the cooperation problem to a set of parameters that, if known by every agent in the group, can be used to select the decision variable so that the cooperation constraint is achieved. In the example problem, if every vehicle knows the desired position of the front, then each can regulate its position to align with the front. Therefore, let θ^* be the desired position of the uniform front. In the example problem, we will assume that θ^* evolves according to

$$\dot{\theta}^* = v.$$

Our method assumes that the cooperation constraint can be written as a function of the coordination variable. Note that for our example problem, the cooperation constraint given in (1) can be bounded by a function of the coordination variable θ^*, as

$$J_{\text{constraint}} \leq \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left[(y_i-\theta^*)^2+(y_j-\theta^*)^2\right] < \epsilon.$$

To facilitate minimization of the auxiliary cooperation objective subject to the cooperation constraint, we desire to write the cooperation objective in terms of θ^*. To this end, we assume that the cooperation objective can be expressed as a convex function of myopic objective functions for each vehicle. The myopic objective functions depend only on the coordination variable as well as the situation state and decision variable of the individual vehicle. This myopic objective function is called the coordination function [30] and is denoted $J_{\text{cf},i}(x_i, u_i, \theta^*)$. In the example problem, we have

$$J_{\text{objective}} = \sum_{i=1}^{N} J_{\text{cf},i}$$

where

$$J_{\text{cf},i} = \int_t^\infty q\big(y_i(\tau)-\theta^*(\tau)\big)^2 + r(u_i(\tau)-v)^2 d\tau. \qquad (2)$$

The coordination function parameterizes the effect of the coordination variable on the objectives of each agent, i.e., the coordination function describes how the myopic objective of each agent changes with changes in the coordination variable.

Posing the cooperation problem in terms of coordination variables and coordination functions will usually reduce the dimensionality of the problem. The example problem is already a scalar problem and so the dimensionality has not been reduced.

C. Centralized Cooperation Scheme

Given the terminology introduced in the previous two sections, we can pose the cooperation scenario as the following optimization problem:

$$\theta^* = \arg\min\left\{\lim_{t\to\infty}\sum_{i=1}^{N} J_{\text{cf},i}(\theta; x_i, u_i)\right\}$$
$$\text{subject to}: \lim_{t\to\infty} J_{\text{constraint}}(\theta; \mathcal{X}, \mathcal{U}) < \epsilon. \qquad (3)$$

A fundamental part of our approach to cooperative control is the design of a centralized strategy that solves this optimization problem. Centralized strategies are usually easier to design than decentralized strategies. (Note that the centralized algorithm will be problem dependent.) In the process of solving problem (3), the centralized algorithm produces a decision variable for the ith vehicle denoted [30]

$$u_i = f_i^\dagger(\theta^*, x_i) \qquad (4)$$

where we assume that $f_i^\dagger$ is continuous in θ^*. Equations (3) and (4) represent what we term the cooperation algorithm.

For the example problem, the centralized solution requires that each vehicle knows the position of the front $\theta^*(t)$. Accordingly, the vehicles implement the control law

$$u_i = v + k(\theta^* - y_i)$$

where $k = \sqrt{q/r}$ is chosen to minimize the coordination function given in (2). Given that $\dot{\theta}^* = v$, it is straightforward to show that using this strategy yields $y_i(t) \to \theta^*(t)$ for each $i = 1, \ldots, N$, which implies that

$$\lim_{t\to\infty} J_{\text{constraint}} \to 0$$

i.e., the cooperation constraint is satisfied for every ϵ. Fig. 2 shows a simulation plot of the centralized solution where $v = 0.1$ and $k = 1$.

D. Consensus Building

The final step in our approach is to use consensus schemes to decentralize the cooperation algorithm. Multiple vehicle cooperation requires communication between vehicles. In real-world environments the communication links will be noisy and nonpersistent, and the communication topology will be dynamically changing and unknown to each team member. Therefore, centralized solutions are rarely feasible. The key insight is that if each agent locally instantiates the cooperation algorithm, and the inputs to each local instantiation are identical, then

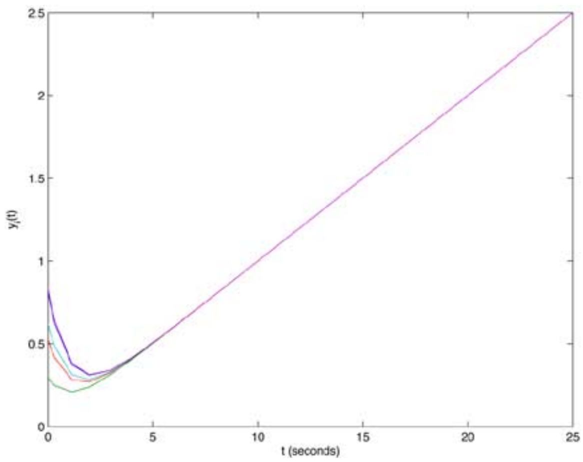

Fig. 2. ***Centralized solution to example problem. Position of uniform front is transmitted to all vehicles from centralized location.***

assuming that the algorithm is deterministic, it will produce identical outputs on each vehicle. However, if the local inputs are different, than each vehicle will compute a different instantiation of the coordination variable which we label as θ_i. Therefore, from (4) we see that the decision variable for the ith vehicle is given by

$$u_i = f_i^{\dagger}(\theta_i, x_i).$$

Since $f_i^{\dagger}$ is continuous, $f_i^{\dagger}(\theta_i) \rightarrow f_i^{\dagger}(\theta^*)$ as $\theta_i \rightarrow \theta^*$. Therefore, the objective of the consensus algorithm is to ensure that $\theta_i \rightarrow \theta_j$ for every i, j, despite noisy communication channels and time-varying communication topologies.

Our approach to consensus building is built on the intuitive notion of compromise. Each agent adjusts its instantiation of θ_i to be a weighted average of those agents with whom it communicates [47], [49]. For continuous-time systems the consensus strategy is given by

$$\dot{\theta}_i = \sum_{j=1}^{n} g_{ij}(t)K_{ij}\big(\big(\theta_j + \nu_{ij}(t)\big) - \theta_i\big) \tag{5}$$

where $g_{ij}(t)$ is one if there is a communication channel from the jth vehicle to the ith vehicle at time t, and zero otherwise, and K_{ij} is a weighting matrix and ν_{ij} is the communication noise. For discrete-time systems, the consensus strategy is given by

$$\begin{aligned}\theta_i[n+1] = {} & \theta_i[n] \\ & + \sum_{j=1}^{n} g_{ij}[n]K_{ij}\big(\big(\theta_j[n] + \nu_{ij}[n+1]\big) - \theta_i[n]\big)\end{aligned} \tag{6}$$

where n is the sample index and $\sum_{j=1}^{n} g_{ij}[n]K_{ij} = 1$ for each n. In [49], we have shown the following result.

Theorem 3.1: Assume that the communication noise is zero. If there exist infinitely many consecutive uniformly bounded time intervals such that the union of the communication graph across each interval has a spanning tree and the weights in K_{ij} are bounded, then (5) guarantees that $\theta_i(t) \rightarrow \theta_j(t)$ and (6) guarantees that $\theta_i[n] \rightarrow \theta_j[n]$ asymptotically.

The conditions for consensus in Theorem 3.1 are surprisingly mild. In essence, the theorem states that if the agents communicate with other agents sufficiently often, all vehicles will come into agreement.

Theorem 3.1 guarantees consensus in the absence of communication noise. If communication noise is present in the system, then we need to show that ϵ consensus is achieved, i.e.,

$$\lim_{t\rightarrow\infty} \sum_{ij} \big\| \theta_i(t) - \theta_j(t) \big\| < \epsilon.$$

To that end, we stack the local instantiation of the coordination variables as $\boldsymbol{\theta} = (\theta_1^T, \ldots, \theta_N^T)^T$ and the noise vector as $\boldsymbol{\nu} = (\nu_{11}^T, \ldots, \nu_{NN}^T)^T$ and write (5) as

$$\dot{\boldsymbol{\theta}} = A(t)\boldsymbol{\theta} + B(t)\boldsymbol{\nu}. \tag{7}$$

In [54], we have shown that (7) is input-to-state stable, which implies that the consensus error is uniformly bounded by a gain times the power in the communication noise.

A fundamental result in nonlinear control theory is that the cascade of two input-to-state stable systems is also input-to-state stable [55]. Consider the control diagram shown in Fig. 3. The consensus algorithm on each vehicle is input-to-state stable from the communication noise to the consensus error. Therefore, if the cooperation algorithm is input-to-state stable from the consensus error to the cooperation constraint, then the cascade system is input-to-state stable from the communication noise to the cooperation constraint, implying that ϵ cooperation is achieved for low-enough levels of communication noise.

The application of the consensus strategy given in (5) to the example problem implies that each vehicle updates its coordination variable according to

$$\dot{\theta}_i = v + \alpha \sum_{j \neq i} g_{ij}(t)(\theta_j - \theta_i).$$

Each vehicle then implements the modified control law

$$u_i = f_i^{\dagger}(\theta_i, y_i) \triangleq v + k(\theta_i - y_i)$$

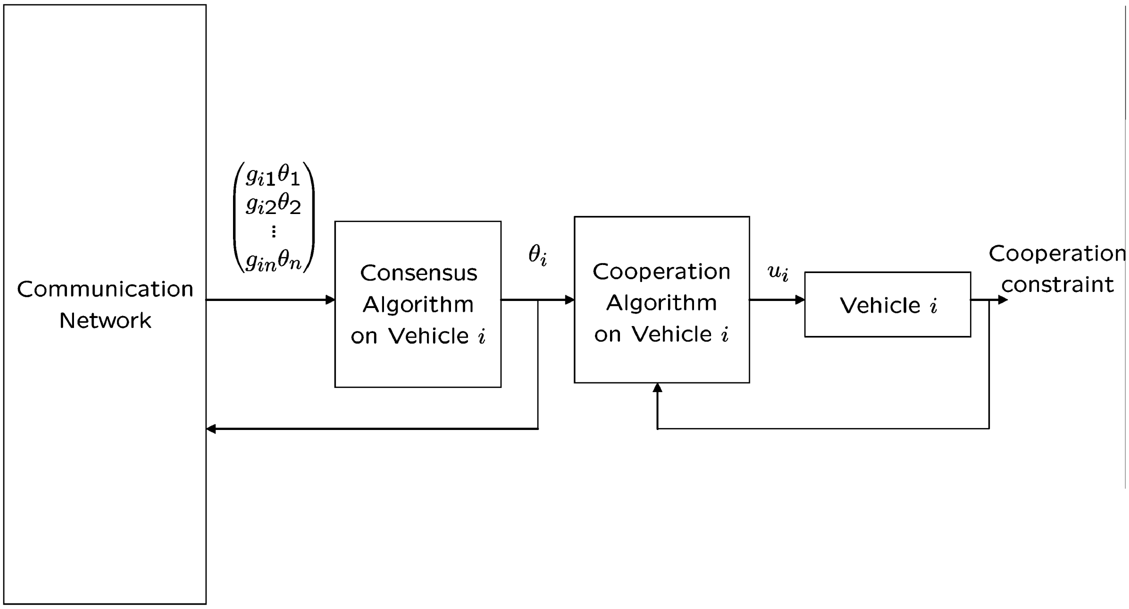

Fig. 3. *Cascade system where output of consensus-building mechanism drives coordination algorithm, which uses feedback from agent to achieve coordination.*

where k can be chosen to minimize the modified coordination function

$$J_{\mathrm{cf},i} = \int_t^\infty q(y_i(\tau) - \theta_i(\tau))^2 + r(u_i(\tau) - v)^2 d\tau. \tag{8}$$

Since the system

$$\frac{d}{dt}(y_i - \theta_i) = -k(y_i - \theta_i) + k\nu_{ij}$$

is input-to-state stable, we are guaranteed to achieve ϵ cooperation. Note that since each vehicle tracks its own notion of the front in an optimal manner [with respect to (8)], this does not imply that the overall team has optimal performance. In fact, centralized solutions that minimize the coordination function will always have better performance than decentralized solutions. Fig. 4 shows a simulation plot of the decentralized solution where $v = 0.1$ and $k = 1$ and the standard deviation of the noise on the communication channels is $\sigma = 0.1$. The first subplot shows the evolution of the coordination variables. The second subplot shows the evolution of the longitudinal position of the vehicles, and the third subplot shows the number of unidirectional communication links that are active in the network as a function of time. Note that updates of θ_i only occur when the ith vehicle is communicating with another vehicle. Despite the low levels of communication, the coordination constraint is still satisfied.

IV. APPLICATIONS OF METHODOLOGY

We have applied the approach of Section III to a variety of problems. To demonstrate the applicability of the approach, we provide a brief sketch of some of these applications.

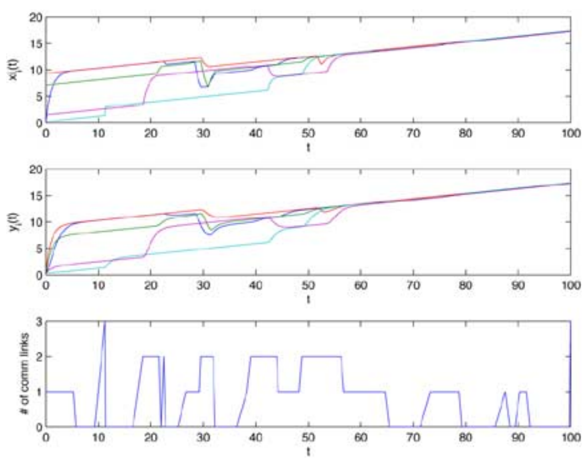

Fig. 4. *Decentralized solution to Example 1. Consensus algorithm is used to negotiate common position of uniform front despite low levels of communication between vehicles.*

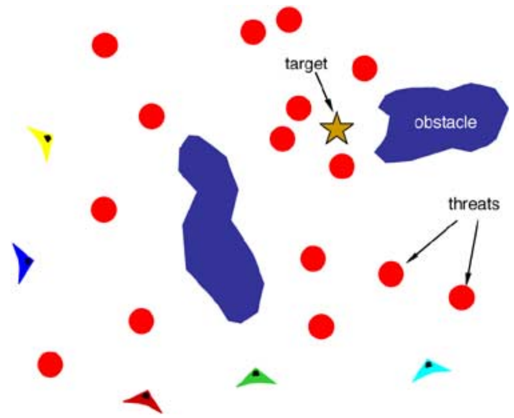

Fig. 5. ***Cooperative timing scenario where multiple vehicles must maneuver through threat and obstacle field to rendezvous at target simultaneously.***

A. Cooperative Timing

Cooperative timing tasks for UAVs are of interest in many military missions. A simple example of this is shown in Fig. 5, where several UAVs are spatially distributed over a wide area. The goal is for the UAVs to arrive at their target simultaneously (or perhaps with a specified time spacing) while avoiding threats and terrain-based obstacles [30]. We assert that cooperation is achieved if the simultaneous arrival constraint is satisfied and the quality of the cooperation is measured by the degree to which threats and obstacles are avoided.

This problem can be placed in the cooperative control framework of Section III by defining the arrival time as the coordination variable θ^*. As expected, the arrival time for each vehicle is a function of the situation state x_i and the decision variable u_i. In this case, the situation state describes the location of threats and terrain obstacles as well as the location of the target. The decision variable is the trajectory that is flown, which can be represented by a set of waypoints and a flight velocity. The coordination function $J_{\mathrm{cf},i}$ for each vehicle describes the threat and obstacle exposure it faces versus the arrival times that it can achieve [30]. The cooperation constraint $J_{\mathrm{constraint}}$ requires that all vehicles arrive at the target simultaneously, which is implicitly achieved through the selection of a team-optimal arrival time as the coordination variable θ^*. The coordination variable is chosen to minimize the cooperation objective $J_{\mathrm{objective}}$, which is the maximum threat cost faced by any team member.

B. Cooperative Search

Another example of cooperative control that is amenable to our formulation is that of cooperative search. In this problem, a team of UAVs is to fly in a loose echelon formation with forward-looking sensors scanning for observation targets and hazards [50] as shown in Fig. 6. The UAVs are required to stay close enough to their neighbors to maintain communication but must also stay far enough apart to avoid colliding with one another. Individually, the UAVs have the objective of observing as many targets as possible while avoiding the hazards as they fly over the region of interest.

In this problem, the cooperation constraint $J_{\mathrm{constraint}}$ is to maintain the loose formation by staying within communication range of neighboring UAVs but not crossing paths with neighboring UAVs. The cooperation objective $J_{\mathrm{objective}}$ is to maximize the number of targets observed by the team as it traverses the region of interest. The situation state x_i is comprised of the observation target locations, the hazard locations, and the formation row ordering of the UAVs. The decision variable u_i is the trajectory that the ith UAV flies over the observation region. The coordination variable for each vehicle is the lateral range that it traverses as it flies its chosen trajectory. The coordination function expresses the number of targets observed as a function of the lateral range (minimum and maximum displacement laterally). Based on coordination function information, lateral ranges can be determined for each UAV to maximize the number of targets viewed and satisfy the communication and collision avoidance constraint. From the lateral range information, each UAV can choose its own trajectory to maximize the number of targets that it views.

C. Cooperative Fire Surveillance

The final example that we will consider is cooperative fire surveillance. As shown in Fig. 7, multiple UAVs are distributed around the perimeter of a growing forest fire [17]. The goal for the team is to monitor and track the perimeter as it evolves and to communicate the

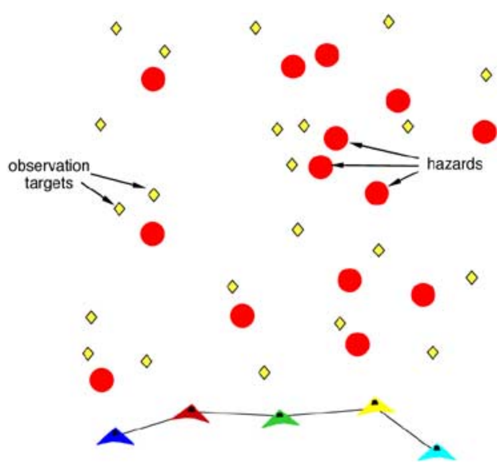

Fig. 6. ***Cooperative search scenario where multiple vehicles must maneuver around threats and pass over observation targets while maintaining communication connectivity and avoiding collisions.***

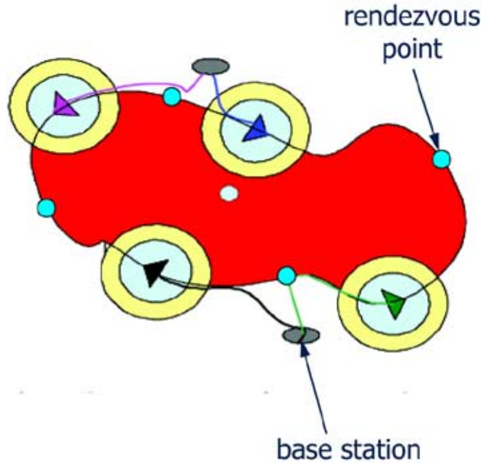

Fig. 7. *Cooperative fire surveillance scenario.*

coordinates of the perimeter to fire fighters on the ground. The UAVs have a limited communication range that requires them to work cooperatively to relay perimeter information to the base station on the ground. In this scenario, each UAV patrols a segment of the perimeter. When it encounters another UAV, it exchanges information and reverses direction until it encounters another UAV or arrives at a predetermined rendezvous point.

In this problem, the situation state x_i describes the fire perimeter and ground station location. The decision variable u_i is the next rendezvous point. The cooperation constraint $J_{\text{constraint}}$ is to equalize the path lengths flown by the UAVs around the periphery of the fire. The cooperation objective $J_{\text{objective}}$ is to minimize the latency between the time that perimeter information is sensed and the time that it is received at the ground station. The coordination variable θ^* is the length of the path last flown, which is a function of the situation state and the decision variable. The coordination function $J_{\text{cf},i}$ models the contribution to communication latency as a function of segment length. In this particular problem, the cooperation constraint and the cooperation objective are coupled. As shown in [17], if the cooperation constraint is satisfied (by equalizing the path lengths), then the cooperation objective will be minimized (communication latency will be minimal). Because of this coupling, it is not necessary to explicitly share coordination functions among vehicles.

V. EXPERIMENTAL TESTBED

Over the past several years, BYU has developed a reliable and robust testbed for autonomous miniature air vehicles [56], [57]. Fig. 8 shows the key elements of our testbed. Fig. 8(a) shows BYU's Kestrel autopilot which is equipped with a Rabbit 3000 29 MHz processor, rate gyros, accelerometers, absolute and differential pressure sensors, and GPS. The autopilot measures only $1.5 \times 2.0 \times 0.75$ in and weighs 18 g. Fig. 8(b) shows the airframes used for the flight tests reported in this paper. The airframes are 48-in-wingspan Zagi XS EPP foam flying wings that were selected for their durability and adaptability to different mission scenarios. Embedded in the airframe are the autopilot, batteries, a 1000-mW 900-MHz radio modem, a GPS receiver, video transmitter, and a small analog camera. Fig. 8(c) shows the ground station components. A laptop runs the Virtual Cockpit software that interfaces through a communication box to the UAVs. An *RC* transmitter is used as a stand-by fail-safe mechanism to facilitate safe operations.

Cooperative flight tests with multiple UAVs are challenging to perform and development of this capability has required several years of experimentation and refinement. One significant challenge in working with air vehicles is that the design/test cycle is considerably longer than for ground robots. Flight experiments also entail a greater level of risk, since minor errors can lead to catastrophic crashes. In doing experiments with UAVs, crashes are inevitable. Choosing an airframe that survives

Fig. 8. *Hardware platform used to obtain experimental results. (a) Kestrel autopilot designed at BYU. (b) Airframes used for this paper. (c) Ground station components for our testbed.*

crashes greatly shortens the design and testing cycle. Experiments must be conducted in locations and at speeds where possible crashes do not pose a threat to life or property.

We have taken several steps to mitigate the logistical challenges of UAV flight experiments as follows.

1) *Airframe Selection*: The airframes that we use are designed to withstand crashes. Our airframes are constructed of EPP foam, Kevlar, and carbon fiber. Our workhorse airframe is the commercial off-the-shelf Zagi XS flying wing constructed of EPP foam [58]. The Zagi airframe was designed for midair *RC* dog fights and is extremely durable. It is inexpensive, hand-launchable, belly landable, and easy to construct, modify, and repair. The EPP foam exterior provides protection to the electronic components during crashes.
2) *Preflight Protocols*: Our standard preflight protocol includes a standard check list to test for the most common causes of crashes. Items on the list include testing the telemetry link, calibrating the sensors, and verifying proper deflection of control surfaces.
3) *In-Flight Parameter Changes*: We have developed a software architecture that allows any autopilot parameter to be modified in-flight. Parameters that can be modified in-flight include control loop gains and state estimation gains. Allowing parameters to be changed in-flight facilitates a rapid tune and debug cycle.
4) *Switchable Software Modules*: We have the capability to switch between different software modules in-flight. This has recently been used to test adaptive control algorithms on our UAVs. PID control was used during auto-takeoff and land but switched with an adaptive control module during flight. This capability allows rapid prototyping of algorithms without extensive validation and verification efforts.
5) *Data Logging and Display*: We can monitor and log any variable on the autopilot and display that data during the flight. Again, this capability allows algorithms to be debugged and tuned without requiring the UAV to be landed.
6) *Hardware-in-the-Loop Simulator*: We use a medium fidelity hardware-in-the-loop simulator that interfaces directly to the autopilot and runs in real time, allowing the autopilot code to be tested and debugged prior to flight experimentation.

In addition to the experimental capabilities described above, we developed two critical technologies before successfully demonstrating cooperative timing missions on UAVs: reliable intervehicle communication and robust trajectory tracking in wind.

Multiple-Vehicle Communication: One of the challenges with miniature UAVs is that size, weight, and power constraints make high-speed wide-bandwidth, digital communication links impractical. As an alternative, we use inexpensive low-power half-duplex 9600-baud radio-frequency modems, which require all vehicles to share the same communication channel and the use of a turn-taking scheme to avoid packet collisions.

Cooperative control experiments have two competing communication demands. First, telemetry data from each of the vehicles must be communicated to the ground station on a regular basis for data logging and system monitoring. Second, coordination information must be sent between vehicles and between the ground station and vehicles. From an individual vehicle communication perspective, there are two important requirements: low latency of high priority data transmission and confirmation of properly receiving the transmission. To meet these communication challenges, the ground station acts as a server and controls which vehicles can communicate what information at what time. The transmissions can be divided into three categories: vehicle telemetry data, commands for the vehicle, and coordination information between vehicles. The ground station attempts to request telemetry data from each of the vehicles at a rate based on the number of vehicles (typically 3 to 4 Hz) and all other data is transmitted on a needs basis. The ground station dynamically assigns a priority to each type of transmission based on urgency, time in queue, and previous frequency of transmission. This allows high-priority messages to be delivered with low latency but also prevents any type of transmission from flooding the communication channel. Although its bandwidth is suboptimal, this communication system meets the demanding requirements posed by cooperative UAVs and is very robust to packet loss.

Robust Trajectory Tracking in Wind: Airspeeds for the small UAVs that we fly are typically between 10 and 18 m/s, or equivalently 20–40 mi/h. A 10-mi/h wind, which is typical at a 300-ft altitude on calm days, is a substantial disturbance for a small UAV. For the cooperative timing experiments presented in this paper, the UAV is directed to be at a certain location at a specific time. Critical to completing the mission successfully are location and timing, which are significantly affected by wind. Experimental demonstration of cooperative timing requires high-fidelity trajectory tracking algorithms that mitigate the effects of wind.

For timing missions, preserving path length is critical in order to be able to satisfy the timing constraints. Flying extra distance due to poor path following adds unwanted time to the mission. To address these issues, a vector field path-following method has been implemented to enable the UAVs to accurately follow waypoint paths. It has been shown in [59] and [60] that creating vector fields of desired heading to direct the UAV onto its desired path will result in asymptotically decaying tracking error in the presence of wind, provided ground track heading and ground speed are used in the control law. Use of this

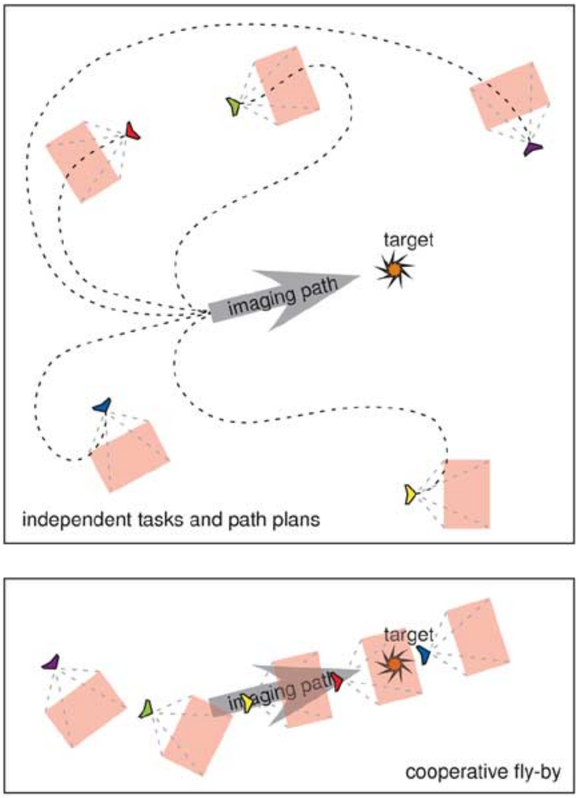

Fig. 9. *Persistent aerial surveillance. UAVs are initially performing an auxiliary task. Upon command, they coordinate their action to fly over target at fixed intervals of time.*

method has significantly decreased path following errors, and consequently overall path length errors, thus enabling successful cooperative timing experiments.

VI. DECENTRALIZED COOPERATIVE SURVEILLANCE

The objective of this section is to illustrate the design methodology introduced in Section III through a detailed example and to demonstrate the effectiveness of the approach using high-fidelity simulation and flight tests of fixed-wing miniature air vehicles. The design problem centers on cooperative aerial surveillance. We will consider two related variants of this problem. The first variant is persistent imaging, depicted in Fig. 9, where a team of N UAVs equipped with imaging sensors is tasked to persistently image a known target. If the field-of-view of the sensor is small with respect to the turning radius of the UAVs, the solution to this problem will require the team of UAVs to fly over the target at regular intervals. The second variant of the cooperative aerial surveillance problem is cooperative identification where a team of UAVs is required to fly over a target simultaneously, but along different approach angles. In the taxonomy introduced in Section II, the decentralized aerial surveillance problems are objectively coupled.

A. Solution Methodology

The first step in addressing the aerial surveillance problem is to identify the cooperation constraint. Let $\hat{z}$ represent the location of the target which we assume is known to every vehicle, and let $z_i(t)$ denote the location of the ith vehicle at time t. The cooperation constraint is given by

$$J_{\text{constraint}}(\theta^*) = \sum_{i=1}^{N} \|z_i(\theta^* + \gamma_i \Delta) - z\|^2 \qquad (9)$$

where Δ is the desired spacing and $\gamma_i = (k-1)$ when the ith vehicle is assigned the kth position in the surveillance sequence. Note that if $J_{\text{constraint}} = 0$, then the vehicles fly by the target location at equally spaced intervals Δ seconds apart.

In (9), θ^* is the time that the first vehicle passes over the target. It is clear that by increasing the loiter time, θ^* can be made arbitrarily large. Therefore, we need to introduce an auxiliary optimization criteria that selects between the many possibilities for θ^*. Toward that end, we assume that a suitable path planning algorithm is available for planning waypoint paths from the current location of the UAV z_i to the target z in a certain time T. The algorithm will be denoted by the notation

$$\mathcal{W} = \text{planPath}(z_i, \hat{z}, T).$$

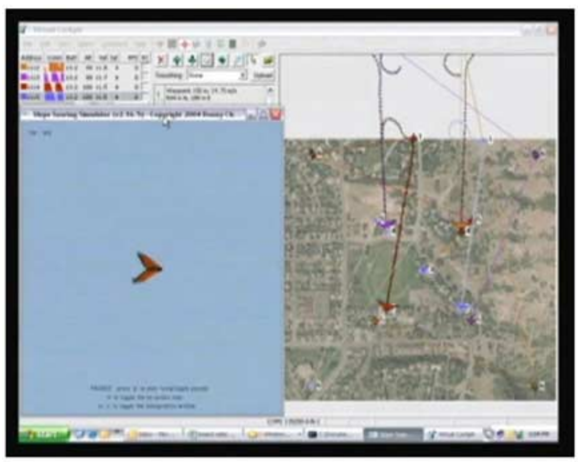

Fig. 10. *Screen shot of flight simulator (bottom left window in blue) and virtual cockpit (background). Simulation environment enables rapid prototyping of cooperative control problems for autonomous miniature air vehicles.*

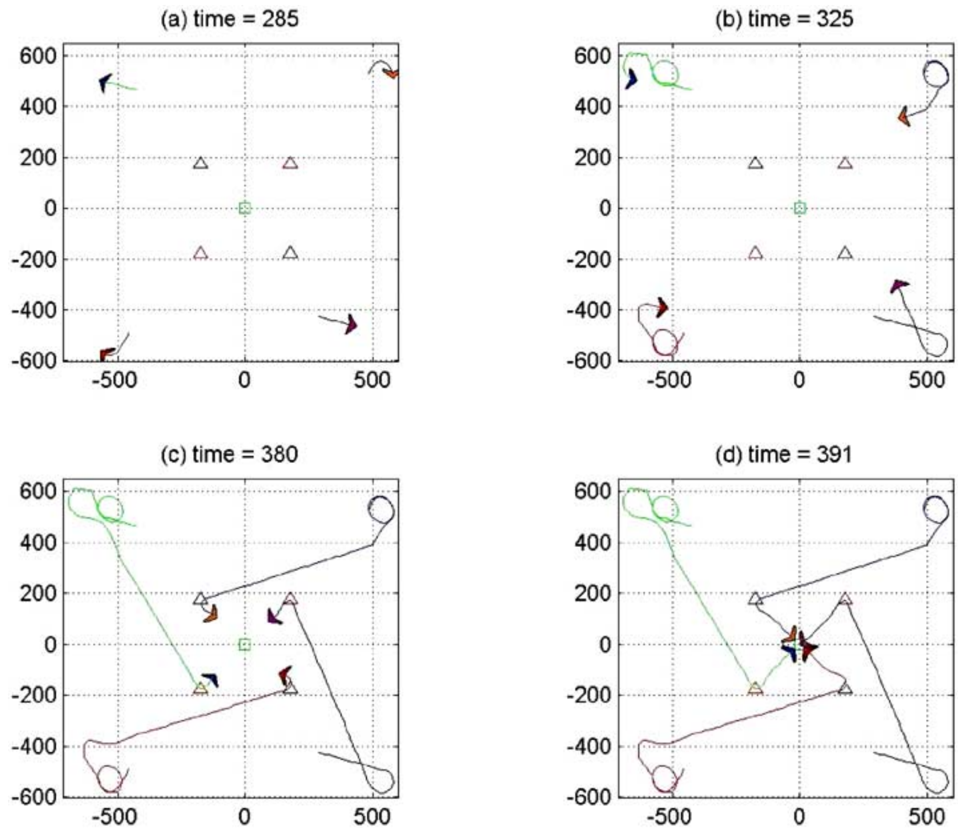

Fig. 11. ***Simulation results of cooperative identification mission. (a) UAVs are loitering around specified waypoint. (b) Mission execution command is issued and UAVs plan their approach trajectories. (c) and (d) Execution of mission. UAV are flying at distinct altitudes.***

In addition, we assume a function Length($\mathcal{W}$) that returns the path length of $\mathcal{W}$. The fuel expended in traversing a path is approximated by

$$\text{fuel} = c_f v_i \times \text{Length}(\text{planPath}(z_i, \hat{z}, T))$$

where v_i is the airspeed along the path and c_f is a constant. Therefore, fuel will be minimized by selecting the cooperation objective as

$$J_{\text{objective}}(\theta^*) = \sum_{i=1}^{N} c_f v_i \times \text{Length} \times (\text{planPath}(z_i(t_0), \hat{z}, t_0 + \theta^* + \gamma_i \Delta)) \quad (10)$$

where t_0 is the time at which the path planner is executed.

The second step is to make a suitable choice for the coordination variable. It is clear from the discussion above that the instant in time that the first vehicle passes over the target is a suitable coordination variable. As seen in (9) and (10), the cooperation constraint and the cooperation

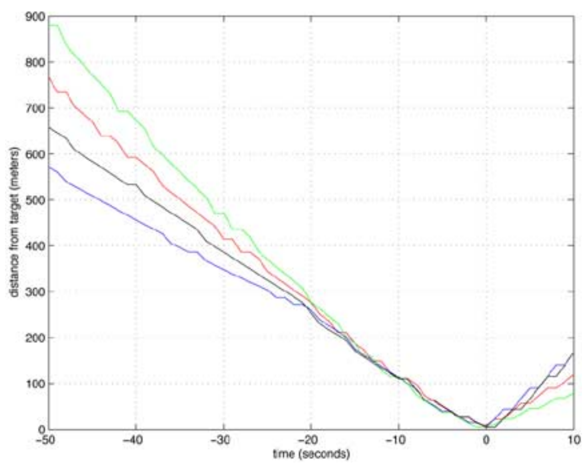

Fig. 12. ***Plot of average distance from target, verses time, for simulated cooperative identification problem.***

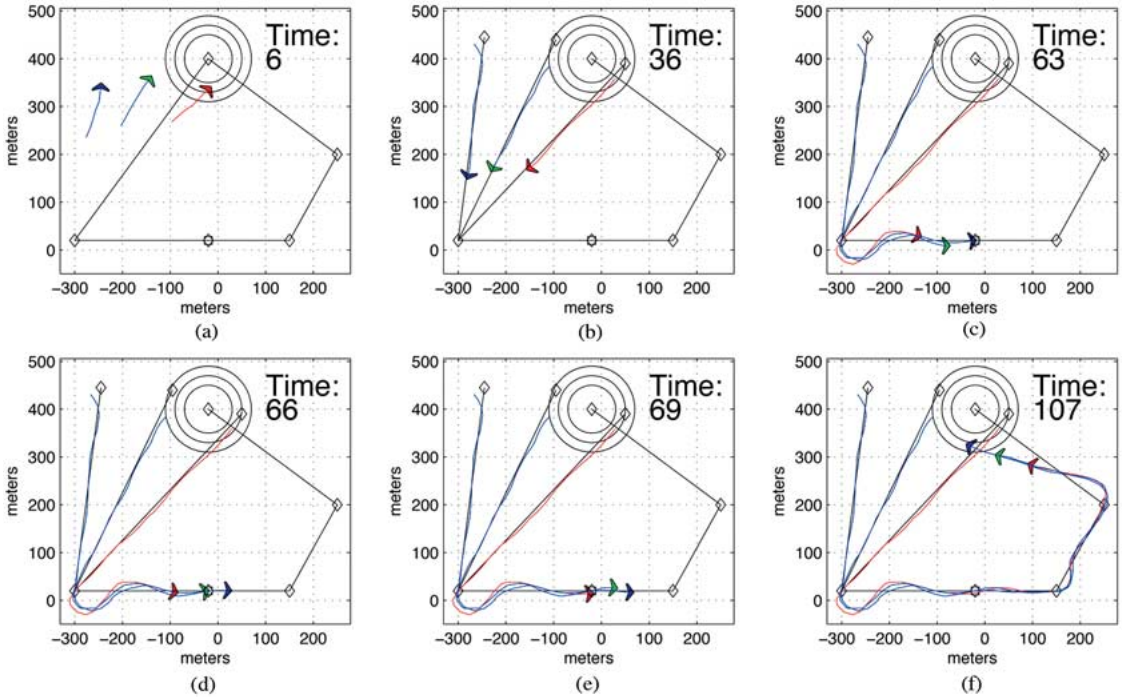

Fig. 13. ***Telemetry data for flight tests of persistent imaging mission. (a) UAVs approaching loiter waypoint. (b) UAVs en route to entry waypoint after fly-by command has been issued. (c), (d), and (e) UAVs transition through imaging target. (f) UAVs returning to their loiter points.***

objective can be written in terms of the coordination variable. The coordination function is given by

$$J_{\mathrm{cf},i}(\theta^*) = c_f v_i \times \mathrm{Length}\big(\mathrm{planPath}(z_i(t_0), \hat{z}, t_0 + \theta^* + \gamma_i \Delta)\big).$$

The third step is to devise a centralized cooperation algorithm that solves (3). The formulation of the path planning problem ensures that the cooperation constraint is trivially satisfied. The objective function can be optimized with an MILP solver where γ_i are integers and θ^* is

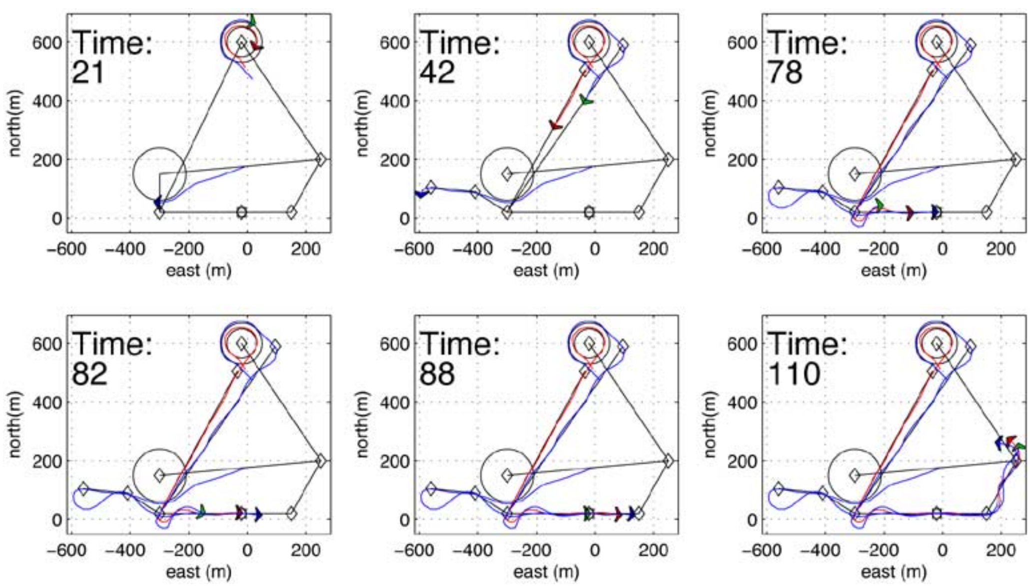

Fig. 14. ***Telemetry data for persistent imaging when one UAV is significantly closer to image point than rest of team.***

real. As it turns out, this particular problem has sufficient structure and it admits an analytic solution [30]. Once θ^* has been determined (by a centralized unit), then, according to (4), the ith vehicle implements the path given by

$$u_i = \text{planPath}(z_i(t_0), \hat{z}, t_0 + \theta^* + \gamma_i \Delta). \tag{11}$$

The final step is to decentralize the algorithm using a consensus scheme. To do so, let θ_i denote the ith vehicles instantiation of the coordination variable θ^*. Instead of (11), the ith vehicle implements

$$u_i = \text{planPath}(z_i(t_0), \hat{z}, t_0 + \theta_i + \gamma_i \Delta).$$

Given communication with the other vehicles, the coordination variable θ_i is then adjusted according to the consensus dynamics given in (5).

In the cooperative identification variant of this problem, we desire all vehicles to arrive at the target location simultaneously and at different approach angles. For this problem, we set $\Delta = 0$ and pass the approach angle as an additional input to the path planning algorithm. In the next two sections, we will present simulation results for the decentralized cooperative identification problem and flight test results for the centralized persistent imaging and cooperative identification problem.

B. Simulation Results

To enable rapid prototyping of cooperative control algorithms, we have developed a medium-fidelity simulation environment for autonomous miniature air vehicles.

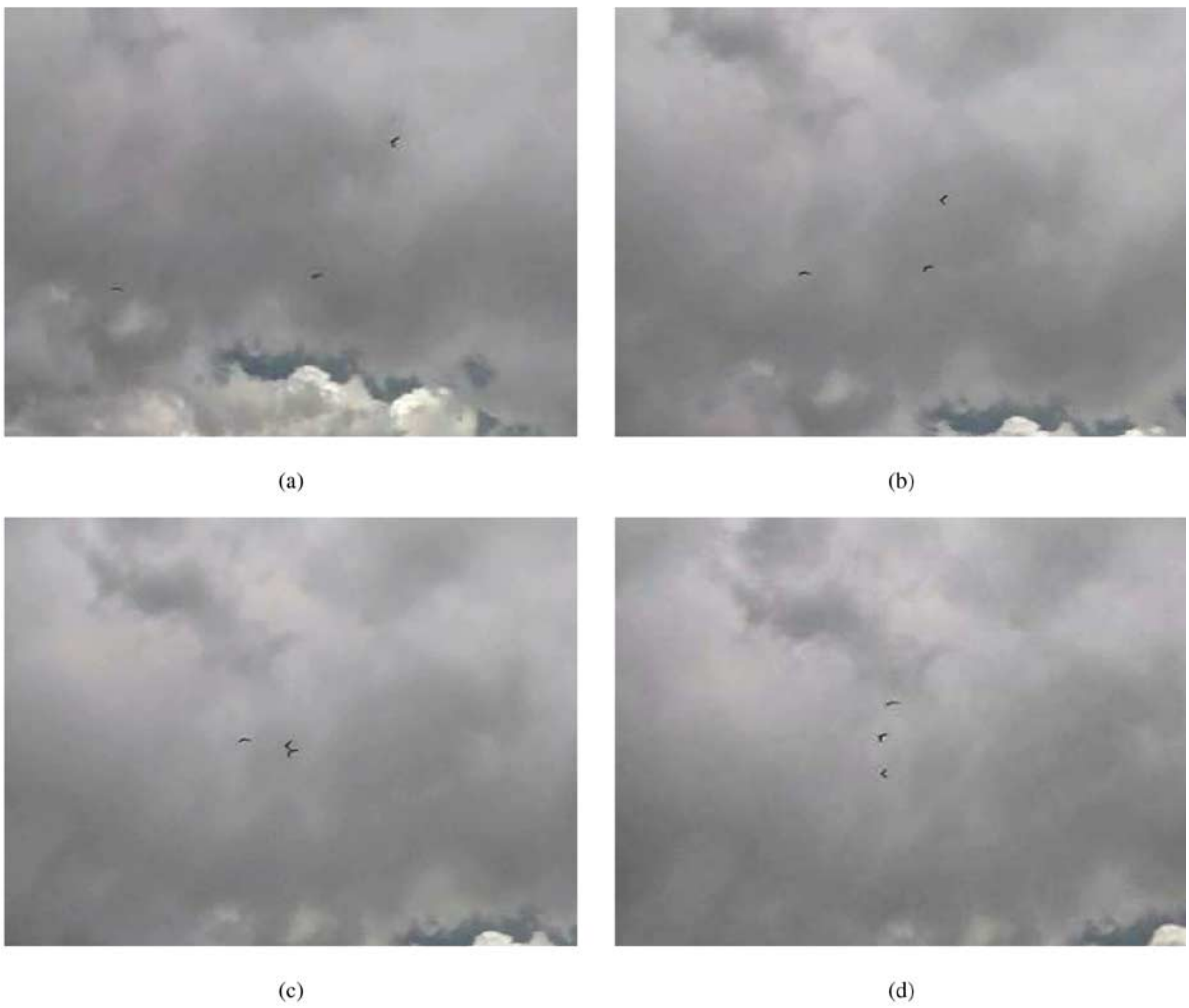

Fig. 15. ***Still shots of cooperative timing scenario. (a) and (b) Three UAVs approaching rendezvous point. (c) Rendezvous to within 0.3 s. (d) UAVs immediately after rendezvous.***

The simulation environment consists of two components. The first is a 6-DOF flight simulator with digital elevation model (DEM) terrain data [61] and realistic wind models. The second component is an autopilot module that executes the same code that is implemented on the physical autopilot. The autopilot module connects to the same ground station software that is used to fly the miniature air vehicles. A screen shot of the flight simulator and the virtual cockpit are shown in Fig. 10.

The cooperative identification problem was simulated using four UAVs that were tasked to arrive at the target simultaneously with arrival angles differing by 90°. The average time to reach consensus to within 0.02 units was 6.2 s, where one communication packet per second was allowed to be sent by each UAV to another UAV selected randomly from the team. Simulation results are shown in Fig. 11. Subplots (a) and (b) show the four UAVs loitering until the mission execution command is issued at the end of subplot (b). The UAVs are all flying at distinct, preassigned altitudes to avoid collision as they pass over the target. Subplots (c) and (d) show the UAVs executing the cooperative identification mission.

Fig. 12 plots the average distance from the target verses time for each UAV and demonstrates the effectiveness of the decentralized coordination algorithm.

C. Flight Tests

Flight tests were performed using three UAVs that were commanded to persistently image a target with fixed intervals of $\Delta = 3$ s. The three UAVs were initially commanded to loiter at specified GPS coordinates. The UAVs were then commanded to perform a persistent imaging mission and then return to their loiter coordinates. The mission was repeated six times during a 30-min

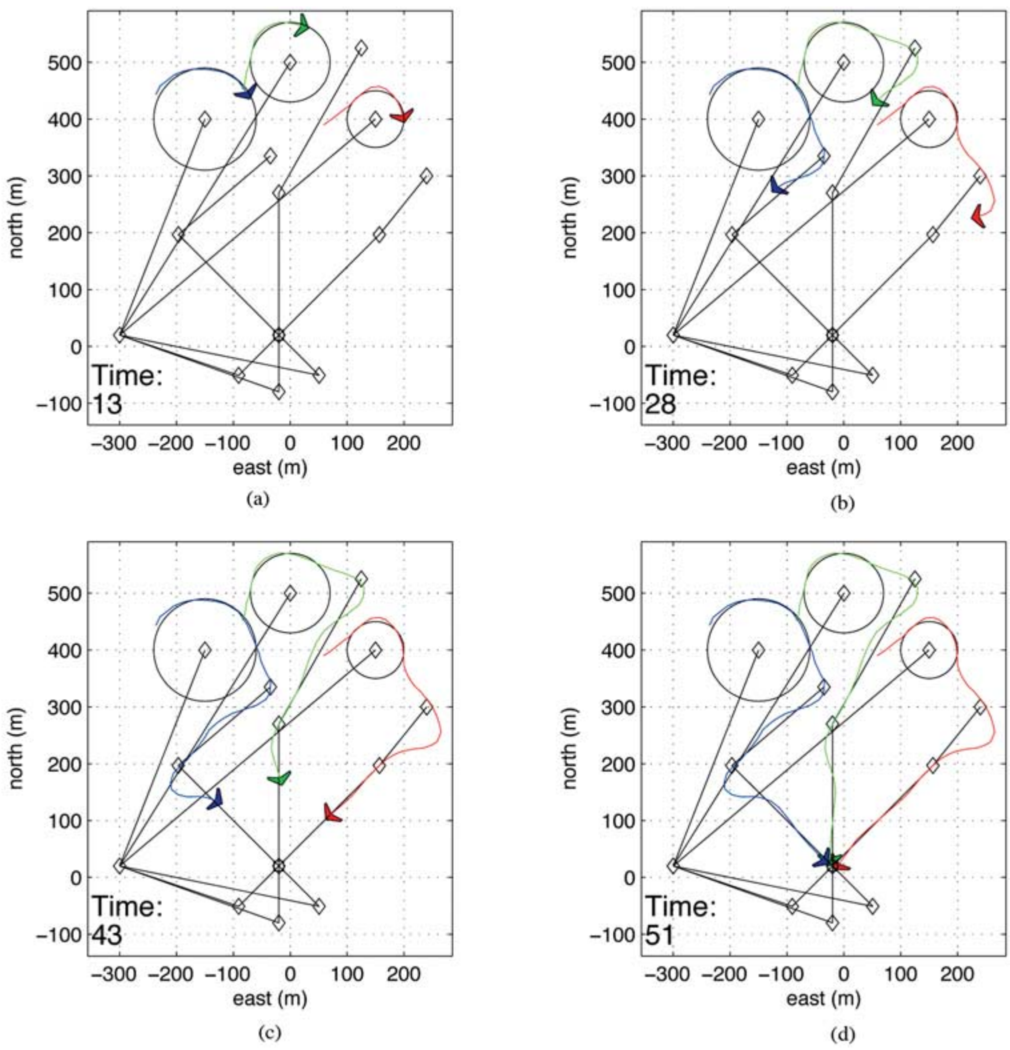

Fig. 16. ***Telemetry data for simultaneous arrival mission. (a) UAVs have just received cooperative rendezvous signal. (b) UAVs are orbiting prescribed loiter points. (c) UAVs en route to rendezvous point. (d) UAVs at rendezvous point.***

flight. The wind speeds during the flight were between 30% and 60% of the UAV airspeed and were from the southwest. The average error in arrival time for the six runs, in spite of the high wind conditions, was approximately 0.6 s.

Plots of the telemetry data are shown in Fig. 13. The imaging target is depicted by a circle, whereas the transition waypoints produced by the path planner are depicted by diamonds. Fig. 14 shows the case where one of the UAVs starts the mission from a significantly different initial target distance than the other vehicles. Note that the vehicle that is closest to the imaging point must plan a path that increases its distance, thus allowing the rest of the team to catch up with the vehicle.

In addition to the persistent imaging scenario, we have also flight tested the simultaneous arrival mission using three UAVs. Still shots of an arrival mission are shown in Fig. 15. Fig. 15(a) and (b) shows the UAVs as they approach the cooperative rendezvous point. Fig. 15(c) shows the simultaneous arrival to within 0.3 s, and (d) shows the UAV departing the rendezvous point. Telemetry plots for this mission are shown in Fig. 16.

VII. CONCLUSION

In this paper, we have provided an overview of a general design methodology for multiple vehicle decentralized cooperative control problems. The methodology consists of four main design steps. The first step is to quantify the meaning of *cooperation* as a function $J_{\text{constraint}}$ that is a positive definite function of the situation state and the decision variables. In addition, auxiliary optimization criteria may be encoded as a positive definite function $J_{\text{objective}}$ called the cooperation objective function. The second step is to isolate the information that is essential for pair-wise coordination and to call this information the coordination variable θ^*. The cooperation constraint $J_{\text{constraint}}$ and the cooperation objective $J_{\text{objective}}$ are then formulated in terms of the coordination variables. The third step is to design a centralized cooperation algorithm that solves the optimization problem given in (3) and determines a decision variable as in (4). The cooperation algorithm must be designed so that the mapping from perturbations in the coordination variable to the cooperation constraint are input-to-state stable. The final step is to instantiate the cooperation algorithm on every vehicle and then implement a consensus-seeking strategy that guarantees that the team will come into agreement on the coordination variable if there is sufficient communication.

We have illustrated the approach with a simple example and by sketching the application of other cooperation problems that we have addressed. We also described in some detail the application to cooperative aerial surveillance using fixed-wing miniature UAVs and provided flight test results. ■

REFERENCES

[1] S. Akella and S. Hutchinson, "Coordinating the motions of multiple robots with specific trajectories," in *Proc. IEEE Int. Conf. Robotics Automation*, Washington, DC, 2002, pp. 624–631.

[2] W. Blake and D. Multhopp, "Design, performance and modeling considerations for close formation flight," *Proc. AIAA Navigation, Guidance Control Conf.*, 1998, pp. 476–486.

[3] D. F. Chichka and J. L. Speyer, "Solar-powered, formation-enhanced aerial vehicle systems for sustained endurance," in *Proc. Amer. Control Conf.*, Philadelphia, PA, Jun. 1998.

[4] A. W. Proud, M. Pachter, and J. J. D'Azzo, "Close formation flight control," *Proc. AIAA Guidance, Navigation, Control Conf. Exhibit*, 1999, pp. 1231–1246.

[5] M. Pachter, J. J. D'Azzo, and A. W. Proud, "Tight formation flight control," *AIAA J. Guidance, Control Dynamics*, vol. 24, no. 2, pp. 246–254, Mar.–Apr. 2001.

[6] C. Schumacher and S. N. Singh, "Nonlinear control of multiple UAVs in close-coupled formation flight," in *Proc. AIAA Guidance, Navigation, and Control Conf.*, Denver, CO: Amer. Inst. Aeronautics Astronautics, Aug. 2000, paper no. AIAA 2000-4373.

[7] M. R. Anderson and A. C. Robbins, "Formation flight as a cooperative game," in *Proc. AIAA Guidance, Navigation Control Conf.*, Boston, MA: Amer. Inst. Aeronautics Astronautics, Aug. 1998, pp. 244–251, Paper AIAA-98-4124.

[8] S. T. Pledgie, Y. Hao, A. M. Ferreira, S. K. Agrawal, and R. Murphey, "Groups of unmanned vehicles: Differential flatness, trajectory planning, and control," in *Proc. IEEE Int. Conf. Robotics Automation*, Washington, DC, May 2002, pp. 3461–3466.

[9] F. Giulietti, L. Pollini, and M. Innocenti, "Autonomous formation flight," *IEEE Control Syst. Mag.*, vol. 20, no. 6, pp. 34–44, Dec. 2000.

[10] J. A. Fax and R. M. Murray. (2001, Jul.). Graph Laplacians and stabilization of vehicle formations, Eng. Applied Sci., California Inst. Technol., Pasadena, Tech. Rep. CDS 01-007. [Online]. Available: http://www.cds.caltech.edu/~murray/cgi-bin/htdblist.cgi?papers/config.db

[11] ——, "Graph Laplacians and stabilization of vehicle formations," in *Proc. IFAC World Congr.*, Barcelona, Spain, 2002.

[12] P. Chandler, S. Rasumussen, and M. Pachter, "UAV cooperative path planning," presented at the AIAA Guidance, Navigation and Control Conf., Denver, CO, Aug. 2000, AIAA Paper AIAA-2000-4370.

[13] T. W. McLain and R. W. Beard, "Coordination variables, coordination functions, and cooperative timing missions," in *Proc. Amer. Control Conf.*, Denver, CO, Jun. 2003, pp. 296–301.

[14] J. Bellingham, M. Tillerson, A. Richards, and J. P. How, "Multi-task allocation and path planning for cooperating UAV," in *Cooperative Control: Models, Applications Algorithms. Conference Coordination, Control Optimization*, Nov. 2001, pp. 1–19.

[15] R. W. Beard, T. W. McLain, M. Goodrich, and E. P. Anderson, "Coordinated target assignment and intercept for unmanned air vehicles," *IEEE Trans. Robot. Automat.*, vol. 18, no. 6, pp. 911–922, Dec. 2002.

[16] T. McLain and R. Beard, "Cooperative rendezvous of multiple unmanned air vehicles," presented at the AIAA Guidance, Navigation Control Conf., Denver, CO, Aug. 2000, Paper AIAA-2000-4369.

[17] D. W. Casbeer, D. B. Kingston, R. W. Beard, T. W. McLain, S.-M. Li, and R. Mehra, "Cooperative forest fire surveillance using a team of small unmanned air vehicles," *Int. J. Syst. Sci.*, vol. 37, no. 6, pp. 351–360, May 2006.

[18] D. B. Kingston, R. S. Holt, R. W. Beard, T. W. McLain, and D. W. Casbeer, "Decentralized perimeter surveillance using a team of UAVs," in *Proc. AIAA Guidance, Navigation, Control Conf.*, Aug. 2005.

[19] J. Bellingham, M. Tillerson, A. Richards, and J. P. How, Multi-task allocation and path planning for cooperating UAVs," in *Cooperative Control: Models, Applications Algorithms.* Boston, MA: Kluwer, Jan. 2003, ch. 2.

[20] A. Richards, J. Bellingham, M. Tillerson, and J. How, "Coordination and control of UAVs," presented at the AIAA Guidance, Navigation Control Conf., Monterey, CA, Aug. 2002, AIAA-2002-4588.

[21] M. Alighanbari, Y. Kuwata, and J. How, "Coordination and control of multiple UAVs with timing constraints and loitering," in *Proc. Amer. Control Conf.*, Jun. 2003, pp. 5311–531.

[22] E. Frazzoli, M. A. Dahleh, and E. Feron, "Real-time motion planning for agile autonomous vehicles," *J. Guidance, Contr., Dynamics*, vol. 25, no. 1, pp. 116–129, Jan.–Feb. 2002.

[23] N. Faiz, S. K. Agrawal, and R. M. Murray, "Trajectory planning of differentially flat systems with dynamics and inequalities," *AIAA J. Guidance, Control Dynamics*, vol. 24, no. 2, pp. 219–227, Mar.–Apr. 2001.

[24] O. A. Yakimenko, "Direct method for rapid prototyping of near-optimal aircraft trajectories," *AIAA J. Guidance, Control Dynamics*, vol. 23, no. 5, pp. 865–875, Sep.–Oct. 2000.

[25] G. Yang and V. Kapila, "Optimal path planning for unmanned air vehicles with kinematic and tactical constraints," in *Proc. IEEE Conf. Decision Control*, Las Vegas, NV, 2002, pp. 1301–1306.

[26] E. P. Anderson and R. W. Beard, "An algorithmic implementation of constrained extremal control for UAVs," presented at the AIAA Guidance, Navigation Control Conf., Monterey, CA, Aug. 2002, AIAA Paper 2002-4470.

[27] G. Inalhan, D. M. Stipanovic, and C. J. Tomlin, "Decentralized optimization with application to multiple aircraft coordination," in *Proc. IEEE Conf. Decision Control*, Las Vegas, NV, 2002, pp. 1147–1155.

[28] S. Sastry, G. Meyer, C. Tomlin, J. Lygeros, D. Godbole, and G. Pappas, "Hybrid control in air traffic management systems," in *Proc. 34th IEEE Conf. Decision Control*, New Orleans, LA, 1995, pp. 1478–1483.

[29] J. K. Howlett, "Path planning for sensing multiple targets from an aircraft," Master's thesis, Mechanical Eng., Brigham Young Univ., Provo, UT, 2002.

[30] T. W. McLain and R. W. Beard, "Coordination variables, coordination functions, and cooperative timing missions," *AIAA J. Guidance, Contr. Dynamics*, vol. 28, no. 1, pp. 150–161, Jan. 2005.

[31] W. Kang and H.-H. Yeh, "Coordinated attitude control of multi-satellite systems," *Int. J. Robust Nonlinear Contr.*, vol. 12, pp. 185–205, 2002.

[32] W. Kang, N. Xi, and A. Sparks, "Formation control of autonomous agents in 3-D workspace," in *Proc. IEEE Int. Conf. Robotics Automation*, San Francisco, CA, Apr. 2000, pp. 1755–1760.

[33] S. Sheikholeslam and C. A. Desoer, "Control of interconnected nonlinear dynamical systems: The platoon problem," *IEEE Trans. Automat. Contr.*, vol. 37, no. 6, pp. 806–810, Jun. 1992.

[34] P. K. C. Wang and F. Y. Hadaegh, "Coordination and control of multiple microspacecraft moving in formation," *J. Astronautical Sci.*, vol. 44, no. 3, pp. 315–355, 1996.

[35] R. W. Beard, J. Lawton, and F. Y. Hadaegh, "A feedback architecture for formation control," *IEEE Trans. Contr. Syst. Technol.*, vol. 9, no. 6, pp. 777–790, Nov. 2001.

[36] J. Lawton and R. Beard, "A projection approach to spacecraft formation attitude control," presented at the 23rd Annu. AAS Guidance Control Conf., Breckenridge, Colorado, Feb. 2000, Paper AAS 00-011.

[37] N. E. Leonard and E. Fiorelli, "Virtual leaders, artificial potentials and coordinated control of groups," in *Proc. IEEE Conf. Decision Control*, Orlando, FL, Dec. 2001, pp. 2968–2973.

[38] D. J. Stilwell and B. E. Bishop, "Platoons of underwater vehicles," *IEEE Contr. Syst. Mag.*, vol. 20, no. 6, pp. 45–52, Dec. 2000.

[39] R. Emery, K. Sikorski, and T. Balch, "Protocols for collaboration, coordination and dynamic role assignment in a robot team," in *Proc. IEEE Int. Conf. Robotics Automation*, Washington, DC, May 2002, pp. 3008–3015.

[40] W. Ren, R. W. Beard, and T. W. McLain, "Coordination variables and consensus building in multiple vehicle systems," in *Cooperative Control* vol. 309, *Lecture Notes in Control and Information Systems*, V. Kumar, N. Leonard, A. S. Morse, Eds. Block Island, RI: Springer-Verlag, 2004, pp. 171–188.

[41] 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, Sep. 2004.

[42] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. Automat. Contr.*, vol. 48, no. 6, pp. 988–1001, Jun. 2003.

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

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

[45] 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.

[46] 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 Cooperative Control*, vol. 309, *Springer-Verlag Series: Lecture Notes in Control and Information Sciences*, V. Kumar, N. E. Leonard, and A. S. Morse, Eds., 2004, pp. 171–188.

[47] W. Ren and R. W. Beard, "Consensus seeking in multi-agent systems under dynamically changing interaction topologies," *IEEE Trans. Automat. Contr.*, vol. 50, no. 5, pp. 655–661, May 2005.

[48] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Syst. Contr. Lett.*, vol. 53, pp. 65–78, 2004.

[49] 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.

[50] R. W. Beard and T. W. McLain, "Multiple UAV cooperative search under collision avoidance and limited range communication constraints," in *Proc. IEEE Conf. Decision Control*, Maui, HI, Dec. 2003, pp. 25–30.

[51] C. Schumacher, P. R. Chandler, and S. J. Rasmussen, "Task allocation for wide area search munitions via network flow optimization," in *Proc. AIAA Guidance, Navigation Control Conf.*, Montreal, Canada, 2001.

[52] W. Ren and R. W. Beard, "A decentralized scheme for spacecraft formation flying via the virtual structure approach," in *Proc. Amer. Control Conf.*, Denver, CO, Jun. 2003, pp. 1746–1751.

[53] F. L. Lewis, *Optimal Control*. New York: Wiley, 1986.

[54] D. B. Kingston, W. Ren, and R. W. Beard, "Consensus algorithms are input-to-state stable," in *Proc. Amer. Control Conf.*, Portland, OR, Jun. 2005, pp. 1686–1690.

[55] H. K. Khalil, *Nonlinear Systems, 3rd Ed. Upper Saddle River, NJ: Prentice–Hall, 2002.*

[56] R. Beard, D. Kingston, M. Quigley, D. Snyder, R. Christiansen, W. Johnson, T. McLain, and M. Goodrich, "Autonomous vehicle technologies for small fixed wing UAVs," *AIAA J. Aerospace, Comput., Information, Commun.*, vol. 2, no. 1, pp. 92–108, Jan. 2005.

[57] R. W. Beard, D. Lee, M. Quigley, S. Thakoor, and S. Zornetzer, "A new approach to observation of descent and landing of future Mars mission using bioinspired technology innovations," *AIAA J. Aerospace Comput., Information, Commun.*, vol. 2, no. 1, pp. 65–91, Jan. 2005.

[58] [Online]. Available: http://zagi.com

[59] D. R. Nelson, D. B. Barber, T. W. McLain, and R. W. Beard, "Vector field path following for small unmanned air vehicles," in *Proc. American Control Conf.*, 2006, pp. 5788–5794.

[60] D. R. Nelson, "Cooperative control of miniature air vehicles," Master's thesis, Brigham Young Univ., Provo, UT, Dec. 2005.

[61] USGS Digital Elevation Models. [Online]. Available: http://data.geocomm.com/dem/

ABOUT THE AUTHORS

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, N.Y.

Since 1996, he has been with the Electrical and Computer Engineering Department, 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. His research interests include micro air vehicles, autonomous systems, and cooperative control.

Dr. Beard is an Associate Editor for the IEEE Control Systems Society Conference Editorial Board and for the *IEEE Control Systems Magazine*.

Timothy W. McLain received the B.S. degree in mechanical engineering, in 1986, and the M.S. degree in mechanical engineering, in 1987, from Brigham Young University, Provo, UT. In 1995, he received the Ph.D. degree in mechanical engineering from Stanford University, Stanford, CA.

From 1987 to 1989, he was employed at the Center for Engineering Design, University of Utah. Since 1995, he has been with the Department of Mechanical Engineering, Brigham Young University, Provo, UT, where he is currently an Associate Professor. In the summers of 1999 and 2000, he was a Visiting Scientist at the Air Force Research Laboratory, Air Vehicles Directorate. His research interests include cooperative control of multiple vehicle systems and modeling and control of microelectromechanical systems.

Dr. McLain is a member of the AIAA Unmanned Systems Program Committee.

Derek B. Nelson received the B.S. degree in mechanical engineering, in 1994, and the M.S. degree in mechanical engineering, in 1995, from Brigham Young University, Provo, UT.

He is currently employed at the Northop Grumman Corporation, San Diego, CA, and works in the guidance, navigation, and controls group on the Global Hawk project.

Derek Kingston received the B.S. and M.S. degrees in electrical engineering from Brigham Young University, Provo, UT, in 2002 and 2004, respectively. He is currently pursuing the Ph.D. degree in electrical engineering at the same university.

His research interests include decentralized control and state estimation.

David Johanson received the B.S. degree in computer engineering, in 2004, from Brigham Young University, Provo, UT and is currently pursuing the M.S. degree in electrical engineering from the same university.

His research interests include computer vision, autonomous systems, and multithreaded high-performance computing.