Decentralized Receding Horizon Control and Coordination of Autonomous Vehicle Formations

Tamás Keviczky, *Member, IEEE*, Francesco Borrelli, *Member, IEEE*, Kingsley Fregene, *Member, IEEE*, Datta Godbole, *Member, IEEE*, and Gary J. Balas, *Fellow, IEEE*

Abstract—**This paper describes the application of a novel methodology for high-level control and coordination of autonomous vehicle teams and its demonstration on high-fidelity models of the organic air vehicle developed at Honeywell Laboratories. The scheme employs decentralized receding horizon controllers that reside on each vehicle to achieve coordination among team members. An appropriate graph structure describes the underlying communication topology between the vehicles. On each vehicle, information about neighbors is used to predict their behavior and plan conflict-free trajectories that maintain coordination and achieve team objectives. When feasibility of the decentralized control is lost, collision avoidance is ensured by invoking emergency maneuvers that are computed via invariant set theory.**

Index Terms—**Constrained optimization, formation flight, hierarchical decomposition, receding horizon control, set invariance.**

I. Introduction

INTEREST in the coordination and control of various classes of unmanned autonomous vehicles has grown significantly over the past decade. These vehicles include unmanned ground vehicles (UGVs), unmanned underwater vehicles (UUVs), and unmanned air vehicles (UAVs). The main motivation for this trend is the wide range of military and civilian applications where teams of these vehicles working together have the potential of providing a low-cost and efficient alternative to existing technology. For instance, several distributed sensing applications are envisioned for, but not limited to, the UAV domain including synthetic aperture radar (SAR) interferometry, surveillance, damage assessment, reconnaissance, chemical or biological agent monitoring, exploration, vegetation growth analysis, assessment of topographical changes, etc. [1]. Many of these applications often necessitate the development of control system design techniques for large and tight UAV formations. Indeed, formation control and reconfiguration has become a core problem whose efficient and flexible solution will positively impact many distributed sensing applications envisaged for autonomous vehicles. We will use UAV formation flight to motivate much of the discussion in the remainder of this paper.

From a control engineering perspective, formation flight can be seen as a large control problem in which the objective is to compute the inputs that drive the vehicles along trajectories which maintain relative positions as well as safe distances between each UAV pair. Several approaches to formation control have been proposed in the literature (see [2]–[4] and references therein). Other techniques that have been applied to this problem include linear matrix inequalities (LMI) [1], control Lyapunov functions [5], and vision-based frameworks [6]. Optimal control problem formulation has been one of the most used frameworks for tackling this problem [1], [7]–[12]. In this framework, formation flight is formulated as a minimization of the error between UAV relative distances and desired displacements. Such a formulation allows collision avoidance requirements to be easily included as additional constraints between each UAV pair. Although centralized optimal or suboptimal approaches have been used in different studies (see, for instance, [9] and [10]), it is clear that, as the number of vehicles increases, the solution of such large-scale, centralized, nonconvex optimization problems becomes prohibitive. This is true even when the most advanced optimization solvers and much simplified linear vehicle dynamics are used [13]. The main challenge is to formulate simpler decentralized problems which result in a formation behavior similar to what is obtainable with a centralized approach.

The great body of decentralized control examples [14]–[22], deals with dynamically coupled systems. Motivated by the type of vehicles and formation tasks considered in this paper, we neglect dynamical coupling between individual vehicles. We are interested in decentralized control for dynamically decoupled and independently actuated systems. Most of the current decentralized control design techniques for formation control have been successfully developed for oversimplified vehicle dynamics without explicitly taking constraints into account [23]–[28]. Even where constraints are considered, this is done only implicitly. For instance, collision avoidance constraints are usually translated into minimization of barrier/potential functions whose value becomes very high when a direction which leads to collisions is chosen. Such an approach does not account for real-life issues like limited speed, acceleration, and actuation authority of the vehicle. Moreover, once a certain formation scheme has been designed, they are quite difficult or impossible to reconfigure on-the-fly (for instance, when

Manuscript received June 1, 2005; revised September 20, 2006. Manuscript received in final form March 20, 2007. Recommended by Associate Editor D. A. Schoenwald. The work of T. Keviczky and G. J. Balas was supported in part by the Defense Advanced Research Projects Agency under the Software Enabled Control Program (Dr. John Bay Program Manager) under Contract USAF/AFMC F33615-99-C-1497 (Dale Van Cleave, Technical Contract Monitor).

T. Keviczky is with Delft Center for Systems and Control, Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: t.keviczky@tudelft.nu).

F. Borrelli is with the Dipartimento di Ingegneria, Università degli Studi del Sannio, 82100 Benevento, Italy (e-mail: francesco.borrelli@unisannio.it).

K. Fregene and D. Godbole are with the Honeywell Laboratories, Minneapolis, MN 55418 USA (e-mail: kingsley.fregene@honeywell.com; datta.godbole@honeywell.com).

G. J. Balas is with the Department of Aerospace Engineering and Mechanics, University of Minnesota, Minneapolis, MN 55455 USA (e-mail: balas@aem.umn.edu).

Digital Object Identifier 10.1109/TCST.2007.903066

previously unknown obstacles pop up in the path of the vehicle formation). The scheme proposed in this paper attempts to overcome these limitations and other inherent difficulties associated with achieving formation flight for constrained, highly nonlinear autonomous vehicles via a hierarchical decomposition of the problem.

In such a decomposition, the lower level comprises the UAV dynamics equipped with efficient guidance and control loops. At the higher level, the controlled UAV can be represented sufficiently well as a constrained multi-input–multi-output (MIMO) linear or piecewise linear system. For this class of systems, a decentralized constrained optimization-based control framework is constructed based on [29] and [30]. Specifically, receding horizon control (RHC) is utilized to achieve formation flight and other cooperative tasks. Within this framework, individual vehicles use neighbor information to predict their behavior in order to avoid collisions and act cooperatively. The proposed approach can handle constrained MIMO linear models as well as constrained MIMO piecewise linear models of unmanned vehicles. The optimization problem is formulated and solved as small mixed-integer linear programs (MILPs) which can be translated into equivalent gain scheduled controllers requiring significantly lower computational effort that enables real-time implementation. Furthermore, different maneuvering objectives such as formation keeping, formation splitting, and joining are dynamically attained by changing appropriate terms in the cost function. The procedure is also robust to changing number of vehicles in the team since individual vehicles may rely on information from any number of neighbors in order to determine their control policies. These benefits are demonstrated on a high-fidelity model of the organic air vehicle (OAV) developed at Honeywell Laboratories, Minneapolis, MN.

This paper is organized into two parts. The first part introduces the OAV dynamics and develops the decentralized control scheme, while the second part demonstrates its application to OAV formation flight.

Section II briefly describes the dynamics of the OAV followed by its lower-level guidance and control design in Section III. Section IV gives an overview of the decentralized constrained optimization-based approach proposed in [30], which is used in this paper for higher-level coordinated controller synthesis in a modified form. Its main features and advantages are highlighted, along with major issues and unsolved problems. Collision avoidance is addressed in Section V. Section V-A presents results in invariant set theory that form the basis of a collision-free extension of the basic decentralized scheme described in Sections V-B and V-C. Practical considerations and computational tools that facilitate the solution approach are highlighted in Section V-D. Simulation results carried out on high-fidelity models of the vehicles are presented in Section VI. The simulation examples are followed by concluding remarks in Section VII.

II. OAV Dynamics

The OAV is a scalable autonomous ducted-fan vehicle which is being developed as a demonstrator vehicle to meet agility, ease of deployment, and other requirements for future UAV systems. Because of its vertical takeoff and landing capability as well as its relatively small size, such a vehicle offers several potential tactical advantages for company- and other lower-level reconnaissance and surveillance activities. There are also numerous envisaged applications of smaller-scale versions of this vehicle for homeland security. Future applications could conceivably deploy hundreds of OAVs flying in formation to accomplish specific objectives.

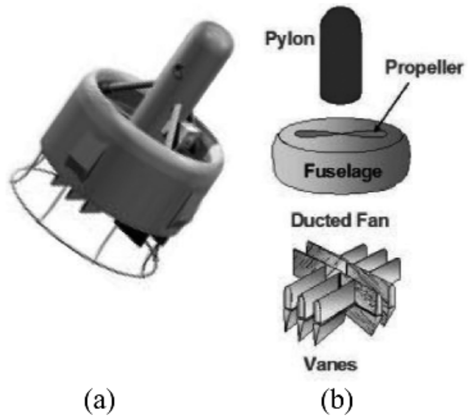


Fig. 1. OAV modeling. (a) OAV. (b) Components of the OAV.

The OAV, depicted in Fig. 1(a), exhibits a highly nonlinear, constrained multivariable character. The vehicle translates and rotates by modulating thrust and propeller wash along its axis via the deflection of specific vane sets. For instance, to translate in a forward direction, the vehicle "tilts" by an angle proportional to the desired speed of travel in the given direction. Tilt angles are similar to pitch and roll angle in the helicopter coordinate system notation. The body x-axis points forward in the duct inlet plane. This is also the direction in which a mounted camera would point. It lies along the hinge of vane sets that are used for rolling motion. The body y-axis points to the right in the duct inlet plane. This axis lies along the hinge axis of vane sets used to pitch the vehicle. The body z-axis points down along propeller axis.

The vehicle is modeled by (conceptually) breaking it up into its constituent parts and then developing physics-based models for each component. These building blocks are shown in Fig. 1(b).

The four sets of vanes include left and right *pitch* vanes and forward and aft *roll* vanes. In the dynamic model of the OAV, the inputs are the vane deflections and the thrust along the propeller axis (throttle). Aerodynamic and propulsion models that map these quantities to forces and moments on the vehicle's rigid body are developed from first principles using the geometry of the surfaces and by treating the problem as a basic aircraft model.

III. Low-Level Control Design/High-Level Modeling

Nonlinear control of the "inner loop" (i.e., attitude and rate loops) and the "outer loop" (i.e., position and velocity loops) is accomplished via nonlinear dynamic inversion [31] and robust multivariable control. The nonlinearities of the various loops are cancelled (to a certain degree) by inversion and desired dynamics are imposed on the resulting system so that the behavior resembles a set of integrators. Due to imperfect inversion, the

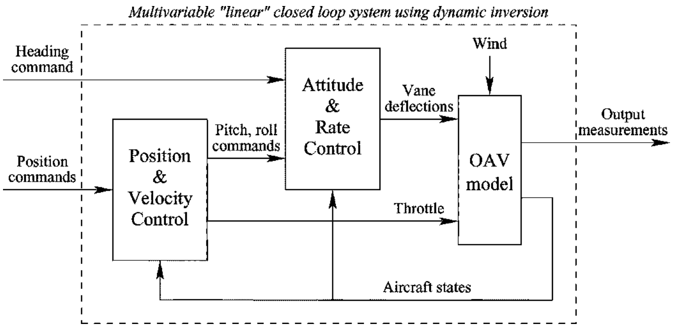


Fig. 2. Controlled OAV model.

response of these state variables tend to be more similar to a low-order transfer function rather than a pure integrator.

The overall control scheme is depicted in Fig. 2. This kind of structure has been implemented at Honeywell as a reusable set of flight control tools called multiapplication controls—Honeywell (MACH). MACH combines nonlinear design methods, classical flight control structures and design parameters, and robust multivariable design principles. Apart from the OAV, it has been used successfully on several programs including the F-18 High-Alpha Research Vehicle and is currently being applied in DARPA's microair vehicles (MAV) program.

High-level commands to the MACH-controlled system take the form of desired N, E, h-positions and heading angles (see Fig. 2). The vehicle may also be commanded via way points expressed in terms of desired positions/heading or corresponding velocities/heading rate to these coordinates. The position/velocity control system takes these position commands as inputs and generates corresponding tilt (pitch, roll) and throttle commands. The tilt and heading commands are the inputs to the attitude/rate control system. Its outputs are actual vane deflections required to accomplish the commanded maneuver. These vane deflections, as well as the throttle and wind disturbances are the inputs to the OAV model.

Assuming near-perfect dynamic inversion, the dynamics from the commanded position and heading to the outputs (which may be selected as actual positions and heading) is that of a *multivariable linear or piecewise linear system.* Although imperfect dynamic inversion may introduce slight coupling among other state variables, the heading dynamics can be considered decoupled from the N, E, h position. Therefore, they are not considered in the remainder of this paper. In OAV formation flight, vehicles at the higher level can be modeled as a third-order constrained linear system in each axis. The inputs to the system dynamics are commanded positions along the N, E, h-axes, and the outputs are positions and velocities along the N, E, h-axes.

The high-level closed-loop OAV dynamics are described using the following discrete-time model:

$$\begin{aligned} x_{k+1} &= f(x_k, u_k) \\ y_k &= h(x_k) \end{aligned} \tag{1}$$

where the state update function $f : \mathbb{R}^9 \times \mathbb{R}^3 \to \mathbb{R}^9$ and output map $h : \mathbb{R}^9 \to \mathbb{R}^6$ are linear or piecewise linear functions of its inputs. The states and inputs of the vehicle at time k are denoted by $x_k \in \mathbb{R}^9$ and $u_k \in \mathbb{R}^3$, respectively. In particular

$$u_k = \begin{bmatrix} N_{\text{cmd}} \\ E_{\text{cmd}} \\ h_{\text{cmd}} \end{bmatrix} \quad y_k = \begin{bmatrix} N \\ E \\ h \\ \dot{N} \\ \dot{E} \\ \dot{h} \end{bmatrix} = \begin{bmatrix} y_{k,\text{pos}} \\ y_{k,\text{vel}} \end{bmatrix} \quad x_k = \begin{bmatrix} x_{k,\text{pos}} \\ x_{k,\text{vel}} \end{bmatrix} \tag{2}$$

and $x_{k,\text{pos}} \in \mathbb{R}^3$ is the vector of N, E, h coordinates and $x_{k,\text{vel}} \in \mathbb{R}^6$ denotes a vector of states corresponding to N-axis, E-axis, and h-axis velocity components at time k. Speed and acceleration of the OAV are constrained as follows:

$$\begin{aligned} y_{\text{vel}} &\in \mathcal{Y}_v = \{z \in \mathbb{R}^3 \mid \underline{y_{\text{vel}}} \le z_i \le \overline{y_{\text{vel}}}, i = 1, 2, 3\} \\ \dot{y}_{\text{vel}} &\in \mathcal{Y}_a = \{z \in \mathbb{R}^3 \mid \underline{\dot{y}_{\text{vel}}} \le z_i \le \overline{\dot{y}_{\text{vel}}}, i = 1, 2, 3\}. \end{aligned} \tag{3}$$

The constrained dynamical model (1) directly accounts for practical limitations on the vehicles' velocity and acceleration. It also allows redesign and modification of the inner-loop controllers to track velocity or acceleration references directly.

OAV autonomous formation flight is rendered tractable using high-level closed-loop vehicle dynamics (1) for formation control. A group of OAVs can be controlled by commanding either desired position or speed or even acceleration in order to achieve certain mission objectives. In the simulation results presented in Section VI, the decentralized control framework accomplishes mission objectives via the position command tracking mode of the OAV guidance and control system (1)–(2). The proposed decentralized control design scheme is presented in the following section for a generic linear or piecewise linear multivariable high-level vehicle model with arbitrary independent variables u_k (such as position, speed, or acceleration commands), in order to maintain a general exposition.

IV. High-Level Control Strategy

In this section, we introduce a decentralized receding horizon control (RHC) scheme used for high-level control of the OAV based on [30]. The main concept of RHC is to use a *model* of the plant to *predict* the future evolution of the system [32]. Based on this prediction, at each time step t a certain performance index is optimized under operating constraints with respect to a sequence of future input moves. The first of such optimal moves is the *control* action applied to the plant at time t. At time $t + 1$, a new optimization is solved over a shifted prediction horizon.

The decentralized RHC scheme is formulated for dynamically decoupled systems, such as OAVs flying in formation, which are coupled only by the cost function and constraints of an optimization problem. The main idea of the proposed framework is to break a centralized RHC problem into problems of smaller sizes [29], [30]. Each RHC controller is associated with a different vehicle and computes the local control inputs based only on its states and that of its neighbors. On each vehicle, the current state and the model of its neighbors are used to predict their possible trajectories and move accordingly (similar to what we do while driving cars). The information exchange topology and inter-vehicle constraints are described by a graph structure in the problem formulation. The cost function will depend on the formation's mission and include terms that minimize relative distances and/or velocities between neighboring vehicles. The coupling constraints arise from collision avoidance. The interaction graph is full (each vehicle has to avoid all the other vehicles) but it is approximated with a time-varying graph based on a "closest spatial neighbors" model.

A more formal discussion of the scheme proposed in [30] follows with an extension to handle time-varying graphs and coupling constraints. Section V will discuss further modifications using invariant sets and emergency maneuvers, which lead to guaranteed collision avoidance.

A. Problem Formulation and Decentralized Control Scheme

Consider a set of N_v linear decoupled dynamical systems, where the ith system is described by the discrete-time time-invariant state equation

$$\begin{aligned} x_{k+1}^i &= f^i\left(x_k^i, u_k^i\right) \\ y_k^i &= h^i\left(x_k^i\right) \end{aligned} \tag{4}$$

where $x_k^i \in \mathbb{R}^{n^i}$, $u_k^i \in \mathbb{R}^{m^i}$, $f^i : \mathbb{R}^{n^i} \times \mathbb{R}^{m^i} \to \mathbb{R}^{n^i}$, $h^i : \mathbb{R}^{n^i} \to \mathbb{R}^p$ are states, inputs, state update function, and output function of the ith vehicle, respectively. Let $\mathcal{U}^i \subseteq \mathbb{R}^{m^i}$ and $\mathcal{Y}^i \subseteq \mathbb{R}^p$ denote the set of feasible inputs and outputs of the ith vehicle

$$y_k^i \in \mathcal{Y}^i \quad u_k^i \in \mathcal{U}^i \quad k \geq 0 \tag{5}$$

where $\mathcal{Y}^i$ and $\mathcal{U}^i$ are given polytopes.

The set of N_v constrained systems will be referred to as the *multivehicle system*. Let $\tilde{x}_k \in \mathbb{R}^{\tilde{n}}$ with $\tilde{n} = \sum_i n^i$ and $\tilde{u}_k \in \mathbb{R}^{\tilde{m}}$ with $\tilde{m} = \sum_i m^i$ be the vectors which collect the states and inputs of the multivehicle system at time k, i.e., $\tilde{x}_k = [x_k^1, \ldots, x_k^{N_v}]$, $\tilde{u}_k = [u_k^1, \ldots, u_k^{N_v}]$, with

$$\tilde{x}_{k+1} = f(\tilde{x}_k, \tilde{u}_k). \tag{6}$$

The equilibrium pair of the ith vehicle is denoted by (x_e^i, u_e^i), and $(\tilde{x}_e, \tilde{u}_e)$ denotes the corresponding equilibrium for the {multivehicle system}.

So far the individual vehicles belonging to the {multivehicle system} are completely decoupled. We consider an optimal control problem for the {multi-vehicle system} where cost function and constraints couple the dynamic behavior of individual systems. A graph topology is used to represent the coupling in the following way. The ith system is associated with the ith node of the graph. If an edge (i, j) connecting the ith and jth node is present, then the cost and the constraints of the optimal control problem will have a component which is a function of both x^i and x^j. The graph will be *undirected*, i.e., $(i, j) \in \mathcal{A} \Rightarrow (j, i) \in \mathcal{A}$ and furthermore, the edges representing coupling change with time. Therefore, before defining the optimal control problem, we need to define a graph (which can be time-varying)

$$\mathcal{G}(t) = (\mathcal{V}, \mathcal{A}(t)) \tag{7}$$

where $\mathcal{V}$ is the set of nodes $\mathcal{V} = \{1, \ldots, N_v\}$ and $\mathcal{A}(t) \subseteq \mathcal{V} \times \mathcal{V}$ the set of time-varying edges (i, j) with $i \in \mathcal{V},\ j \in \mathcal{V}$.

Using the graph structure defined previously, the optimization problem is formulated as follows. Denote with $\tilde{x}_k^i$ the states of all neighbors of the ith vehicle at time k, i.e., $\tilde{x}_k^i = \{x_k^j \in \mathbb{R}^{n^j} | (i, j) \in \mathcal{A}(k)\}$, $\tilde{x}_k^i \in \mathbb{R}^{\tilde{n}_k^i}$ with $\tilde{n}_k^i = \sum_{j|(i,j)\in\mathcal{A}(k)} n^j$. Analogously, $\tilde{u}_k^i \in \mathbb{R}^{\tilde{m}_k^i}$ and $\tilde{y}_k^i \in \mathbb{R}^{\tilde{p}_k^i}$ denote the inputs and outputs of all the neighbors of the ith vehicle at time k, respectively. Let

$$g^{i,j}(x^i, x^j) \leq 0 \tag{8}$$

define interconnection constraints between the ith and the jth vehicle, with $g^{i,j} : \mathbb{R}^{n^i} \times \mathbb{R}^{n^j} \to \mathbb{R}^{nc^{i,j}}$. We will often use the following shorter notation for the interconnection constraints defined between the ith vehicle and all its neighbors:

$$g_k^i\left(x_k^i, \tilde{x}_k^i\right) \leq 0 \tag{9}$$

with $g_k^i : \mathbb{R}^{n^i} \times \mathbb{R}^{\tilde{n}_k^i} \to \mathbb{R}^{nc_k^i}$, where $nc_k^i = \sum_{j|(i,j)\in\mathcal{A}(k)} nc^{i,j}$. For formation flight, these constraints define nonconvex collision avoidance requirements in the following way:

$$g^{i,j}\left(x_k^i, u_k^i, x_k^j, u_k^j\right) = d_{\text{safe}} - \left\| y_{k,\text{pos}}^i - y_{k,\text{pos}}^j \right\|_p. \tag{10}$$

The parameter d_{safe} in constraint (10) represents a lower bound on the norm of relative position between neighboring vehicles. In this paper, we will consider $p = 1, \infty$ which lead to square protection zones around vehicles. In general, collision avoidance guarantees between any OAV pair in a formation flight problem would necessitate the use of a full graph for describing inter-vehicle constraints (OAV protection zones should not intersect for every vehicle pair). This would prevent the use of any approach but a centralized one. For practical decentralization purposes, it is usually sufficient for each vehicle to consider only a neighboring subset of all vehicles to accomplish formation flight. Due to possible changes of the required formation, this subset is likely to change leading to a time-varying interconnection graph. Furthermore, the allowed number of vehicles in these neighboring subsets might be limited, leading to a directed interconnection structure and graph in (7). The proposed scheme can be naturally extended to cover these situations as well.

Consider the following overall cost function:

$$l(\tilde{x}, \tilde{u}) = \sum_{i=1}^{N_v} l_k^i\left(x^i, u^i, \tilde{x}_k^i, \tilde{u}_k^i\right) \tag{11}$$

where $l^i : \mathbb{R}^{n^i} \times \mathbb{R}^{m^i} \times \mathbb{R}^{\tilde{n}_k^i} \times \mathbb{R}^{\tilde{m}_k^i} \to \mathbb{R}$ is the cost associated with the ith vehicle and is a function of its states and the states of its neighbor nodes. Assume that l is a positive convex function with $l(\tilde{x}_e, \tilde{u}_e) = 0$. The cost function l_k^i in (11) is assumed to have the following form:

$$l_k^i\left(x^i, u^i, \tilde{x}_k^i, \tilde{u}_k^i\right) = \left\| Q \begin{bmatrix} y^i - y_f^i \\ \tilde{y}_k^i - \tilde{y}_f^i \\ \Delta y_k^i - \Delta y_f^i \\ \Delta \tilde{y}_k^i - \Delta \tilde{y}_f^i \end{bmatrix} \right\|_p + \left\| R \begin{bmatrix} u^i \\ \tilde{u}_k^i \end{bmatrix} \right\|_p \tag{12}$$

where Δy_k^i is a stacked vector of relative outputs $(y^i - y^j)$ for all j such that $(i, j) \in \mathcal{A}(k)$. It represents the difference between the ith vehicle outputs and its neighbors' outputs. The variable $\Delta\tilde{y}$ denotes a similar collection of relative outputs between the neighbors of the ith vehicle, i.e., it is comprised of $(y^q - y^r)$ output differences for all q, r such that $(i,q) \in \mathcal{A}(k)$ and $(i,r) \in \mathcal{A}(k)$. Subscript f denotes steady-state final reference values for all variables. The previous general cost function involves weights on the outputs and control inputs of the ith OAV and its neighbors, as well as weighting the relative outputs between them.

Centralized solutions to the OAV formation control problem are characterized by the presence of a large number of states, and can be solved using techniques such as MILP path planning described, for instance, in [12]. However, for large-scale systems the computational requirements grow combinatorially with problem size, making such an approach intractable and impractical. We tackle the complexity associated with the design of controllers for such class of large-scale systems by using decentralized optimal control schemes.

We formulate a decentralized receding horizon control (DRHC) scheme based on the cost function, system models, and constraints previously defined. Given a certain graph interconnection structure $\mathcal{G}(t)$, let the following finite-time optimal control problem $\mathcal{P}_i(t)$ be associated with the ith vehicle at time t:

$$\min_{\tilde{U}_t^i} \sum_{k=0}^{N-1} l_t^i\left(x_{k,t}^i, u_{k,t}^i, \tilde{x}_{k,t}^i, \tilde{u}_{k,t}^i\right) + l_N^i\left(x_{N,t}^i, \tilde{x}_{N,t}^i\right) \quad (13a)$$

$$\text{subj. to}\; x_{k+1,t}^i = f^i\left(x_{k,t}^i, u_{k,t}^i\right) \quad (13b)$$

$$y_{k,t}^i = h^i\left(x_{k,t}^i\right),$$

$$y_{k,t}^i \in \mathcal{Y}^i, \quad u_{k,t}^i \in \mathcal{U}^i \quad (13c)$$

$$k = 1, \dots, N-1$$

$$x_{k+1,t}^j = f^j\left(x_{k,t}^j, u_{k,t}^j\right), \quad (i,j) \in \mathcal{A}(t) \quad (13d)$$

$$y_{k,t}^j = h^j\left(x_{k,t}^j\right)$$

$$y_{k,t}^j \in \mathcal{Y}^i, \quad u_{k,t}^j \in \mathcal{U}^j, \quad (i,j) \in \mathcal{A}(t) \quad (13e)$$

$$k = 1, \dots, N-1$$

$$g^{i,j}\left(x_{k,t}^i, u_{k,t}^i, x_{k,t}^j, u_{k,t}^j\right) \le 0, \quad (i,j) \in \mathcal{A}(t) \quad (13f)$$

$$k = 1, \dots, N-1$$

$$g^{q,r}\left(x_{k,t}^q, u_{k,t}^q, x_{k,t}^r, u_{k,t}^r\right) \le 0 \quad (13g)$$

$$(i,q) \in \mathcal{A}(t), \quad (i,r) \in \mathcal{A}(t)$$

$$k = 1, \dots, N-1$$

$$x_{N,t}^i \in \mathcal{X}_f^i, \quad x_{N,t}^j \in \mathcal{X}_f^j, \; (i,j) \in \mathcal{A}(t) \quad (13h)$$

$$x_{0,t}^i = x_t^i, \quad \tilde{x}_{0,t}^i = \tilde{x}_t^i \quad (13i)$$

where $\tilde{U}_t^i \triangleq [u_{0,t}^i, \tilde{u}_{0,t}^i, \dots, u_{N-1,t}^i, \tilde{u}_{N-1,t}^i] \in \mathbb{R}^s$, $s \triangleq (\tilde{m}^i + m^i)N$ denotes the optimization vector, $x_{k,t}^i$ denotes the state vector of the ith vehicle predicted at time $t+k$ obtained by starting from the state x_t^i and applying to system (4) the input sequence $u_{0,t}^i, \dots, u_{k-1,t}^i$. The tilded vectors denote the prediction vectors associated with the neighboring systems *assuming a constant interconnection graph.* Denote by $\tilde{U}_t^{i*} = [u_{0,t}^{*i}, \tilde{u}_{0,t}^{*i}, \dots, u_{N-1,t}^{*i}, \tilde{u}_{N-1,t}^{*i}]$ the optimizer of problem $\mathcal{P}_i(t)$. Note that problem $\mathcal{P}_i(t)$ involves only the state and input variables of the ith vehicle and its neighbors at time t.

We will define the following decentralized RHC scheme. At time t

1) Compute graph connection $\mathcal{A}(t)$ according to a chosen policy.
2) Each vehicle i solves problem $\mathcal{P}_i(t)$ based on measurements of its state x_t^i and the states of all its neighbors $\tilde{x}_t^i$.
3) Each vehicle i implements the first sample of $\tilde{U}_t^{i*}$

$$u_t^i = u_{0,t}^{*i}. \quad (14)$$

4) Each vehicle repeats steps 1)–4) at time $t+1$, based on the new state information x_{t+1}^i, $\tilde{x}_{t+1}^i$.

In order to solve problem $\mathcal{P}_i(t)$ each vehicle needs to know its current states, its neighbors' current states, its terminal region, its neighbors' terminal regions, and models and constraints of its neighbors. Based on such information each vehicle computes its optimal inputs and its neighbors' optimal inputs assuming a constant set of neighbors over the horizon. The input to the neighbors will only be used to predict their trajectories and then discarded, while the first component of the ith optimal input of problem $\mathcal{P}_i(t)$ will be implemented on the ith vehicle. The solution of the ith subproblem will yield a control policy for the ith vehicle of the form $u_t^i = c_t^i(x_t^i, \tilde{x}_t^i)$.

A detailed discussion on feasibility and stability issues of decentralized RHC schemes goes beyond the scope of this paper. Some important observations can be found in [29], [30], and [33]–[37]. The main research topics include: 1) the choice of the graph topology when it is not fixed or unique and 2) the choice of local prediction horizons, terminal weights, and terminal regions and its effect on the global performance and feasibility. Addressing these issues typically involves modifying the RHC scheme to some extent either by making more restrictive assumptions (such as using special graph structure in [36] and [38]) or introducing additional degrees-of-freedom (such as exchange of optimizers between neighbors in [34]). However, even in its basic formulation the proposed framework provides the following several advantages:

- can handle constrained MIMO linear models as well as constrained MIMO piecewise linear models of OAVs;
- different maneuvering objectives can be achieved by changing appropriate terms in the cost function (e.g., formation keeping, formation joining, and formation flying);
- individual vehicles use neighbor information to predict their behavior in order to avoid collisions and act in a cooperative, rather than worst-case manner (similarly to what we do while driving cars);
- the problem is formulated and solved as small MILPs which can be translated into equivalent gain scheduled controllers for real-time implementation.

A schematic diagram of the hierarchical decomposition approach to OAV formation control is shown in Fig. 3. Each OAV is controlled by its lower-level guidance and control loops, which receive commands from the on-board decentralized RHC algorithm operating in closed-loop based on local measurements. In addition to local measurements, other information (such as state estimates) can be exchanged with neighbors. Exchange of other types of data is also allowed (e.g., predicted optimal solution sequence) if sufficient communication bandwidth is available. The mission manager is located on a ground station, which instructs

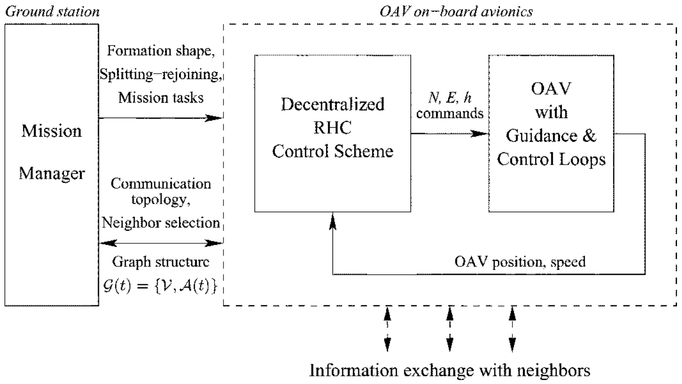

Ground station
OAV on-board avionics
Mission Manager
Formation shape, Splitting–rejoining, Mission tasks
Communication topology, Neighbor selection
Graph structure $\mathcal{G}(t) = \{\mathcal{V}, \mathcal{A}(t)\}$
Decentralized RHC Control Scheme
N, E, h commands
OAV with Guidance & Control Loops
OAV position, speed
Information exchange with neighbors

Fig. 3. Decentralized scheme for OAV formation control.

the formation to perform certain mission-specific maneuvers by simply modifying their position and velocity reference values. The maneuver will be performed autonomously without interaction with the mission manager. The ground station is also capable of assigning an interconnection graph structure for the group of OAVs. Alternatively, the interconnection structure can be determined "on-the-fly" by the OAVs in a decentralized fashion based on, for instance, a closest spatial neighbor policy. There might be multiple on-board RHC controllers, which are generated using different cost functions and activated only in the corresponding phases of a mission. In Section V, we describe how collision avoidance can be ensured within the decentralized framework described so far.

V. Guaranteed Collision Avoidance

As highlighted in Section IV-A, decentralized RHC formulations by themselves do not guarantee collision avoidance for two reasons. First, the interconnection graph $\mathcal{G}$ cannot be chosen full otherwise decentralization does not lead to reduced problem complexity. Second, the local RHC problems are not guaranteed to be always feasible due to the mismatch between predicted and actual neighbor trajectories.

In this paper, we make use of emergency controllers and their invariant sets to define protection zones that always guarantee collision avoidance when the local RHC subproblems become infeasible. The following sections provide a detailed description of this methodology. Section V-A reports important results from set invariance theory that we will rely on to define and calculate protection zones based on invariant sets associated with emergency controllers. The choice of emergency controllers is described in detail in Section V-B. Section V-C concludes by formulating a modified version of the decentralized RHC scheme that incorporates additional constraints based on invariant sets and switching to emergency controllers if infeasibility is encountered.

A. Set Invariance Definitions

This section provides a brief summary of invariant sets for constrained systems. The main objective is to describe some background concepts which are used in the next sections. Our description is based entirely on [39].

In the following, we focus on autonomous LTI systems without disturbances, assuming that a certain controller for the emergency maneuver has been chosen. Theoretical results and computational tools are readily available for nonautonomous and robust versions of the following definitions and theorems. A numerical approach for the case of nonlinear systems is discussed in [40]. A comprehensive survey of papers on set invariance can be found in [41].

Assume the following discrete-time dynamical system is given:

$$x_{k+1} = f(x_k, u_k) \tag{15}$$

where x_k is the system state and u_k is the control input. The system is subject to constraints on the control inputs and the states

$$u_k \in \mathcal{U} \subset \mathbb{R}^m \tag{16a}$$

$$x_k \in \mathcal{X} \subseteq \mathbb{R}^n. \tag{16b}$$

The set $\mathcal{U}$ is compact, while $\mathcal{X}$ is closed. It is assumed that the system and the constraints are time invariant.

An admissible control input, sequence, or law is one which satisfies the input constraints $\mathcal{U}$. It is of interest to determine which subset of a given set is compatible with the input and output constraints. This segment defines the concept of the input admissible sets.

If the system is in closed-loop with the control law

$$u_k = c(x_k)$$

then the input admissible set is the subset of a given Ω in which the control law satisfies the input constraints.

Definition 1 (Input Admissible Set): Given a control law $u_k = c(x_k)$, the *input admissible* subset of $\Omega \subseteq \mathbb{R}^n$ is given by

$$\Omega^c \triangleq \{x_k \in \Omega \,|\, c(x_k) \in \mathcal{U}\}. \tag{17}$$

The closed-loop system is given by

$$x_{k+1} = f(x_k, c(x_k))$$

and the constraints on the state can be replaced by

$$x_k \in \mathcal{X}^c \triangleq \{x_k \in \mathcal{X} \,|\, c(x_k) \in \mathcal{U}\}.$$

Note that if the constraints on the input $\mathcal{U}$ are given as a hyperrectangle and the control law is given by an appropriate saturation function (as is often the case), then $\Omega^c = \Omega$ (assuming that the control law is defined over Ω). If the system is LTI and $u_k = \mathrm{sat}(Kx_k)$, then the resulting closed-loop system can be treated as a piecewise affine system. Statements about systems without control inputs will also apply to closed-loop systems, bearing in mind that the state constraints should be replaced by the input admissible subset, where necessary.

Given a set Ω in the state-space and an initial state $x_0 \in \Omega$, it is of interest to determine whether the evolution of the system will remain inside the set for all time.

Definition 2 (Positively Invariant Set): The set $\Omega \subset \mathbb{R}^n$ is *positively invariant* for the system $x_{k+1} = f(x_k)$ if and only if $\forall x_0 \in \Omega$ the system evolution satisfies $x_k \in \Omega, \forall k \geq 0$.

In other words, Ω is positively invariant if and only if

$$x_k \in \Omega \Rightarrow x_{k+1} \in \Omega. \tag{18}$$

The union of two positively invariant sets is positively invariant, however, the same statement cannot be made about the intersection of two positively invariant sets. Numerical computation of invariant sets makes use of the Pontryagin Difference and the Minkowski sum, details can be found in [39].

Section V-B makes use of these set theoretic algorithms in the calculation of the invariant set associated with an OAV emergency controller, which performs a specific maneuver.

B. OAV Emergency Maneuver and Invariant Set

States and inputs of the OAV are constrained based on the physics of the problem. Two types of constraints are considered: a set of "emergency" constraints and a set of "nominal" constraints. The "nominal" constraints define the operating region of the vehicle under normal operation. Nominal constraints are more restrictive than the actual operating limits of the OAV. Maximum performance is used only in emergency situations and is defined by the following set of "emergency" constraints:

- nominal constraints

$$\begin{aligned} x_{\text{vel}} &\in \mathcal{X}_v \\ u &\in \mathcal{U}; \end{aligned} \tag{19}$$

- emergency constraints as described by the vehicle's limitations in (3)

$$\begin{aligned} x_{\text{vel}} &\in \mathcal{X}_v^{\text{ER}} \\ u &\in \mathcal{U}^{\text{ER}}. \end{aligned} \tag{20}$$

Consider a single OAV (1) and a state-feedback emergency controller

$$u_k = c(x_k, r_e) \tag{21}$$

which controls the vehicle to a chosen reference r_e under the constraints (20). Denote by t_e the time instant when an emergency maneuver starts. The closed-loop OAV dynamics during emergency maneuver are

$$x_{k+1} = f(x_k, c(x_k, r_e)), \quad \text{for } k \ge t_e. \tag{22}$$

The emergency controller $c(x_k, r_e)$ can be designed to achieve different objectives depending on the type of UAV and on its mission. For instance the emergency maneuver could consist of bringing the vehicle to a full stop. For winged UAVs it could perform a continuous flight in a circle of a given radius. A similar concept is used in [42] in the form of basis states as hard terminal constraints, which provide implicit safety guarantees to ensure feasibility of a receding horizon path planning scheme. Although more applicable to a hovering vehicle such as the OAV, without loss of generality, we assume in this paper that an emergency maneuver started at t_e brings the vehicle to a full stop (zero terminal speed) at the position which it had at time t_e, i.e., $r_e = [x_{t_e,\text{pos}}, x_{t_e,\text{vel}}]$, where states $x_{t_e,\text{pos}}$ correspond to the reference position output values and the states $x_{t_e,\text{vel}}$ correspond to zero velocities at that position.

We define a protection zone $\mathcal{Y}_p^{\text{ER}} \subset \mathbb{R}^3$ centered at $y_{t_e,\text{pos}}$ to be a polytope in the x, y, z space containing the OAV position during emergency maneuvers. In order to guarantee this property, we compute offline the set $\Xi(t_e) \in \mathbb{R}^9$ of vehicle states for any time t_e such that position outputs of the closed-loop dynamics (22) for $k \ge t_e$ and $x_{t_e} \in \Xi(t_e)$ lie in the emergency region $\mathcal{Y}_p^{\text{ER}}$. $\Xi(t_e)$ is a positively invariant set of system (22) subject to constraints on input commands and velocity states (20) and on position defined by $\mathcal{Y}_p^{\text{ER}}$

$$x \in \Xi(t_e) \iff \begin{array}{l} x_{\text{vel}} \in \mathcal{X}_v^{\text{ER}}, c(x) \in \mathcal{U}^{\text{ER}}, y_{\text{pos}} \in \mathcal{Y}_p^{\text{ER}}(t_e) \\ h(f(x, c(x))) \in \Xi(t_e) \quad \forall t \ge t_e. \end{array} \tag{23}$$

If the emergency maneuver (21) is started when all the states are in $\Xi(t_e)$, then the OAV is guaranteed to satisfy emergency constraints on input commands and velocities and to stay within the protection zone $\mathcal{Y}_p^{\text{ER}}$.

If $c(x)$ is a linear state-feedback controller, then $\Xi(0)$ can be easily computed with simple techniques using polyhedral manipulations as described in [39]. The set $\Xi(k)$ is just a translation of the set $\Xi(0)$ to the position $y_{k,\text{pos}}$.

In conclusion, once Ξ has been computed, in order to guarantee that the vehicle performs the emergency maneuver within the emergency region, the nominal constraints (19) have to be augmented with the constraint

$$x_k \in \Xi(k) \tag{24}$$

to ensure that these maneuvers always start from within Ξ.

In this section, a double integrator vehicle model is used along each spatial dimension for illustration of the previous ideas. The states are position and velocity and the control input is acceleration. The same results will be presented for the high-level linear OAV model as well in Section VI.

A cross section of the maximal positively invariant set for the double integrator model is shown in Fig. 4 using the following protection zone, velocity, and acceleration limits:

$$\begin{aligned} y_{\text{pos}} &\in \mathcal{Y}_p^{\text{ER}}(t_e) = \left\{z \in \mathbb{R}^3 | ||y_{t_e,\text{pos}} - z||_\infty \le 5 \text{ ft}\right\} \\ x_{\text{vel}} &\in \mathcal{X}_v^{\text{ER}} = \{z \in \mathbb{R}^3 | ||z||_\infty \le 10 \text{ ft/s}\} \\ u &\in \mathcal{U}^{\text{ER}} = \{z \in \mathbb{R}^3 | ||z||_\infty \le 3 \text{ ft/s}^2\} \end{aligned} \tag{25}$$

where an LQR regulator was used as the emergency controller. The trajectories of the OAV performing an emergency stop will lie in the set $\Xi(t_e)$ if at time t_e the state of the OAV x_{t_e} belongs to the set $\Xi(t_e)$. Since $\Xi(t_e)$ is centered at $y_{t_e,\text{pos}}$, constraint (24) becomes $x_{t_e,\text{vel}} \in \Xi_v, \Xi_v = \{x_{\text{vel}} \in \mathbb{R}^3 | (\vec{0}, x_{\text{vel}}) \in \Xi\}$. The set Ξ_v will constrain the speed of the OAV to lie within bounds from which an emergency stop can be accomplished without violating $\mathcal{Y}_p^{\text{ER}}$. Ξ is a polyhedron and, therefore, Ξ_v will be a polyhedron as well. The size of Ξ_v is a function of $\mathcal{Y}_p^{\text{ER}}$, $\mathcal{X}_v^{\text{ER}}$ and $\mathcal{U}^{\text{ER}}$. By analyzing the results in this simple example, we can observe that the bigger $\mathcal{U}^{\text{ER}}$ is, the faster the OAV can stop, which leads to a bigger set Ξ_v from which an emergency maneuver can start. The smaller $\mathcal{Y}_p^{\text{ER}}$ is, the smaller the set of initial velocities becomes from which the vehicle can stop in $\Xi(t_e)$. In other words, we have formalized using mathematical tools, the tradeoff between the nominal vehicle speed limits and the extent to which vehicles can accelerate/decelerate.

Once the invariant set $\Xi(k)$ has been computed, decentralized RHC controllers are designed for formation flying which: 1) enforce the additional constraint (24) based on emergency maneuver invariant set; 2) use protection zones larger than $\Xi(k)$; and 3) switch to an emergency maneuver when the constrained optimization problem becomes infeasible. These modifications are incorporated in the design of decentralized RHC controllers as shown in Section V-C.

C. *Modified Decentralized Control Scheme*

The speed and acceleration of each OAV is constrained according to (19) and (24). Every OAV can perform the same emergency maneuver and to each of them we assign a binary logic state $x^i_{k,L} = 1$ if the ith vehicle is performing an emergency maneuver at time k, and $x^i_{k,L} = 0$ if the vehicle is operating under nominal conditions. Based on this state augmentation and the emergency invariant set described in Section V-B, consider the following modification to the basic decentralized problem (13). The modified finite-time optimal control problem $\mathcal{P}^{\text{inv}}_i(t)$ associated with the ith system at time t has the form

$$\min_{\tilde{U}^i_t} \sum_{k=0}^{N-1} l^i_t\left(x^i_{k,t}, u^i_{k,t}, \tilde{x}^i_{k,t}, \tilde{u}^i_{k,t}\right) + l^i_N\left(x^i_{N,t}, \tilde{x}^i_{N,t}\right) \tag{26a}$$

$$\text{subj. to } x^i_{k+1,t} = f^i\left(x^i_{k,t}, u^i_{k,t}\right) \tag{26b}$$

$$y^i_{k,t} = h^i(x^i_{k,t})$$

$$x^i_{k,t,\text{vel}} \in \mathcal{X}_v, \quad u^i_{k,t} \in \mathcal{U} \tag{26c}$$

$$\text{if } x^j_{t,L} = 0 \begin{cases} x^j_{k+1,t} = f^j\left(x^j_{k,t}, u^j_{k,t}\right) \\ y^j_{k,t} = h^j\left(x^j_{k,t}\right) \\ x^j_{k,t,\text{vel}} \in \mathcal{X}_v, \quad u^j_{k,t} \in \mathcal{U} \end{cases} \tag{26d}$$

$$\text{if } x^j_{t,L} = 1 \begin{cases} x^j_{k+1,t} = f^j\left(x^j_{k,t}, c\left(x^j_{k,t}\right)\right) \\ y^j_{k,t} = h^j\left(x^j_{k,t}\right) \end{cases} \tag{26e}$$

$$g^{i,j}\left(x^i_{k,t,\text{pos}}, x^j_{k,t,\text{pos}}\right) \le 0, \quad (i,j) \in \mathcal{A}(t) \tag{26f}$$

$$g^{q,r}\left(x^q_{k,t,\text{pos}}, x^r_{k,t,\text{pos}}\right) \le 0 \tag{26g}$$

$$(i,q) \in \mathcal{A}(t), \quad (i,r) \in \mathcal{A}(t),$$

$$x^i_{k,t,\text{vel}} \in \Xi_v, \tag{26h}$$

$$x^j_{k,t,\text{vel}} \in \Xi_v, \quad (i,j) \in \mathcal{A}(t) \tag{26i}$$

$$x^i_{N,t} \in \mathcal{X}^i_f, \quad x^j_{N,t} \in \mathcal{X}^j_f, \quad (i,j) \in \mathcal{A}(t),$$

$$x^i_{0,t} = x^i_t, \quad \tilde{x}^i_{0,t} = \tilde{x}^i_t.$$

Using problem $\mathcal{P}^{\text{inv}}_i(t)$, we modify the basic decentralized RHC control scheme (14) in the following way. At time t (assuming logic states $x^i_{t,L}$ are initialized to zero at time 0):

1) compute graph connection $\mathcal{A}(t)$ based on the relative distance between the vehicles

$$\mathcal{A}(t) = \left\{(i,j) \in \mathcal{V} \times \mathcal{V} \;\middle|\; \left\| y^i_{t,\text{pos}} - y^j_{t,\text{pos}} \right\| \le d_{\min} \right\} \tag{27}$$

that is the set of all the edges, which connect two vehicles whose distance is less than or equal to $d_{\min}$;

2) ith OAV solves problem $\mathcal{P}^{\text{inv}}_i(t)$;

3) if $\mathcal{P}^{\text{inv}}_i(t)$ is feasible then $x^i_{t,L} = 0$ and the ith OAV implements the first sample of $\tilde{U}^{i*}_t$

$$u^i_t = u^{*i}_{0,t} \tag{28}$$

4) else $x^i_{t,L} = 1$ is set, and the ith OAV implements the emergency controller

$$u^i_t = c\left(x^i_t\right) \tag{29}$$

5) each OAV repeats steps 1)–4) at time $t+1$, based on the new state information x^i_{t+1}, $\tilde{x}^i_{t+1}$.

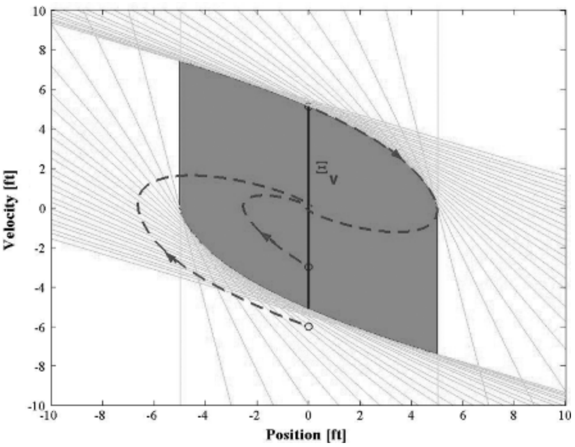


Fig. 4. Cross section of the invariant set $\Xi(t_c)$ for a double integrator vehicle model in one spatial dimension, showing the resulting Ξ_v constraint set, which has to be enforced on nominal speed in order to remain inside the invariant region using the emergency controller.

With the use of the emergency logic state information $x^i_{t,L}$, each vehicle takes into account a possible emergency maneuver of its neighbors for more accurate predictions. The inclusion of step 4) and constraints (26h)–(26i) guarantee collision-free formation flight.

The main idea of the scheme is to use an emergency stop maneuver if a vehicle's local RHC problem is infeasible. Constraint (26h) on each vehicle ensures that the emergency maneuver can be performed as discussed in Section V–B. If the polyhedron $\mathcal{Y}^{\text{ER}}_p$ is contained within the protection zones then the vehicles are guaranteed not to collide.

The previous scheme can lead to three different behaviors: 1) the vehicles fly or hover in formation (with occasional use of the emergency controllers to recover feasibility); 2) all the vehicles are in an emergency status, i.e., $x^i_{t,L} = 1$, $\forall i \in \mathcal{V}$; or 3) a subset of vehicles are in emergency status and the rest are under nominal operation, yet kept from achieving their desired goals due to the coupling in cost between the emergency and nominal vehicles' relative positions. The neighboring nominal vehicles also remain stationary in an equilibrium where the cost decrement of moving towards the target point is balanced by the cost increment that would result from leaving the "formation" of stalled vehicles. This results in a dead-lock, where not all vehicles are in emergency mode.

In the first case, vehicles could occasionally spend a finite amount of time performing an emergency maneuver before their RHC problem recovers from infeasibility. The second and third cases, however, could end up with a dead-lock situation where all or some subset of OAVs are in an emergency status. In this case, a centralized scheme would be needed in order to let the OAVs recover from the dead-lock condition. A feasible solution to such a centralized scheme with sufficiently long horizon is always guaranteed to exist (e.g., the vehicles could fly one after the other starting from the outside).

D. Practical Considerations

The control scheme presented in Section V-C could lead to frequent switches to emergency controllers and in some instances it may also be conservative. In this section, we mention practical ways of addressing these two issues.

In order to reduce the frequent occurrence of emergency maneuvers, we can modify problem $\mathcal{P}_i^{\text{inv}}(t)$ in two ways. It is possible to make use of slack variables to avoid optimal maneuvers that lead to touching protection zones. The other approach can be used to establish inter-vehicle coordination rules by means of binary decision variables in the cost function or in the constraints of the local decentralized controllers.

Simulations with the decentralized scheme applied to formation flight showed that constraints can often become active during maneuvers. This implies that a small error between the predicted trajectories of neighbors and their real trajectories can lead to infeasibility of the decentralized scheme. Optimal maneuvers can be moved away from the boundary of protection zones by modifying constraints (26f)–(26g) as

$$g^{i,j}\left(x^i_{k,t,\text{pos}}, x^j_{k,t,\text{pos}}\right) \leq d_{\text{safe}}(\varepsilon^{k,i,j} - 1) \tag{30a}$$

$$0 \leq \varepsilon^{k,i,j} \leq 1 \tag{30b}$$

and weighting in the cost the slack variables $\varepsilon^{k,i,j}$ using the extra term

$$\sum_{j|(i,j)\in\mathcal{A}} \rho\varepsilon^{k,i,j}. \tag{31}$$

The parameters $\rho > 0$ and $d_{\text{safe}} > 0$ are user-defined. The higher d_{safe} is, the less compact the formation will be.

In order to improve coordination and the likelihood of feasibility of the decentralized scheme, different "right-of-way" priorities can be introduced which allows to have better, in some sense coordinated predictions about neighbors' trajectories. This can be easily achieved if protection zones are modeled as parallelepipeds and the disjunctions are modeled as binary variables [43]. "Right-of-way" priorities can be translated into weights and constraints on the binary variables which describe the location of a vehicle with respect to a parallelepipedal protection zone of another vehicle (six binary variables in three dimensions for each vehicle couple). These ideas are explored in more detail in [44] and [45] by employing so-called coordinating functions.

Note that these practical techniques will not imply feasibility by themselves but reduce the frequency of emergency maneuvers avoiding undesirable formation behavior.

There are a few important practical observations that are due regarding the proposed emergency maneuver-based collision avoidance. Notice that if the protection zone of each vehicle is chosen to be equal to the invariant set $\mathcal{Y}_p^{\text{ER}}$ described in Section V-B, the emergency maneuver guarantees only collision avoidance, not protection zone-sized separation of vehicles at all times. This means that protection zones should be chosen larger than the invariant set calculated in Section V-B if a certain minimum separation is required.

We also point out that due to the discrete nature of the problem formulation and controller implementation, at a certain time instant vehicles can become infeasible, when the protection zones have already been violated. This situation can easily occur due to disturbances, model mismatch, or incorrect predictions about neighbors. The emergency controller should still guarantee collision avoidance in this case, which can be achieved again, by enlarging the protection zones to account for the one-time-step worst-case behavior of neighboring vehicles. In other words, the emergency invariant sets should be contained in a set that is obtained by shrinking the protection zone with a one-step worst-case maneuver of the neighbor.

The conservativeness of the presented scheme is a function of several parameters. The key parameters affecting the size of Ξ_v are $\mathcal{Y}_p^{\text{ER}}$, $\mathcal{X}_v^{\text{ER}}$, and $\mathcal{U}^{\text{ER}}$. These all influence the selection of the emergency region and the protection zones. If $\mathcal{Y}_p^{\text{ER}}$ is very small, then depending on the acceleration limits, Ξ_v might become small, which implies that the OAVs could only fly at very low speeds. On the other hand, if $\mathcal{Y}_p^{\text{ER}}$ is big, then the vehicles are required to fly very far from each other. As a result the system might perform far from its optimal point. It is also conceivable to use an hybrid strategy: once the desired formation has been reached, we can relax the nominal constraints on speed and use maximum vehicle performance, whereas in case of reconfiguration the constraints $x_{\text{vel}} \in \Xi_v$ are reinserted. This will allow to fly at low speeds only during decentralized maneuvers.

The computation of the decentralized controllers presented in this paper makes use of algorithms that rely on the most advanced results in the field of computational geometry, mathematical programming solvers, constrained optimal control, invariant set computation, and hybrid systems. In particular, we use results from hybrid systems and optimal control theory which allow us to formulate constrained optimal control problems and compute their equivalent lookup tables, which are easily implementable in real-time on the OAV hardware. Recently, a new framework for modeling constrained switched systems and an algorithm to synthesize piecewise affine (PWA) optimal controllers for such systems has been proposed [46]. Based on such framework, the design of the decentralized controllers will be performed in two steps. First, the decentralized RHC controllers based on linear or piecewise linear OAV model are tuned in simulation to achieve desired performance. The RHC controllers are not directly implementable, as it would require the mixed-integer linear programs to be solved online on each OAV. Therefore, for implementation, in the second phase the explicit PWA form of the RHC law is computed offline by using the multiparametric mixed integer programming solver presented in [46]. The use of equivalent PWA form of the RHC law will have several advantages. It is straightforward to implement on an OAV platform as a simple lookup table of gain-scheduled controllers. It can also be easily verified (an online optimization solver is impossible to verify). Its worst-case computational time can also be computed immediately.

VI. High-Fidelity Simulation Results

This section presents details of the decentralized control scheme (26)–(28) described in Section IV-A applied to the formation flight of OAVs flying at a certain altitude. The RHC control laws are implemented on the actual guidance and control loops and the high-fidelity nonlinear OAV dynamics based on wind-tunnel data. We will use two scenarios to illustrate the presented methodology. First, a formation of six vehicles interconnected along a line is required to perform a stationary formation change for subsequent surveillance purposes without colliding with each other. In the second example, a formation of six vehicles is required to move along the x-axis with a certain velocity while avoiding popup obstacles and collisions. After performing obstacle avoidance, the formation is split into two separate parts heading into different directions. Throughout this second example, the interconnection will be described by a directed graph, which is determined by the two closest visible neighbors to each vehicle.

Within the decentralized RHC framework, each vehicle is modeled as a low-order linear system in two dimensions that represent the highly nonlinear OAV dynamics under low-level control, flying at a certain altitude. The high-level linear models have constraints on states and inputs. The control inputs calculated using the decentralized RHC strategy are implemented on the actual nonlinear vehicle models and guidance/control loops in the simulations. The coupling between vehicles stems only from the common objective of the team (moving in formation) and its constraints (vehicles are not allowed to violate each others protection zones).

The simplified OAV dynamics is obtained by using linear identification techniques based on position step responses of the closed-loop controlled nonlinear OAV model. A second-order position command to velocity output discrete-time SISO LTI model with 0.6 second time delay has been identified based on the nonlinear model response. Capturing the velocity response of the nonlinear model is important for the formulation of characteristic physical vehicle constraints. The velocity output of the identified model is augmented with an integrator to obtain position as an additional output. The position response of the OAV can be considered decoupled and identical in both North and East directions. The identified high-level dynamics (4) of a single OAV is described by the following discrete-time LTI model with sampling time of 0.2 s:

$$\begin{aligned} x^i_{k+1} &= Ax^i_k + Bu^i_k \\ y^i_k &= Cx^i_k \end{aligned} \tag{32}$$

where

$$x^i_k = \begin{bmatrix} x^i_{k,\mathrm{pos}} \\ x^i_{k,\mathrm{vel}} \end{bmatrix} \quad u^i_k = \begin{bmatrix} \mathrm{x}_{\mathrm{cmd}} \\ \mathrm{y}_{\mathrm{cmd}} \end{bmatrix} \quad y^i_k = \begin{bmatrix} \mathrm{x} \\ \mathrm{y} \\ \frac{d\mathrm{x}}{dt} \\ \frac{d\mathrm{y}}{dt} \end{bmatrix} = \begin{bmatrix} y^i_{k,\mathrm{pos}} \\ y^i_{k,\mathrm{vel}} \end{bmatrix}$$

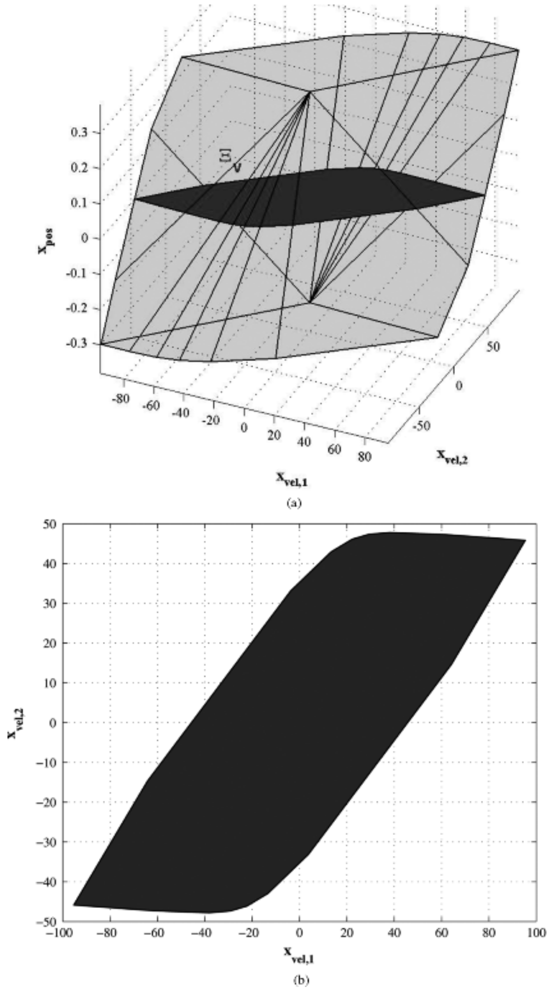


Fig. 5. Invariant set $\Xi(t_e)$ and its cross section for the identified OAV model in one spatial dimension, showing the resulting Ξ_v constraint set, which has to be enforced on states related to the vehicle's velocity in order to remain inside the invariant region using the emergency controller. (a) The maximal positively invariant set in the 3-D state space. (b) Cross section of the invariant set at zero position, representing velocity state constraints Ξ_v.

and $x^i_{k,\mathrm{pos}}$ is the vector of x and y coordinates and $x^i_{k,\mathrm{vel}}$ denotes a vector of states corresponding to x-axis and y-axis velocity components. The speed and acceleration of each OAV are constrained as in (3).

The maximal positively invariant set was calculated for the identified OAV model using the following protection zone, velocity, and position command limits:

$$\begin{aligned} y_{\mathrm{pos}} &\in \mathcal{Y}^{\mathrm{ER}}_p(t_e) = \{y_{\mathrm{pos}} \in \mathbb{R}^2 | \|y_{t_e,\mathrm{pos}} - y_{\mathrm{pos}}\|_\infty \le 0.3 \text{ ft}\} \\ x_{\mathrm{vel}} &\in \mathcal{X}^{\mathrm{ER}}_v = \{x_{\mathrm{vel}} \in \mathbb{R}^4 | \|y_{\mathrm{vel}}\|_\infty \le 24 \text{ ft/s}\} \\ u &\in \mathcal{U}^{\mathrm{ER}} = \{u \in \mathbb{R}^2 | \|u\|_\infty \le 500 \text{ ft}\} \end{aligned} \tag{33}$$

and an LQR regulator as the emergency controller. The polytopic invariant set is shown in Fig. 5(a) for only three states corresponding to one spatial dimension. The coordinate axes correspond to these three states, two of which are related to the velocity output of the model, and one represents the vehicle's position. The cross section of the invariant set at zero position is also highlighted in Fig. 5(a). This cross section is depicted separately in Fig. 5(b) and will be a polytope as well. This set represents the additional state constraints that must be enforced on velocity-related states in the decentralized RHC scheme in order to guarantee collision avoidance as described in Sections V-B and V-C.

The prediction model used in the receding horizon scheme is constructed using "Δu" formulation

$$\begin{bmatrix} \hat{x}^i_{k+1} \\ u^i_k \end{bmatrix} = \overbrace{\begin{bmatrix} A & B \\ 0 & I_2 \end{bmatrix}}^{A^i} \begin{bmatrix} \hat{x}^i_k \\ u^i_{k-1} \end{bmatrix} + \overbrace{\begin{bmatrix} B \\ I_2 \end{bmatrix}}^{B^i} \Delta u^i_k$$

$$\hat{y}^i_k = \underbrace{[C \quad D]}_{C^i} \begin{bmatrix} \hat{x}^i_k \\ u^i_{k-1} \end{bmatrix} + \underbrace{D}_{D^i} \Delta u_k. \tag{34}$$

It is important to emphasize that the approach proposed in this paper can easily accommodate heterogeneous, higher order, more complex linear or piecewise-linear models that describe the particular UAV dynamics with higher fidelity.

Given the system dynamics and a "two-closest-neighbor" interconnection policy, each node solves the decentralized optimization problem (26) with:

1) linear cost function

$$\begin{aligned} l^i\left(x^i_{k,t}, \Delta u^i_{k,t}, \tilde{x}^i_{k,t}, \Delta\tilde{u}^i_{k,t}\right) &= \left\|Q_{\Delta u}[\Delta u^i_{k,t}, \Delta\tilde{u}^i_{k,t}]\right\|_\infty + \\ \max\Bigg(&\left\|Q_{\text{pos}}\left(y^i_{k,\text{pos}} - y^i_{f,\text{pos}}\right)\right\|_\infty, \\ &\left\|Q_{\text{vel}}\left(y^i_{k,\text{vel}} - y^i_{f,\text{vel}}\right)\right\|_\infty, \\ \max_{j,(i,j)\in\mathcal{A}(t)} &\left\|\tilde{Q}_{\text{pos}}\left(y^j_{k,\text{pos}} - y^j_{f,\text{pos}}\right)\right\|_\infty, \\ \max_{j,(i,j)\in\mathcal{A}(t)} &\left\|\tilde{Q}_{\text{vel}}\left(y^j_{k,\text{vel}} - y^j_{f,\text{vel}}\right)\right\|_\infty, \\ \max_{j,(i,j)\in\mathcal{A}(t)} &\left\|Q_{r\text{pos}}\left(\left(y^i_{k,\text{pos}} - y^j_{k,\text{pos}}\right) - \left(y^i_{f,\text{pos}} - y^j_{f,\text{pos}}\right)\right)\right\|_\infty, \\ \max_{j,(i,j)\in\mathcal{A}(t)} &\left\|Q_{r\text{vel}}\left(\left(y^i_{k,\text{vel}} - y^j_{k,\text{vel}}\right) - \left(y^i_{f,\text{vel}} - y^j_{f,\text{vel}}\right)\right)\right\|_\infty, \\ \max_{\substack{q,r,(i,q)\in\mathcal{A}(t)\\(i,r)\in\mathcal{A}(t)}} &\left\|\tilde{Q}_{r\text{pos}}\left(\left(y^q_{k,\text{pos}} - y^r_{k,\text{pos}}\right) - \left(y^q_{f,\text{pos}} - y^r_{f,\text{pos}}\right)\right)\right\|_\infty, \\ \max_{\substack{q,r,(i,q)\in\mathcal{A}(t)\\(i,r)\in\mathcal{A}(t)}} &\left\|\tilde{Q}_{r\text{vel}}\left(\left(y^q_{k,\text{vel}} - y^r_{k,\text{vel}}\right) - \left(y^q_{f,\text{vel}} - y^r_{f,\text{vel}}\right)\right)\right\|_\infty \Bigg); \end{aligned}$$

2) no terminal cost and constraint $l^i_N(x^i_{N,t}, \tilde{x}^i_{N,t}) = 0$, $\mathcal{X}_f \equiv \mathbb{R}^4$;

3) identical vehicle dynamics;

4) linear constraints on inputs and outputs

$$|\Delta u| \leq [50 \quad 50]', \quad |y| \leq [500 \quad 500 \quad 24 \quad 24]'$$

5) nonconvex interconnection constraints

$$\begin{aligned} g^{i,j}\left(x^i_k, u^i_k, x^j_k, u^j_k\right) &= d_{\text{safe}} - \left\|y^i_{k,\text{pos}} - y^j_{k,\text{pos}}\right\|_\infty \\ g^{q,r}\left(x^q_k, u^q_k, x^r_k, u^r_k\right) &= d_{\text{safe}} - \left\|y^q_{k,\text{pos}} - y^r_{k,\text{pos}}\right\|_\infty. \end{aligned}$$

Note that the cost function includes terms that weigh the maximum control effort and the infinity norm of a vector which collects all the absolute and relative errors of the ith node and its neighbors with respect to the final reference values. The infinity norm of a vector is defined as $\|v\|_\infty \triangleq \max_i |v_i|$, where $v = [v_1 \; v_2 \; \cdots \; v_n]'$.

Notice also, that the interconnection constraints in item 5 define square protection zones around vehicles that cannot intersect each other. The scaled separation distance was chosen to be $d_{\text{safe}} = 0.6$ in each spatial dimension. Solutions generated by node i enforce these collision avoidance constraints not only between itself and its neighbors, but among the neighbors as well.

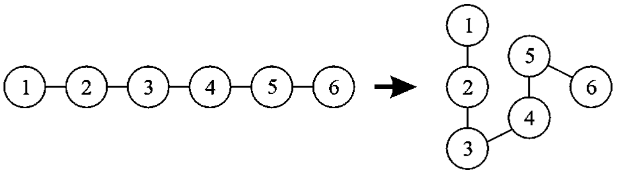


Fig. 6. Six-vehicle formation graphs.

A. Stationary Surveillance With Formation Change

The first high-fidelity simulation example illustrates a stationary formation change for surveillance. The problem setup is defined by assigning the fixed graph shown in Fig. 6 to the six vehicles that are lined up one after another

$$\mathcal{A}(t) = \{(1,2), (2,3), (3,4), (4,5), (5,6)\}, \quad \text{for all } t \geq 0.$$

The objective is to move the vehicles into a wedge formation given in Fig. 6, representing a change in formation for stationary surveillance. The vehicles must get from their initial positions y_0 to designated target points y_f while taking up the associated formation defined by the relative positions and speeds at the targets

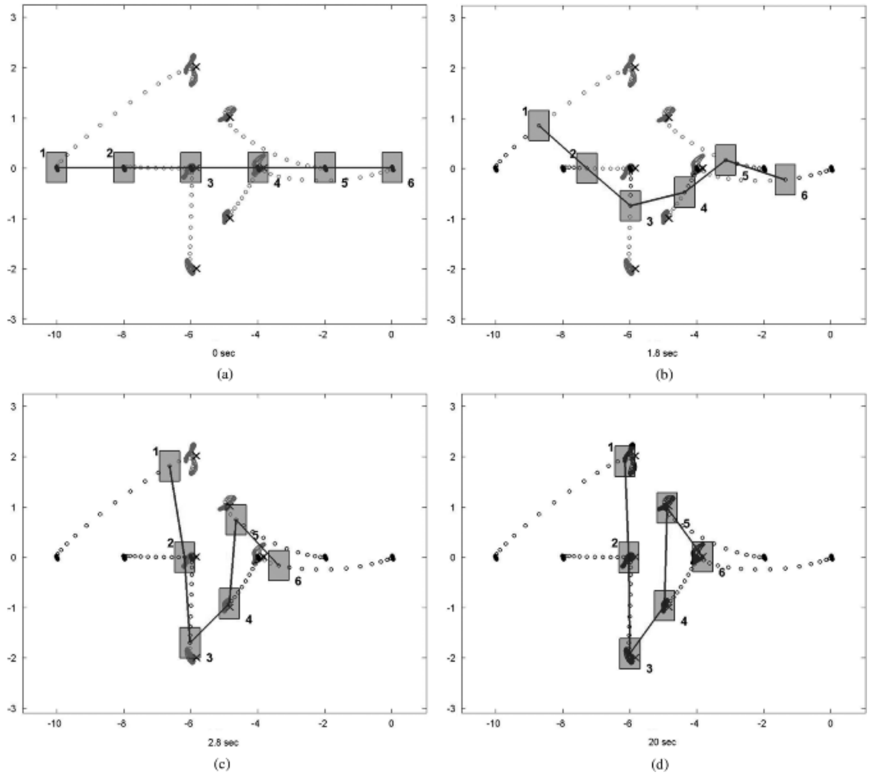
$$\begin{aligned} y^1_0 &= [-10 \quad 0 \quad 0 \quad 0]', & y^1_f &= [-6 \quad 2 \quad 0 \quad 0]', \\ y^2_0 &= [-8 \quad 0 \quad 0 \quad 0]', & y^2_f &= [-6 \quad 0 \quad 0 \quad 0]', \\ y^3_0 &= [-6 \quad 0 \quad 0 \quad 0]', & y^3_f &= [-6 \quad -2 \quad 0 \quad 0]', \\ y^4_0 &= [-4 \quad 0 \quad 0 \quad 0]', & y^4_f &= [-5 \quad -1 \quad 0 \quad 0]', \\ y^5_0 &= [-2 \quad 0 \quad 0 \quad 0]', & y^5_f &= [-5 \quad 1 \quad 0 \quad 0]', \\ y^6_0 &= [0 \quad 0 \quad 0 \quad 0]', & y^6_f &= [-4 \quad 0 \quad 0 \quad 0]'. \end{aligned}$$

Protection zones of each vehicle were given as $d_{\text{safe}}/2 = 0.3$ in each spatial dimension. Note that from a computational point of view, the decentralized problems solved by each vehicle involve at most three vehicles.

The decentralized scheme (26)–(28) is applied to the problem using a prediction horizon length of 20 steps (4 s). Weights in the cost function were chosen to be $Q_{\Delta u} = 5I_n$ for all vehicles, where $I_n \in \mathbb{R}^n$ denotes the identity matrix. The dimension of the weight $Q_{\Delta u}$ is determined by how many vehicles and control inputs are involved in the optimization problem (e.g., OAV #1 solves for its own two control inputs and the two inputs of its single neighbor, which means $Q_{\Delta u} \in \mathbb{R}^4$). Other weights were chosen as $Q_{\text{pos}} = \tilde{Q}_{\text{pos}} = 20I_2$ and $Q_{\text{vel}} = \tilde{Q}_{\text{vel}} = Q_{r\text{pos}} = Q_{r\text{vel}} = \tilde{Q}_{r\text{pos}} = \tilde{Q}_{r\text{vel}} = 10I_2$.

The previous choice of dynamics, cost, and constraints allow us to rewrite problem (26) as a MILP [12], [47] for which efficient branch-and-bound solvers are available [48]. Note that any other linear or piecewise linear formulation of constraints, cost, and dynamics can be cast as an MILP [47].

Fig. 7 presents snapshots of this six-vehicle simulated scenario. The results indicate that the proposed framework can be applied successfully to the OAV formation control problem. The hierarchical decomposition approach leads to simplified



(a) (b) (c) (d)

Fig. 7. Snapshots of the stationary formation change simulation example.

high-level vehicle dynamics, which enables the use of sophisticated decentralized high-level controllers and provide excellent performance.

B. Moving Formation With Obstacle Avoidance and Splitting

The second, more complex scenario illustrates the effectiveness of the proposed scheme for a group of six vehicles performing popup obstacle avoidance and formation joining and splitting. The interconnection graph will be time-varying in this case, and is obtained by having each OAV communicate with at most two of its closest neighbors. Depending on the line-of-sight obstruction caused by obstacles, the number of visible neighbors might be less than two at a certain time instant. This closest-neighbor interconnection policy means that the graph becomes directed.

The parameters of the local optimization problems were chosen to be exactly the same as in the first example, except for the reference values. Absolute position references were not specified, only absolute velocity (2 ft/s x-axis velocity) and relative position and velocity values corresponding to the wedge vehicle formation described in the first example.

Two popup obstacles appear during the simulation, at 3 and 9 s, respectively. The obstacles block communication links, which is reflected in the time-varying interconnection graph throughout the simulation. The formation is instructed by the mission manager to split into two subgraphs at 21 s into the simulation and move in different directions, by artificially restricting communication among vehicles in different groups and changing their absolute velocity references.

Fig. 8 depicts 12 snapshots of this scenario with six vehicles including popup obstacles, formation joining, and splitting. Initially, six OAVs take off individually and hover in a straight line formation (time frame A in Fig. 8). Then the process of reconfiguration into a wedge formation is started (time frame B). A popup obstacle is avoided by OAV #1 before the reconfiguration is completed (time frame C). All vehicles travel in the wedge formation for some time (time frame D). Another popup obstacle breaks the formation into two groups (and, therefore, prevents communication among them). Group A consists of OAVs #1, #5, and Group B includes OAVs #2, #3, #4, #6. Group members can communicate again after clearing obstacle (time frame E). After the popup obstacle is avoided, the wedge formation is reestablished (time frame F). Another reconfiguration takes place as the OAVs split into two groups. OAVs #1, #2, #5 enter into a triangular formation while OAVs #3, #4, #6 establish a straight line formation (time frame G). At the end of the simulation, all vehicles land on the ground (time frame H). Full movies of both simulation examples can be found at http://www.cds.caltech.edu/~tamas/simulations.html.

VII. Future Work and Concluding Remarks

Hovering OAVs are very sensitive to wind gusts. All the issues discussed previously become more complicated if we take this into account. The decentralized scheme has to be robust to wind gust disturbances which affect the position and speed of the vehicle. Although there is usually enough degrees-of-freedom such that a specific communication topology can be chosen to obtain maximum disturbance rejection properties, the OAV's limited actuation authority suggests a slightly different approach to this problem. This topic is a subject of current research at Honeywell Laboratories.

An overview of recent work on OAV formation flight carried out at the Honeywell Laboratories in Minneapolis was presented. The problem of OAV formation flight was addressed

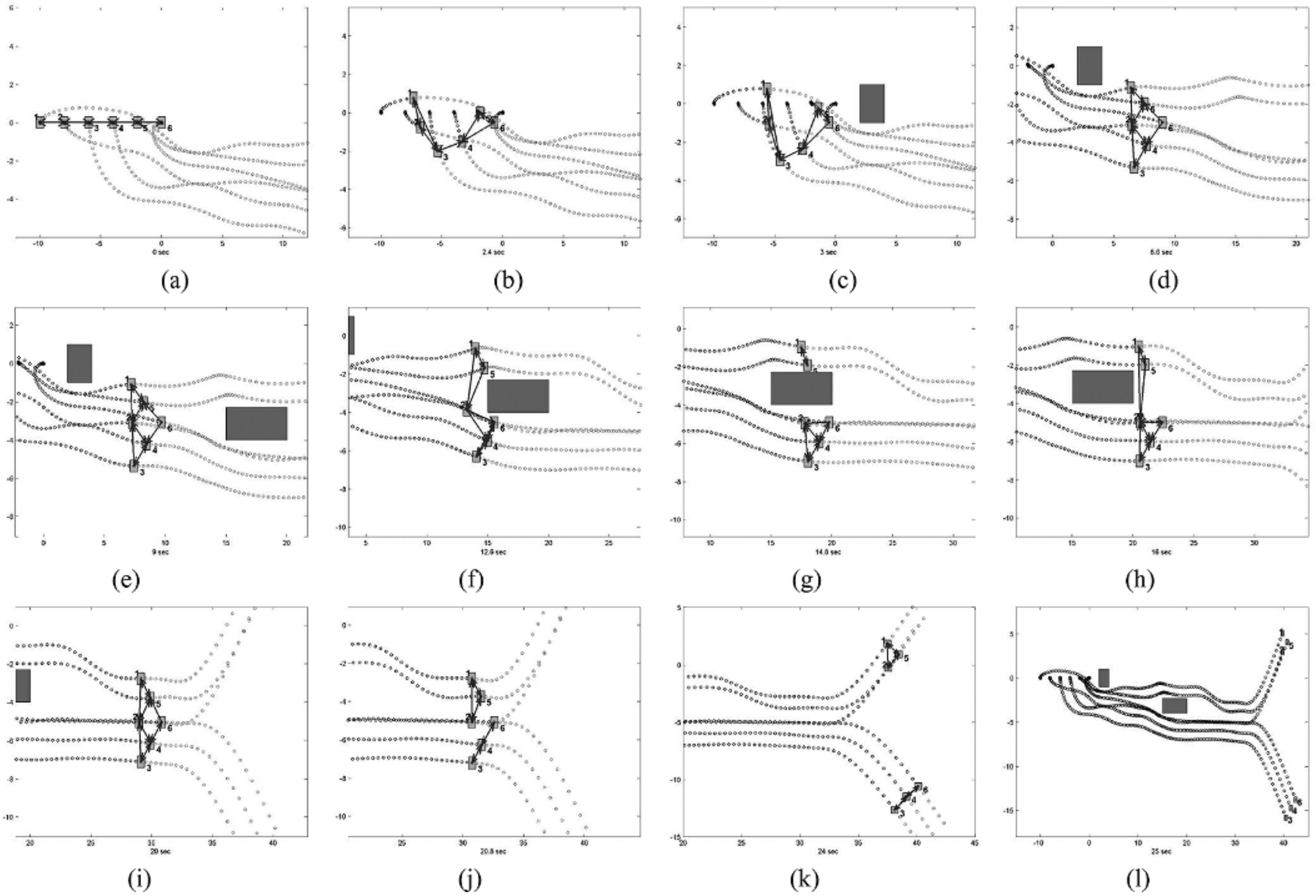


Fig. 8. Snapshots of the formation joining/splitting simulation example with obstacle avoidance. (a) Time frame A. (b) Time frame B. (c) Time frame C. (d) Time frame C, D. (e) Time frame D, E. (f) Time frame E. (g) Time frame F. (h) Time frame F. (i) Time frame F. (j) Time frame G. (k) Time frame G. (l) Time frame H.

by decomposition in a hierarchical fashion. The dynamics of the OAV and its inner-loop control design was discussed. A higher level decentralized optimization-based approach was presented to achieve OAV formation flight and avoid collisions using emergency switching control based on invariant sets as protection zones. The overall scheme was formulated as mixed-integer linear programs of small sizes which are translated into equivalent piecewise affine state-feedback controllers. These can then be downloaded as corresponding lookup tables to the hardware platform of the vehicle and run in real time. The main features and advantages, as well as major issues and current research topics were highlighted.

REFERENCES

[1] D. Stipanovic, G. Inalhan, R. Teo, and C. J. Tomlin, “Decentralized overlapping control of a formation of unmanned aerial vehicles,” *Automatica*, vol. 40, no. 8, pp. 1285–1296, 2004.

[2] W. Ren and R. Beard, “A decentralized scheme for spacecraft formation flying via the virtual structure approach,” *AIAA J. Guid., Control Dyn.*, vol. 27, no. 1, pp. 73–82, Jan. 2004.

[3] J. D. Wolfe, D. F. Chichka, and J. L. Speyer, “Decentralized controllers for unmanned aerial vehicle formation flight,” presented at the AIAA Guid., Nav., Control Conf., San Diego, CA, 1996, AIAA-1996-3833.

[4] J. T. Feddema, C. Lewis, and D. A. Schoenwald, “Decentralized control of cooperative robotic vehicles: Theory and application,” *IEEE Trans. Robot. Autom.*, vol. 18, no. 5, pp. 852–864, Oct. 2002.

[5] W. Kang, N. Xi, and A. Sparks, “Theory and applications of formation control in a perceptive referenced frame,” in *Proc. 39th IEEE Conf. Dec. Control*, 2000, pp. 352–357.

[6] A. Das, R. Fierro, V. Kumar, J. Ostrowski, J. Spletzer, and C. Taylor, “A vision-based formation control framework,” *IEEE Trans. Robot. Autom.*, vol. 18, no. 5, pp. 813–825, Oct. 2002.

[7] A. Richards, J. Bellingham, M. Tillerson, and J. P. How, “Coordination and control of multiple UAVs,” presented at the AIAA Guid., Nav., Control Conf., Monterey, CA, 2002.

[8] D. Shim, H. Kim, and S. Sastry, “Decentralized reflective, model predictive control of multiple flying robots in dynamic enviroment,” Dept. Elect. Eng. Comput. Sci., Univ. California, Berkeley, 2003.

[9] W. B. Dunbar and R. M. Murray, “Model predictive control of coordinated multi-vehicle formation,” in *Proc. 41th IEEE Conf. Dec. Control*, 2002, pp. 4631–4636.

[10] R. Olfati-Saber, W. B. Dunbar, and R. M. Murray, “Cooperative control of multi-vehicle systems using cost graphs and optimization,” in *Proc. Amer. Control Conf.*, 2003, pp. 2217–2222.

[11] F. Giulietti, L. P. L. , and M. Innocenti, “Autonomous formation flight,” *IEEE Control Syst. Mag.*, vol. 20, no. 6, pp. 34–44, Nov. 2000.

[12] A. Richards and J. P. How, “Aircraft trajectory planning with collision avoidance using mixed integer linear programming,” in *Proc. Amer. Control Conf.*, 2002, pp. 1936–1941.

[13] F. Borrelli, D. Subramanian, A. U. Raghunathan, L. T. Biegler, and T. Samad, “MILP and NLP techniques for centralized trajectory planning of multiple unmanned air vehicles,” in *Proc. Amer. Control Conf.*, 2006, pp. 5763–5768.

[14] S. Wang and E. J. Davison, “On the stabilization of decentralized control systems,” *IEEE Trans. Autom. Control*, vol. 18, no. 5, pp. 473–478, Oct. 1973.

[15] M. Rotkowitz and S. Lall, “Decentralized control information structures preserved under feedback,” in *Proc. 41th IEEE Conf. Dec. Control*, 2002, pp. 569–575.

[16] R. D’Andrea and G. E. Dullerud, “Distributed control design for spatially interconnected systems,” *IEEE Trans. Autom. Control*, vol. 48, no. 9, pp. 1478–1495, Sep. 2003.

[17] M. Aldeen, “Class of stabilising decentralised controllers for interconnected dynamical systems,” *IEE Proc. D*, vol. 139, no. 2, pp. 125–134, Mar. 1992.

[18] H. Trinh and M. Aldeen, "Decentralised feedback controllers for uncertain interconnected dynamic systems," *IEE Proc. D*, vol. 140, no. 6, pp. 429–434, Nov. 1993.

[19] H. Jiang, H. Cai, J. F. Dorsey, and Z. Qu, "Toward a globally robust decentralized control for large-scale power systems," *IEEE Trans. Control Syst. Technol.*, vol. 5, no. 3, pp. 309–319, May 1997.

[20] N. R. Sandell, P. Varaiya, M. Athans, and M. G. Safonov, "Survey of decentralized control methods for large scale systems," *IEEE Trans. Autom. Control*, vol. 23, no. 2, pp. 108–128, Apr. 1978.

[21] D. D. Siljak, *Decentralized Control of Complex Systems*. New York: Academic, 1990.

[22] D. D. Siljak, "Decentralized control and computations: Status and prospects," *Ann. Rev. Control*, vol. 20, pp. 131–141, 1996.

[23] A. Bicchi and L. Pallottino, "On optimal cooperative conflict resolution for air traffic management systems," *IEEE Trans. Intell. Transport. Syst.*, vol. 1, no. 4, pp. 221–231, Dec. 2000.

[24] M. Egerstedt and C. F. Martin, "Conflict resolution for autonomous vehicles: A case study in hierarchical control design," *Int. J. Hybrid Syst.*, vol. 2, no. 3, pp. 221–234, Sep. 2002.

[25] A. P. Aguiar and J. Hespanha, "Logic-based switching control for trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty," in *Proc. Amer. Control Conf.*, 2004, pp. 3004–3010.

[26] J. van Schuppen, "Decentralized supervisory control with information structures," in *Proc. Int. Workshop Discrete Event Syst. (WODES)*, 1998, pp. 36–41.

[27] P. Ögren, M. Egerstedt, and X. Hu, "A control Lyapunov function approach to multi-agent coordination," *IEEE Trans. Robot. Autom.*, vol. 18, no. 5, pp. 847–851, Oct. 2002.

[28] A. Benveniste, E. Fabre, and S. Haar, "Markov nets: Probabilistic models for distributed and concurrent systems," *IEEE Trans. Autom. Control*, vol. 48, no. 11, pp. 1936–1950, Nov. 2003.

[29] T. Keviczky, F. Borrelli, and G. J. Balas, "A study on decentralized receding horizon control for decoupled systems," in *Proc. Amer. Control Conf.*, 2004, pp. 4921–4926.

[30] T. Keviczky, F. Borrelli, and G. J. Balas, "Decentralized receding horizon control for large scale dynamically decoupled systems," *Automatica*, vol. 42, no. 12, pp. 2105–2115, Dec. 2006.

[31] D. Enns, D. Bugajski, R. Hendrick, and G. Stein, "Dynamic inversion: An evolving methodology for flight control design," *Int. J. Control*, vol. 59, no. 1, pp. 71–91, Jan. 1994.

[32] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, Jun. 2000.

[33] T. Keviczky, "Decentralized receding horizon control of large scale dynamically decoupled systems," Ph.D. dissertation, Control Sci. Dynam. Syst. Center, Univ. Minnesota, Minneapolis, 2005.

[34] W. B. Dunbar and R. M. Murray, "Receding horizon control of multi-vehicle formations: A distributed implementation," in *Proc. 43th IEEE Conf. Dec. Control*, 2004.

[35] E. Camponogara, D. Jia, B. Krogh, and S. Talukdar, "Distributed model predictive control," *IEEE Control Syst. Mag.*, vol. 22, no. 1, pp. 44–52, Feb. 2002.

[36] A. Richards and J. How, "A decentralized algorithm for robust constrained model predictive control," in *Proc. Amer. Control Conf.*, 2004, pp. 4261–4266.

[37] N. Motee and A. Jadbabaie, "Distributed receding horizon control of spatially invariant systems," in *Proc. Amer. Control* , 2006, pp. 731–736.

[38] T. Keviczky, F. Borrelli, and G. J. Balas, "Hierarchical design of decentralized receding horizon controllers for decoupled system," in *Proc. 43th IEEE Conf. Dec. Control*, 2004, pp. 1592–1597.

[39] E. C. Kerrigan, "Robust constraint satisfaction: Invariant sets and predictive control," Ph.D. dissertation, Dept. Eng., Univ. Cambridge, Cambridge, U.K., 2000.

[40] R. Teo and C. J. Tomlin, "Computing danger zones for provably safe closely spaced parallel approaches," *AIAA J. Guid., Control, Dyn.*, vol. 26, no. 3, pp. 434–443, May/Jun. 2003.

[41] F. Blanchini, "Set invariance in control," *Automatica*, vol. 35, pp. 1747–1767, 1999.

[42] T. Schouwenaars, J. How, and E. Feron, "Receding horizon path planning with implicit safety guarantees," in *Proc. Amer. Control Conf.*, 2004, pp. 5576–5581.

[43] T. Schouwenaars, B. D. Moor, E. Feron, and J. How, "Mixed integer programming for multi-vehicle path planning," in *Proc. Eur. Control Conf.*, 2001, pp. 2603–2608.

[44] T. Keviczky, B. Vanek, F. Borrelli, and G. J. Balas, "Hybrid decentralized receding horizon control of vehicle formations," presented at the Amer. Control Conf., Minneapolis, MN, June 2006.

[45] F. Borrelli, T. Keviczky, G. J. Balas, G. Stewart, K. Fregene, and D. Godbole, "Hybrid decentralized control of large scale systems," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Control and Information Sciences. New York: Springer–Verlag, 2005.

[46] F. Borrelli, *Constrained Optimal Control of Linear and Hybrid Systems*, ser. Lecture Notes in Control and Information Sciences. New York: Springer, 2003, vol. 290.

[47] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, no. 3, pp. 407–427, Mar. 1999.

[48] ILOG, Inc., Gentilly Cedex, France, "CPLEX 7.0 user manual," 2000.

Tamás Keviczky (M'03) received the M.S. degree in electrical engineering from the Budapest University of Technology and Economics, Budapest, Hungary, in 2001, and the Ph.D. degree from the Control Science and Dynamical Systems Center, University of Minnesota, Minneapolis, in 2005.

He is currently an Assistant Professor with the Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands. He has been a Postdoctoral Scholar with the Control and Dynamical Systems, California Institute of Technology, Pasadena. His research interests include optimization, model predictive control, aerospace and automotive applications, and distributed constrained optimization-based control of multiple vehicles and large-scale systems.

Dr. Keviczky was a corecipient of the 2005 AACC O. Hugo Schuck Best Paper Award for Practice for joint work with Prof. G. J. Balas (University of Minnesota).

Francesco Borrelli (M'06) received the "Laurea" degree in computer science engineering from the University of Naples "Federico II," Naples, Italy, in 1998, and the Ph.D. degree from the Automatic Control Laboratory, ETH-Zurich, Zurich, Switzerland, in 2002.

He has been a Research Assistant with the Automatic Control Laboratory, ETH-Zurich, and a Contract Assistant Professor with the Department of Aerospace Engineering and Mechanics, University of Minnesota, Minneapolis. He is currently an Assistant Professor at the "Università del Sannio," Benevento, Italy. He is the author of the book *Constrained Optimal Control of Linear and Hybrid Systems* (Springer Verlag, 2003). His research interests include constrained optimal control, model predictive control, robust control, parametric programming, singularly perturbed systems and automotive applications of automatic control.

Dr. Borrelli was a recipient of the "Innovation Prize 2004" from the ElectroSwiss Foundation.

Kingsley Fregene (S'99–M'03) received the Ph.D. and M.A.Sc. degrees from the University of Waterloo, Waterloo, ON, Canada, in 2002 and 1999, respectively, and the B.Eng. degree (first class honors) from the Federal University of Technology, Owerri, Nigeria, in 1996, all in electrical engineering.

He is a Senior Research Scientist with the Guidance and Control Group, Honeywell Laboratories, Minneapolis. His current interests and R&D activities are in the areas of systems and control for aerospace vehicles, autonomous (multi-) vehicle systems and micro-scale devices (MEMS/NEMS). He has held visiting research positions with the Los Alamos Laboratory, Los Alamos, NM, and Oak Ridge National Laboratory, Oakridge, TN. He has five U.S. patents pending and has also authored/coauthored over 25 journal articles, book chapters, and conference papers that span the areas of power systems control, applied nonlinear control, multirobot systems, control theory, intelligent control, and autonomous vehicles.

Datta Godbole (M'95) received the Ph.D. degree in electrical engineering from University of California, Berkeley, in 1994.

In 1999, he joined Honeywell Laboratories, Minneapolis, MN, where he currently leads a research group in Navigation, Communication, and Control Systems. From 1994 to 1998, he was with California PATH, Berkeley, CA, where he worked on the development of automated and semi-automated highway systems. His research interests include nonlinear and hybrid control systems, air/ground autonomous vehicles, and coordinated operation of multiple manned/unmanned vehicles.

Gary J. Balas (F'89) received the B.Sc. and M.S. degrees in civil and electrical engineering from the University of California, Irvine, in 1982 and 1984, respectively, and the Ph.D. degree in aeronautics and astronautics from the California Institute of Technology, Pasadena, in 1989.

In 1990, he joined the Department of Aerospace Engineering and Mechanics, University of Minnesota, Minneapolis, where he is currently Professor and Head of the Department. He is also Codirector of the Control Science and Dynamical Systems Graduate Program. He is the president of MUSYN Inc. His current research interests include robust and real-time control, linear, parameter-varying systems, receding horizon control, and applications in the aerospace industry.

Dr. Balas held a McKnight-Land Grant Professorship from 1993 to 1995, was a recipient of the Outstanding Young Investigator Award from the ASME Dynamic Systems and Control Division in 1999, and the University of Minnesota Institute of Technology George Taylor Distinguished Research Award in 2003, was a corecipient of the IEEE Control Systems Technology Award in 2005, the AACC O. Hugo Schuck Award in 2005, and was named a Distinguished McKnight Professor at the University of Minnesota, in 2007.