Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

Decentralized Receding Horizon Control of Cooperative Vehicle Formations

Francesco Borrelli, Tamás Keviczky, Kingsley Fregene, Gary J. Balas

Abstract— **An overview of a novel approach to vehicle formation control is presented. The scheme is based on decentralized and cooperative predictive control. The control scheme is being tested on Unmanned Air Vehicle models at the Honeywell Laboratories in Minneapolis. Each vehicle is equipped with efficient guidance and control loops which enable the implementation of the decentralized scheme for higher-level control and coordination. In this framework, collision avoidance and constraint fulfillment are ensured by using a logic that switches to collision-free emergency maneuvers if necessary. The size of protection zones associated with this logic is determined from invariant set theory. The scheme also makes use of logic rules which improve stability and feasibility of the decentralized method by enforcing coordination. The decentralized control laws which respect the rules are computed using hybrid control design techniques. The proposed decentralized control scheme is formulated as optimization problems of small sizes which can be translated into equivalent piecewise affine state-feedback controllers. These controllers can then be downloaded as corresponding look-up tables to the onboard avionics of the vehicle and run in realtime.**

I. INTRODUCTION

Interest in the formation control of Unmanned Air Vehicles (UAVs) has grown significantly over the last years. One of the main motivations is the wide range of military and civilian applications where UAV formations offer the promise of providing a low cost and efficient alternative to existing technology. Among them, distributed sensing applications are envisioned to be the most appealing. Such applications include Synthetic Aperture Radar (SAR) interferometry, surveillance, damage assessment, reconnaissance, chemical or biological agent monitoring, exploration, vegetation growth analysis, assessment of topographical changes, etc. [1]. These kind of applications require the development of control system design techniques for large and tight formations.

From a control engineering perspective, formation flight can be viewed as a large control problem in which we seek to compute the inputs that drive the vehicles along trajectories that maintain relative positions as well as safe distances between each vehicle pair. As is evident from the literature, optimal control problem formulation has been one of the more successful frameworks used to tackle such a problem [1]–[7]. 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 [4], [5], [8]), it is clear that, as the number of vehicles increases, the solution of such large scale, centralized, non-convex optimization problems becomes prohibitive. This is true even when the most advanced optimization solvers and (over)simplified linear vehicle dynamics are used [9]. The main challenge here is to formulate simpler decentralized problems which result in a formation behavior similar to what is obtainable with a centralized approach.

This paper presents an overview of current work on UAV formation flight carried out at Honeywell Laboratories in Minneapolis. In general, UAVs exhibit nonlinear multivariable dynamic coupled with tight state and inputs constraints. Achieving formation flight for this vehicle is a complex task which is rendered tractable 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 system. For this class of systems, a decentralized optimization-based control framework is developed to achieve formation flight and other cooperative tasks. A schematic diagram of the hierarchical decomposition approach to UAV formation control is shown in Figure 1.

In a recent paper [10] we have proposed a method for designing decentralized receding horizon controllers (RHC) for a certain class of large scale systems. A centralized RHC controller is broken into distinct RHC controllers of smaller sizes. Each RHC controller is associated to a different subsystem and computes the local control inputs based only on the states of the subsystem and of its neighbors.

Along with the benefits of a decentralized design, one has to face inherent issues such as difficulties in ensuring stability and feasibility of the system. An invariant set based extension of the proposed decentralized RHC scheme is suggested in [11], which guarantees collision avoidance by switching to an emergency controller or maneuver when feasibility of the decentralized scheme is lost. In order to have collision-free emergency trajectories, protection zones and invariant sets associated to the emergency controllers are used to specify additional constraints in the decentralized optimization problems. In addition to collision avoidance guarantees, one would prefer a certain level of coordination among the vehicles flying in formation. In [12] it was shown how coordination rules can be included in

F. Borrelli is with the Dipartimento di Ingegneria, Università degli Studi del Sannio, 82100 Benevento, Italy, francesco.borrelli@unisannio.it

T. Keviczky and G. J. Balas are with the Department of Aerospace Engineering and Mechanics, University of Minnesota, Minneapolis, MN 55455, USA, {keviczky,balas}@aem.umn.edu

Kingsley Fregene is with the Honeywell Laboratories, Minneapolis, MN 55418, USA, kingsley.fregene@honeywell.com

the decentralized control design by using hybrid system techniques. Such rules improve the overall behavior of the systems and make the control subproblems feasible where traditional design is either infeasible or too conservative. The hybrid constrained optimal control problems are formulated in discrete time [13]. In particular, computational results are obtained by using on-line mixed-integer optimization [13] or the evaluation of an equivalent lookup table obtained by means of parametric programming [14]. However, the main concepts presented in this paper are applicable to any control scheme and design methodology as longs as it can cope with continuous dynamics and logic rules. Theoretical proofs of stability and feasibility in such design schemes are under investigation [15] but in general difficult to give. Nevertheless, the benefits and practicality of these techniques have been proven by extensive simulations.

This paper gives a tutorial overview of these invariant set based and hybrid decentralized RHC techniques applied for the control of vehicle formations. Section II introduces the decentralized control scheme used to approach a general class of large scale control problems, including formation flight. Section III gives a brief synopsis of the collision-free extension of the basic RHC scheme. Benefits of applying hybrid control techniques and possible ways of incorporating logic-based coordination within the decentralized framework are highlighted in Section IV. Computational tools that facilitate the solution approach are highlighted in Section V. Section VI contains some concluding remarks.

II. PROBLEM FORMULATION

A concise description of the decentralized RHC scheme proposed in [10] follows. Consider a set of N_v linear decoupled dynamical systems, the i-th system being described by the discrete-time time-invariant state equation:

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

where $x_k^i \in \mathbb{R}^{n^i}$, $u_k^i \in \mathbb{R}^{m^i}$, $f^i : \mathbb{R}^{n^i} \times \mathbb{R}^{m^i} \to \mathbb{R}^{n^i}$ are state, input and state update function of the i-system, respectively. Let $\mathcal{X}^i \subseteq \mathbb{R}^{n^i}$ and $\mathcal{U}^i \subseteq \mathbb{R}^{m^i}$ denote the set of feasible states and inputs of the i-th system, respectively:

$$x_k^i \in \mathcal{X}^i, \quad u_k^i \in \mathcal{U}^i, \ k \geq 0 \tag{2}$$

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

We will refer to the set of N_v constrained systems as a *team system*. Let $\tilde{x}_k \in \mathbb{R}^{N_v \times n^i}$ and $\tilde{u}_k \in \mathbb{R}^{N_v \times m^i}$ be the vectors which collect the states and inputs of the team 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

$$\tilde{x}_{k+1} = f(\tilde{x}_k, \tilde{u}_k) \tag{3}$$

We denote by (x_e^i, u_e^i) the equilibrium pair of the i-th system and $(\tilde{x}_e, \tilde{u}_e)$ the corresponding equilibrium for the team system.

So far the individual systems belonging to the team system are completely decoupled. We consider an optimal control problem for the team system where cost function and constraints couple the dynamic behavior of individual systems. We use a graph topology to represent the coupling in the following way. We associate the i-th system to the i-th node of the graph, and if an edge (i, j) connecting the i-th and j-th 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)

$$\mathcal{G}(t) = \{\mathcal{V}, \mathcal{A}(t)\} \tag{4}$$

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 arcs (i, j) with $i \in \mathcal{V}, \ j \in \mathcal{V}$.

Once the graph structure has been fixed, the optimization problem is formulated as follows. Denote with $\tilde{x}_k^i$ the states of all neighbors of the i-th system at time k, i.e. $\tilde{x}_k^i = \{x_k^j \in \mathbb{R}^{n^j} | (j, i) \in \mathcal{A}(k)\}$, $\tilde{x}_k^i \in \mathbb{R}^{\tilde{n}_k^i}$ with $\tilde{n}_k^i = \sum_{j|(j,i) \in \mathcal{A}(k)} n_k^j$. Analogously, $\tilde{u}_k^i \in \mathbb{R}^{\tilde{m}_k^i}$ denotes the inputs to all the neighbors of the i-th system at time k. Let

$$g^{i,j}(x^i, x^j) \leq 0 \tag{5}$$

define interconnection constraints between the i-th and the j-th systems, 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 form of the interconnection constraints defined between the i-th system and all its neighbors:

$$g_k^i(x_k^i, \tilde{x}_k^i) \leq 0 \tag{6}$$

with $g_k^i : \mathbb{R}^{n^i} \times \mathbb{R}^{\tilde{n}_k^i} \to \mathbb{R}^{nc^{i,k}}$.

Consider the following cost

$$l(\tilde{x}, \tilde{u}) = \sum_{i=1}^{N_v} l_k^i(x^i, u^i, \tilde{x}_k^i, \tilde{u}_k^i) \tag{7}$$

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 to the i-th system and is a function of its states and the states of its neighbor nodes. Assume that l is a positive convex function with $l(\tilde{x}_e, \tilde{u}_e) = 0$.

In the preliminary study [10], the complexity associated to a centralized optimal control design for such class of large scale systems is tackled by formulating N_v decentralized finite time optimal control problems, each one associated to a different node as detailed next. Each node has information about its current states and its neighbors' current states. Based on such information, each node computes its optimal inputs and its neighbors' optimal inputs. The input to the neighbors will only be used to predict their trajectories and then discarded, while the first component of the optimal input to the node will be implemented where it was computed. The details of the algorithm presented in [10] are discussed next. Let the following finite time optimal control problem $\mathcal{P}_i(t)$ with optimal value function $J_N^{i*}(x_t^i, \tilde{x}_t^i)$ be associated to the i-th system at time t

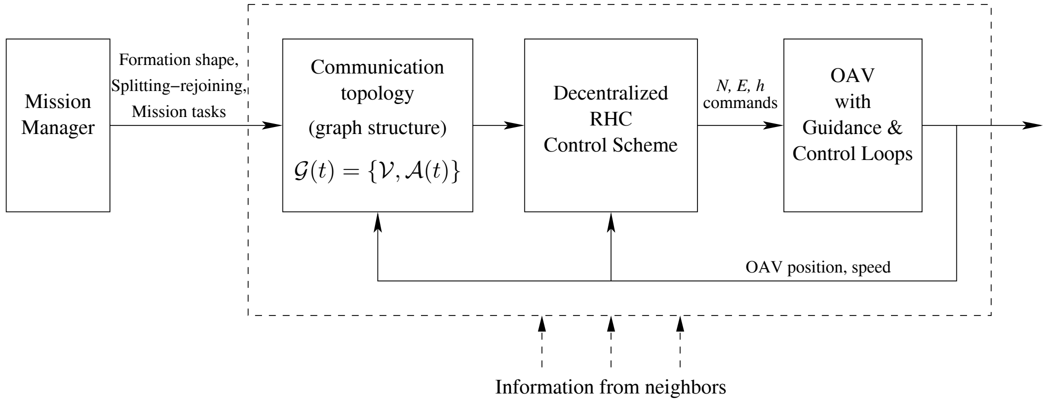

Fig. 1. Decentralized scheme for UAV formation control.

$$\min_{\tilde{U}_t^i} \sum_{k=0}^{N-1} l_t^i(x_{k,t}^i, u_{k,t}^i, \tilde{x}_{k,t}^i, \tilde{u}_{k,t}^i) + l_N^i(x_{N,t}^i, \tilde{x}_{N,t}^i) \quad (8a)$$

$$\text{subj. to} \quad x_{k+1,t}^i = f^i(x_{k,t}^i, u_{k,t}^i), \quad (8b)$$

$$x_{k,t}^i \in \mathcal{X}^i, \quad u_{k,t}^i \in \mathcal{U}^i, \quad k = 1, \ldots, N-1 \quad (8c)$$

$$x_{k+1,t}^j = f^j(x_{k,t}^j, u_{k,t}^j), \quad (i,j) \in \mathcal{A}(t), \quad (8d)$$

$$x_{k,t}^j \in \mathcal{X}^i, \quad u_{k,t}^j \in \mathcal{U}^j, \quad (i,j) \in \mathcal{A}(t), \quad k = 1, \ldots, N-1 \quad (8e)$$

$$g^{i,j}(x_{k,t}^i, u_{k,t}^i, x_{k,t}^j, u_{k,t}^j) \leq 0, \quad (i,j) \in \mathcal{A}(t), k = 1, \ldots, N-1 \quad (8f)$$

$$g^{q,r}(x_{k,t}^q, u_{k,t}^q, x_{k,t}^r, u_{k,t}^r) \leq 0, \quad (q,r) \in \mathcal{A}(t),\ (q,i) \in \mathcal{A}(t),\ (r,i) \in \mathcal{A}(t), \quad k = 1, \ldots, N-1 \quad (8g)$$

$$x_{N,t}^i \in \mathcal{X}_f^i, \quad x_{N,t}^j \in \mathcal{X}_f^j, \quad (i,j) \in \mathcal{A}(t) \quad (8h)$$

$$x_{0,t}^i = x_t^i, \quad \tilde{x}_{0,t}^i = \tilde{x}_t^i \quad (8i)$$

where $\tilde{U}_t^i \triangleq [u_{0,t}^i, \tilde{u}_{0,t}^i, \ldots, u_{N-1,t}^i, \tilde{u}_{N-1,t}^i]' \in \mathbb{R}^s$, $s \triangleq (\tilde{m}^i + m^i)N$ denotes the optimization vector, $x_{k,t}^i$ denotes the state vector of the i-th node predicted at time $t+k$ obtained by starting from the state x_t^i and applying to system (1) the input sequence $u_{0,t}^i, \ldots, u_{k-1,t}^i$. The tilded vectors denote the prediction vectors associated to the neighboring systems *assuming a constant interconnection graph*. Denote by $\tilde{U}_t^{i*} = [u_{0,t}^{*i}, \tilde{u}_{0,t}^{*i}, \ldots, u_{N-1,t}^{*i}, \tilde{u}_{N-1,t}^{*i}]$ the optimizer of problem $\mathcal{P}_i(t)$. Note that problem $\mathcal{P}_i(t)$ involves only the state and input variables of the i-th node 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 node 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 node i implements the first sample of $\tilde{U}_t^{i*}$

$$u_t^i = u_{0,t}^{*i}. \quad (9)$$

4) Each node repeats steps 1 to 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 node 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 node 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 i-th optimal input of problem $\mathcal{P}_i(t)$ will be implemented on the i-th node. The solution of the i-th subproblem will yield a control policy for the i-th node of the form $u_t^i = k_t^i(x_t^i, \tilde{x}_t^i)$.

Even if we assume N to be infinite, the decentralized RHC approach described so far does not guarantee that solutions computed locally are globally feasible and stable The reason is simple: At the i-th node the prediction of the neighboring state x^j is done independently from the prediction of problem $\mathcal{P}_j(t)$. Therefore, the trajectory of x^j predicted by problem $\mathcal{P}_i(t)$ and the one predicted by problem $\mathcal{P}_j(t)$, based on the same initial conditions, are different (since in general, $\mathcal{P}_i(t)$ and $\mathcal{P}_j(t)$ will be different). This will imply that constraint fulfillment will be ensured by the optimizer u_t^{*i} for problem $\mathcal{P}_i(t)$ but not for the centralized problem involving the states of all nodes.

Stability and feasibility of decentralized RHC schemes are active research areas in the control community. 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 [10], [16]–[19]. The main research topics include: (i) the choice of the graph topology when it is not fixed or unique, (ii) the choice of local prediction horizons, terminal weights and terminal regions and its effect on the global performance and feasibility.

In the following section, we give a practical approach based on controller switching and invariant sets to address loss of feasibility in the decentralized RHC scheme and avoid collisions between UAVs. Then, we introduce logic-based coordination rules to improve stability and feasibility of the decentralized method using hybrid control design techniques.

III. GUARANTEED COLLISION AVOIDANCE

For the purpose of UAV formation flight, the interconnection constraints defined in (6) will represent safety distance constraints between the i-th UAV and all its neighbors, as described by the formation graph.

In general, collision avoidance guarantees between any UAV pair in a formation flight problem would necessitate the use of a full graph for describing inter-vehicle constraints (since any UAV protection zone should not intersect with any other UAV protection zone). 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 subset of all neighboring vehicles to accomplish formation flight. Due to possible changes of the required formation, this subset is likely to change leading to time-varying interconnection graph unless we restrict ourselves to the study of rigid formations [5], [20].

Choosing the time-dependence of the set of arcs in this graph is not a straightforward problem, especially under communication constraints. Even if we assume that each vehicle can sense or communicate with every other, there are several ways of selecting who will be considered as a neighbor and who will not. Clearly, a small number of neighbors is preferred, otherwise each UAV would solve a centralized problem, if all of them are taken into account. On the other hand, if we assume a more realistic scenario, where not all vehicles can communicate with or sense every other, a particular neighbor-selection policy can easily lead to a disconnected graph, which could prevent the team system from reaching the desired objective. Conditions on the graph structure and different ways of ensuring the connectedness of the time-varying graph using appropriate neighbor-selection rules are under investigation. One practical way of helping the formation to rejoin (regain connectedness) is to have terminal position references in each maneuver, which would guarantee a connected graph using a particular neighbor-selection policy. For practical purposes, we assume that the time-dependence of the set of arcs will be function of the relative distance between the vehicles. We define here the set $\mathcal{A}(t)$ as

$$\mathcal{A}(t) = \{(i,j) \in \mathcal{V} \times \mathcal{V} \ \mid \ \|x^i_{t,pos} - x^j_{t,pos}\| \leq d_{min}\} \quad (10)$$

that is the set of all the arcs, which connect two vehicles whose distance is less than or equal to d_{min}.

As highlighted in Section II the decentralized RHC problems are not guaranteed to be always feasible due to the mismatch between predicted and actual neighbor trajectories. This problem can be approached in different ways. In [10] collision avoidance guarantees are formulated in terms of robustness of the single decentralized scheme to prediction errors on neighbors' trajectories. A somewhat more conservative approach in [21] relies on a hierarchical decomposition of the formation interconnection graph and employs a feasible set projection technique to guarantee feasibility of the decentralized RHC scheme.

In [11] we make use of emergency controllers and their invariant sets to define protection zones and state constraints that guarantee collision avoidance when the local RHC subproblems become infeasible. The basic idea is described next.

An emergency controller is defined for each UAV, which starts to perform an emergency maneuver and controls the aircraft to a given reference if the feasibility of the local decentralized RHC problem is lost. For instance, the emergency controller can be designed to bring the vehicle to a full stop (e.g. in case of a hovering vehicle such as the UAV), or fly around in a circle (if using a fixed wing aircraft). The maximal positively invariant set of the closed-loop vehicle dynamics under emergency control respecting physical constraints can be calculated and used to aid in the selection of protection zone sizes around vehicles, which guarantee collision avoidance. In addition, more restrictive state constraints might also be necessary to guarantee that each vehicle is operated within this invariant set at any time during normal operation. Since the vehicle is always operated within the invariant set of the emergency controlled closed-loop system, switching to emergency control will always lead to collision-free trajectories.

If a linear state-feedback emergency controller is used with the discrete-time linear time-invariant system models and polyhedral state and input constraints, then the resulting closed-loop maximal positively invariant sets will be polyhedra. These sets can be easily computed off-line with simple techniques using polyhedral manipulations based on [22].

If the local RHC problem becomes feasible again, then normal operation resumes and the emergency control law is disactivated. A logic state variable $x^i_{t,L}$ is introduced for each UAV to indicate whether they are operating in normal or emergency mode. This information is transmitted to neighbors to aid in making more accurate predictions.

The RHC scheme (8)-(9) is modified in the following way. At time t

1) Compute graph connection $\mathcal{A}$(t) according to (10)
2) The i-th UAV solves problem $\mathcal{P}_i(t)$ with additional constraints based on the emergency invariant set and measurements of its state x^i_t and the states of all its neighbors $\tilde{x}^i_t$.
3) If the augmented $\mathcal{P}_i(t)$ is feasible then $x^i_{t,L} = 0$ and the i-th UAV implements the first sample of $\tilde{U}^{i*}_t$
$$u^i_t = u^{*i}_{0,t}. \quad (11)$$
4) else $x^i_{t,L} = 1$ is set, and the i-th UAV implements the emergency controller
5) Each UAV repeats steps 1 to 4 at time $t+1$, based on the new state information x^i_{t+1}, $\tilde{x}^i_{t+1}$.

More details of the modified decentralized RHC scheme with collision avoidance guarantees and possible issues are described in [11].

In [11], [12] inter-vehicle coordination (e.g. "right-of-way") rules were established by means of including binary decision variables in the cost function or in the constraints of the local decentralized controllers. These rules have the ability to help in resolving conflicts between planned vehicle trajectories and reduce the likelihood of collisions. In the following section, incorporation of coordination rules within the decentralized control strategy presented in Section II is shown by using hybrid system techniques.

IV. APPLICATION OF HYBRID THEORY IN DECENTRALIZED CONTROL

In general, it is very difficult to provide feasibility guarantees in a constrained decentralized control problem. Nevertheless everyday life is full of decentralized control problems. Although feasible solutions are not always found or even possible at all, these problems are solved day-by-day relying on certain rules that help coordinate the single subsystem efforts. Examples range from traffic laws to behavior of individuals in a community.

This suggests that it can be beneficial to make use of coordination rules in some decentralized engineering control problems as well. Hybrid control design techniques are able to cope with the hybrid nature of a problem governed by differential equations and logic rules. For this reason it is worthwhile to investigate the benefits of hybrid system techniques in implementing coordination rules within the decentralized control framework presented in Section II. A more formal discussion follows.

We define a *rule element* to be a Boolean-valued function operating on the states of a node and its neighbors' states

$$\varrho \; : \; (x^i, \tilde{x}^i) \rightarrow X, \quad X = \{true, false\}\,. \tag{12}$$

We define a *rule* to be a propositional logic statement involving rule elements

$$\mathcal{R} \; : \; (\varrho_1, \varrho_1, \dots, \varrho_{n-1}) \rightarrow X, \quad X = \{true, false\}\,. \tag{13}$$

The logic statement $\mathcal{R}$ is a combination of "not" ($\neg$), "and" ($\wedge$), "or" ($\vee$), "exclusive or" ($\oplus$), "implies" ($\rightarrow$), and "iff" ($\leftrightarrow$) operators. For instance, the following logic expression of

$$\mathcal{R}\left(X_1, \dots, X_{n-1}\right) \leftrightarrow X_n \tag{14}$$

involving Boolean variables $X_1, \dots, X_n$ can be expressed equivalently with its conjunctive normal form (CNF)

$$\bigwedge_{j=1}^{k} \left(\left(\bigvee_{i \in P_j} X_i \right) \bigvee \left(\bigvee_{i \in N_j} \neg X_i \right) \right), N_j, P_j \subseteq \{1, \dots, n\}. \tag{15}$$

The rule holds and its value is "true" if the statement is evaluated as true based on the rule elements. The rule is not respected and its value is "false" when the underlying statement is false.

We introduce two abstract function classes called *coordinating functions*, which operate on a set of rules and the states of a node and its neighbors

$$\mathscr{F}_c^{\mathcal{C}} \; : \; (\Re, x^i, \tilde{x}^i) \rightarrow \mathbb{R} \tag{16}$$

$$\mathscr{F}_c^{bin} \; : \; (\Re, x^i, \tilde{x}^i) \rightarrow \{0, 1\} \tag{17}$$

where $\Re$ is a set of rules defined in (13). The coordinating functions can be defined to have either continuous or binary values. These function classes rely on rules and states of the system and can be included in the cost function ($\mathscr{F}_c^{\mathcal{C}}$) or constraints ($\mathscr{F}_c^{bin}$) of subproblems. This means that the decentralized problem (8) is modified in the following way

$$\begin{aligned} \min_{\tilde{U}_t^i} \quad & J_N^{dec}(x_t^i, \tilde{x}_t^i) + \mathscr{F}_c^{\mathcal{C}}(\Re, x_t^i, \tilde{x}_t^i) \\ \text{subj. to} \quad & \text{constraints (8b)} - \text{(8i)} \\ & g_c\left(x_{t,k}^i, \tilde{x}_{t,k}^i, \mathscr{F}_c^{bin}(\Re, x_{t,k}^i, \tilde{x}_{t,k}^i)\right) \leq 0, \\ & \quad k = 1, \dots, N-1 \end{aligned}$$

where $J_N^{dec}(x_t^i, \tilde{x}_t^i)$ denotes the cost function in (8a).

If chosen appropriately, coordinating functions have the benefit of guiding towards feasible sequences of decentralized solutions. When $\mathscr{F}_c^{\mathcal{C}}$ is used in the cost function, trajectories which respect rules can be penalized less and have a cost which is less than the cost of trajectories, which do not enforce the rules. When $\mathscr{F}_c^{bin}$ is used in the constraints, the local domain of feasibility is reduced to the domain where only trajectories respecting rules are feasible. A crucial assumption underlying this idea is that each component has to abide by the same or at least similar set of rules.

Remark 1: It is important to point out that the approach of this paper to large-scale control problems is independent of the problem formulation. Continuous-time formulations and other hybrid control design techniques could be used as well.

A hybrid system 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.

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 prediction about neighbors' trajectories. This can be easily achieved if protection zones around vehicles are modeled as parallelepipeds and the disjunctions are modeled as binary variables [9]. "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).

The main idea behind inter-vehicle coordination is to make use of "preferred" decisions in the hybrid control problem that arises due to the non-convex collision avoidance constraints. Details can be found in [12], [23]

V. REAL-TIME IMPLEMENTATION

The presence of nonlinearities and constraints on one hand, and the simplicity needed for real-time implementation on the other, would discourage the design of optimal control strategies as presented above.

The computation of decentralized controllers presented in this section 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, results from hybrid systems and optimal control theory are used, which allow us to formulate constrained optimal control problems and compute their equivalent look-up tables, which are easily implementable in real-time on the UAV 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 [14]. 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 AV 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 on-line on each UAV. Therefore, for implementation, in the second phase the explicit PWA form of the RHC law is computed off-line by using the multiparametric mixed integer programming solver presented in [14]. The use of equivalent PWA form of the RHC law will have several advantages. It is immediate to implement on an UAV platform as a simple look-up table of gain-scheduled controllers. It can also be easily verified (an on-line optimization solver is impossible to verify). Its worst case computational time can also be computed immediately.

VI. CONCLUDING REMARKS

An overview of the current work on UAV formation flight carried out at the Honeywell Laboratories in Minneapolis was presented. The problem of UAV formation flight was addressed by decomposition in a hierarchical fashion. A higher level decentralized optimization-based approach was presented to achieve UAV formation flight and avoid collisions using emergency switching control based on invariant sets as protection zones. Desirable formation behavior was accomplished by the use of logic-based coordination rules within the decentralized scheme. The main features and advantages, as well as major issues and current research topics were highlighted. Simulation examples involving two vehicles and a more complex scenario with six vehicles can be found in [24].

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] A. Richards, J. Bellingham, M. Tillerson, and J. P. How, "Coordination and control of multiple UAVs," in *AIAA Guidance, Navigation, and Control Conference, Monterey, CA*, 2002.

[3] D. Shim, H. Kim, and S. Sastry, "Decentralized reflective model predictive control of multiple flying robots in dynamic enviroment," Department of Electical Engineering and Computer Sciences. University of California at Berkeley, Tech. Rep., 2003.

[4] W. B. Dunbar and R. M. Murray, "Model predictive control of coordinated multi-vehicle formation," in *Proc. 41th IEEE Conf. on Decision and Control*, 2002.

[5] R. Olfati-Saber, W. B. Dunbar, and R. M. Murray, "Cooperative control of multi-vehicle systems using cost graphs and optimization," in *Proc. American Contr. Conf.*, 2003.

[6] F. Giulietti, L. P. L., and M. Innocenti, "Autonomous formation flight," *IEEE Control Systems Magazine*, vol. 20, no. 6, pp. 34–44, 2000.

[7] A. Richards and J. P. How, "Aircraft trajectory planning with collision avoidance using mixed integer linear programming," in *Proc. American Contr. Conf.*, 2002.

[8] A. Raghunathan, V. Gopal, D. Subramanian, L. Biegler, and T. Samad, "Dynamic optimization strategies for 3d conflict resolution of multiple aircraft," *AIAA Journal of Guidance, Control and Dynamics, to appear*, 2003.

[9] F. Borrelli, D. Subramanian, A. U. Raghunathan, L. T. Biegler, and T. Samad, "A comparison between mixed integer programming and nonlinear programming techniques for 3D conflict resolution of multiple aircraft," Department of Aerospace Engineering and Mechanics, University of Minnesota, Minneapolis, Tech. Rep., 2003.

[10] T. Keviczky, F. Borrelli, and G. J. Balas, "A study on decentralized receding horizon control for decoupled systems," in *Proc. American Contr. Conf.*, 2004.

[11] F. Borrelli, T. Keviczky, and G. J. Balas, "Collision-free UAV formation flight using decentralized optimization and invariant sets," in *Proc. 43th IEEE Conf. on Decision and Control*, 2004.

[12] 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 Computer Science. Springer Verlag, Mar. 2005.

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

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

[15] T. Keviczky, F. Borrelli, and G. J. Balas, "Stability analysis of decentralized RHC for decoupled systems," in *44th IEEE Conf. on Decision and Control, and European Control Conf. (submitted)*, Seville, Spain, Dec. 2005.

[16] E. Camponogara, D. Jia, B. Krogh, and S. Talukdar, "Distributed model predictive control," *IEEE Control Systems Magazine*, Feb. 2002.

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

[18] V. Gupta, B. Hassibi, and R. M. Murray, "On the synthesis of control laws for a network of autonomous agents," in *Proc. American Contr. Conf.*, 2004.

[19] A. Richards and J. How, "A decentralized algorithm for robust constrained model predictive control," in *Proc. American Contr. Conf.*, 2004.

[20] T. Eren, P. Belhumeur, B. Anderson, and A. Morse, "A framework for maintaining formations based on rigidity," *15th IFAC World Congress on Automatic Control, Barcelona*, pp. 2752–2757, 2002.

[21] T. Keviczky, F. Borrelli, and G. J. Balas, "Hierarchical design of decentralized receding horizon controllers for decoupled systems," in *Proc. 43th IEEE Conf. on Decision and Control*, 2004.

[22] E. C. Kerrigan, "Robust constraint satisfaction: Invariant sets and predictive control," Ph.D. dissertation, Department of Engineering, University of Cambridge, Cambridge, England, 2000.

[23] T. Keviczky, "Decentralized receding horizon control of large scale dynamically decoupled systems," Ph.D. dissertation, Control Science and Dynamical Systems Center, University of Minnesota, Minneapolis, 2005.

[24] http://www.aem.umn.edu/people/students/keviczky/Simulations.html.