Hierarchical Design of Decentralized Receding Horizon Controllers for Decoupled Systems

Tamás Keviczky, Francesco Borrelli, Gary J. Balas*

Abstract—We consider a set of decoupled dynamical systems and an optimal control problem where cost function and constraints couple the dynamical behavior of the systems. The coupling is described through a connected graph where each system is a node and, cost and constraints of the optimization problem associated to each node are only function of its state and the states of its neighbors.

In a recent report [1] we have proposed a method for designing decentralized receding horizon controllers (RHC). Each RHC controller is associated to a different node and computes the local control inputs based only on the states of the node and of its neighbors. For such a decentralized scheme, stability and feasibility can be ensured in different ways, by modifying cost, constraints and communications structure.

In this paper we focus on decentralized RHC control design through hierarchical decomposition. A certain priority is assigned to each node of the graph and nodes with higher priorities compute control laws for nodes with lower priorities. We study how to ensure stability and feasibility of such scheme when the explicit feasibility domains of the decentralized RHC are available. Moreover, we propose a hierarchical RHC control scheme with stability and feasibility guarantees.

I. INTRODUCTION

The interest in decentralized control goes back to the seventies. Wang and Davison were probably the first in [2] to envision the “increasing interest in decentralized control systems” when “control theory is applied to solve problems for large scale systems”. Since then the interest has grown more than exponentially despite some non-encouraging results on the complexity of the problem [3]. Decentralized control techniques today can be found in a broad spectrum of applications ranging from robotics and formation flight to civil engineering. Such a wide interest makes a survey of all the approaches that have appeared in the literature very difficult and goes also beyond the scope of this paper.

Approaches to decentralized control design differ from each other in the assumptions they make on: (i) the kind of interaction between different systems or different components of the same system (dynamics, constraints, objective), (ii) the model of the system (linear, nonlinear, constrained, continuous-time, discrete-time), (iii) the model of information exchange between the systems, and (iv) the control design technique used.

Dynamically coupled systems have been the most studied. In [2] the authors consider a linear time-invariant system and give sufficient conditions for the existence of feedback laws which depend only on partial system outputs. Recently, in [4] the authors introduce the concept of quadratic invariance of a constraint set with respect to a system. The problem of constructing decentralized control systems is formulated as one of minimizing the closed loop norm of a feedback system subject to constraints on the control structure. The authors show that quadratic invariance is a necessary and sufficient condition for the existence of decentralized controllers. In [5] the authors consider spatially interconnected systems, i.e. systems composed of identical linear time-invariant systems which have a structured interconnection topology. By exploiting the interconnection topology, the authors study decentralized analysis and system control design using $\ell_2$-induced norms and LMIs.

In this paper we focus on decoupled systems. The problem of decentralized control for decoupled systems can be formulated as follows. A dynamical system is composed of (or can be decomposed into) distinct dynamical subsystems that can be independently actuated. The subsystems are dynamically decoupled but have common objectives and constraints which make them interact between each other. Typically the interaction is local, i.e. the goal and the constraints of a subsystem are function of only a subset of other subsystems’ states. The interaction will be represented by an “interaction graph”, where the nodes represent the subsystems and an arc between two nodes denotes a coupling term in the goal and/or in the constraints associated to the nodes. Also, typically it is assumed that the exchange of information has a special structure, i.e., it is assumed that each subsystem can sense and/or exchange information with only a subset of other subsystems. Often the interaction graph and the information exchange graph coincide. A decentralized control scheme consists of distinct controllers, one for each subsystem, where the inputs to each subsystem are computed only based on local information, i.e., on the states of the subsystem and its neighbors.

Our interest in decentralized control for dynamically decoupled systems arises from the abundance of networks of independently actuated systems and the necessity of avoiding centralized design when this becomes computationally prohibitive. Networks of vehicles in formation, production units in a power plant, cameras at an airport, mechanical actuators for deforming surface are just a few examples. Each network has its peculiarity. In formation flight for instance the coupling constraints arise from collision avoidance. The interaction graph is full (each vehicle has to avoid all the
other vehicles) but it is often approximated with a time-varying graph based on a “closest spatial neighbors” model.

In a recent paper [1] we have proposed a method for designing decentralized receding horizon controllers (RHC). A centralized RHC controller is broken into distinct RHC controllers of smaller sizes. Each RHC controller is associated to a different node and computes the local control inputs based only on the states of the node and of its neighbors. The main issue regarding decentralized schemes is that the inputs computed locally are, in general, not guaranteed to be globally feasible and to stabilize the overall team. In general, stability and feasibility of decentralized schemes are very difficult to prove and/or too conservative. As in classical RHC design stability and feasibility can be ensured in different ways, by modifying cost and constraints. In decentralized RHC the communications structure is another degree of freedom which can be used for such goal. Enforcing hierarchy in the communication links can be one way to exploit this degree of freedom. In this report we will follow the lines of the hierarchical decompositions approaches which have been proposed in [6], [7], [8]. We focus on hierarchical decomposition of an RHC scheme. We assume that each node of the graph has a certain priority and use a strategy where nodes with higher priorities compute control laws for nodes with lower priorities. The focus of the paper is to study the feasibility and stability of such schemes when explicit feasibility domains of the decentralized RHC are available.

II. PROBLEM FORMULATION

Consider a set of $N_v$ linear decoupled dynamical systems, the $i$-th system being described by the discrete-time time-invariant state equation:

$$x_{k+1}^{i} = f_i(x_k^i, u_k^i)$$ (1)

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

$$x_k^i \in \mathcal{X}_i, \quad u_k^i \in \mathcal{U}_i, \quad k \geq 0$$ (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 $\tilde{x}_k \in R^{N_v \times n_i}$ and $\tilde{u}_k \in R^{N_v \times m_i}$ be the vectors which collect the states and inputs of the team system at time $k$, i.e. $	ilde{x}_k = [x_1^1, \ldots, x_{N_v}^1, \ldots, x_1^{N_v}, \ldots, x_{N_v}^{N_v}]$, with

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

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

So far the 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. $(i, j) \in A \Rightarrow (j, i) \in A$. Before defining the optimal control problem, we need to define a graph

$$G = (V, A)$$ (4)

where $V$ is the set of nodes $V = \{1, \ldots, N_v\}$ and $A \subseteq V \times V$ the sets of arcs $(i, j)$ with $i \in V$, $j \in V$.

Once the graph structure has been fixed, the optimization problem is formulated as follows. Denote with $\tilde{x}^i$ the states of all neighboring systems of the $i$-th system, i.e. $\tilde{x}^i = \{x^j \in R^{n_j} | (j, i) \in A\}$, $\tilde{x} \in R^{n_i}$ with $\tilde{x}^i = \sum_{j \left( j \in A \right)} n_j$. Analogously, $\tilde{u}^i \in R^{m_i}$ denotes the inputs to all the neighboring systems of the $i$-th system. Let

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

define the interconnection constraints between the $i$-th and the $j$-th systems, with $g^i : R^{n_i} \times R^{n_j} \to R^{n_{i,j}}$. We will often use the following shorter form of the interconnection constraints defined between the $i$-th system and all its neighbors:

$$g^i(x^i, \tilde{x}^i) \leq 0$$ (6)

with $g^i : R^{n_i} \times R^{n_{i}} \to R^{n_{i,i}}$.

Consider the following cost

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

where $l^i : R^{n_i} \times R^{m_i} \times R^{n_{i}} \times R^{m_{i}} \to R$ is the cost associated to the $i$-th system and is a function only of its states and the states of its neighbor nodes.

$$l^i(x^i, u^i, \tilde{x}^i, \tilde{u}^i) = \sum_{(i, j) \in A} l^{i,j}(x^i, u^i, x^j, u^j) + \sum_{(q, r) \in A, (i, q) \in A, (i, r) \in A} l^{q,r}(x^q, u^q, x^r, u^r)$$ (8)

where $l^{i,j} : R^{n_i} \times R^{m_i} \times R^{n_j} \times R^{m_j} \to R$ is the cost function involving two adjacent nodes. Assume that $l$ is a positive convex function and that $l^i(x^i, u^i, x^i, u^i) = 0$ and consider the infinite time optimal control problem

$$\tilde{J}_o(\tilde{x}) \triangleq \min_{\{\tilde{x}_0, \tilde{x}_1, \ldots\}} \sum_{k=0}^{\infty} l(\tilde{x}_k, \tilde{u}_k)$$ (9)

subject to

$$\begin{cases}
  x_{k+1}^i = f^i(x_k^i, u_k^i), & i = 1, \ldots, N_v, \quad k \geq 0 \\
  g^{i,j}(x_k^i, x_k^j) \leq 0, & i = 1, \ldots, N_v, \quad k \geq 0, \quad (i, j) \in A \\
  x_k^i \in \mathcal{X}_i, \quad u_k^i \in \mathcal{U}_i, & i = 1, \ldots, N_v, \quad k \geq 0 \\
  \tilde{x}_0 = \tilde{x}
\end{cases}$$ (10)
For all $\tilde{x} \in \mathbb{R}^{N_v \times n'}$, if problem (10) is feasible, then the optimal input $\tilde{u}^{*}_{0:t} \rightarrow \tilde{u}_{1:t} \ldots$ will drive the $N_v$ systems to their equilibrium points $x^*_i$ while satisfying state, input and interconnection constraints.

**Remark 1:** Throughout the paper we assume that a solution to problem (10) exists and it generates a feasible and stable trajectory for the team system. Our assumption is not restrictive. If there is no infinite time centralized optimal control problem fulfilling the constraints, then there is no reason to look for a decentralized receding horizon controller with the same properties.

**Remark 2:** Since we assumed that the graph is undirected, there will be redundant constraints in problem (10). Note the form of constraints (6) is rather general and it will include the case when only partial information about states of neighboring nodes is involved.

With the exception of a few cases, solving an infinite horizon optimal control problem is computationally prohibitive. An infinite horizon controller can be designed by repeatedly solving finite time optimal control problems in a receding horizon fashion as described next. At each sampling time, starting at the current state, an open-loop optimal control problem is solved over a finite horizon. The optimal command signal is applied to the process only during the following sampling interval. At the next time step a new optimal control problem based on new measurements of the state is solved over a shifted horizon. The resultant controller is often referred to as Receding Horizon Controller (RHC). Assume at time $t$ the current state $\tilde{x}_t$ to be available. Consider the following constrained finite time optimal control problem

$$J^*_N(\tilde{x}_1) \doteq \min \left\{ \sum_{k=0}^{N-1} l(\tilde{x}_{k,t}^i, \tilde{u}_{k,t}) + l_N(\tilde{x}_{N,t}) \right\} \quad (11a)$$

subject to

$$\begin{align*}
x^{i}_{k+1,t} &= f^i(x^{i}_{k,t}, u^{i}_{k,t}), \quad i = 1, \ldots, N_v, \quad k \geq 0 \\
g^{ij}(x^{i}_{k,t}, x^{j}_{k,t}) &\leq 0, \quad i = 1, \ldots, N_v, \quad (i,j) \in A, \\
x^{i}_{k,t} &\in \mathcal{X}^i, \quad u^{i}_{k,t} \in \mathcal{U}^i, \quad i = 1, \ldots, N_v, \quad k = 1, \ldots, N-1 \\
\tilde{x}_{N,t} &\in \mathcal{X}_f, \\
\tilde{x}_{0,t} &= \tilde{x}_t
\end{align*} \quad (11b)$$

where $N$ is the prediction horizon, $\mathcal{X}_f \subseteq \mathbb{R}^{N_v \times n'}$ is a terminal region, $l_N$ is the cost on the terminal state. In (11) we denote with $U_t^i \doteq [\tilde{u}^i_{0:t}, \ldots, \tilde{u}^i_{N-1:t}]^T \in \mathbb{R}^s$, $s \doteq N_v \times mN$ the optimization vector, $x^{i}_{k,t}$ denotes the state vector of the $i$-th node predicted at time $t+k$ obtained by starting from the state $x^{i}_{0,t}$ and applying to system (1) the input sequence $u^{i}_{0:t}, \ldots, u^{i}_{k-1:t}$. The tilded vectors will denote the prediction vectors associated to the team system.

Let $U^*_t = \{\tilde{u}^*_{0:t}, \ldots, \tilde{u}^*_{N-1:t}\}$ be the optimal solution of (11) at time $t$ and $J^*(\tilde{x}_t)$ the corresponding value function. Then, the first sample of $U^*_t$ is applied to the team system (3)

$$\tilde{u}_t = \tilde{u}^*_{0:t}. \quad (12)$$

The optimization (11) is repeated at time $t+1$, based on the new state $x_{t+1}$.

It is well known that stability is not ensured by the RHC law (11)-(12). Usually the terminal cost $l_N$ and the terminal constraint set $\mathcal{X}_f$ are chosen to ensure closed-loop stability. A treatment of sufficient stability conditions goes beyond the scope of this work and can be found in the surveys [9], [10]. We assume that the reader is familiar with the basic concept of RHC and its main issues, we refer to [9] for a comprehensive treatment of the topic. In this paper we will assume that terminal cost $l_N$ and the terminal constraint set $\mathcal{X}_f$ can be appropriately chosen in order to ensure the stability of the closed-loop system.

In general, the optimal input $u^i_t$ to the $i$-th system computed by solving (11) at time $t$, will be a function of the overall state information $\tilde{x}_t$. In [1] we have described a procedure to decompose problem (11) into smaller sub-problems whose independent computation can be distributed over the graph nodes.

In [1] we have proposed a decentralized control scheme where problem (11) is decomposed into $N_v$ finite time optimal control problems, each one associated to a different node as detailed next. Each node knows 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. 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 node $i$ will be implemented at the $i$-th node, where it was computed. The $i$-th subproblem will be a function of the states of the $i$-th node and the states of its neighbors. The solution of the $i$-th subproblem will yield a control policy for the $i$-th node of the form $u^i_t = f^i(x^i_t, \tilde{x}^i_t)$. However, the study in [1] does not guarantee constraint fulfillment.

In the next section we analyze decentralized RHC design when the nodes of the graph $\mathcal{G}$ have priorities assigned to them. We first decompose the original $\mathcal{G}$ into overlapping subgraphs with different hierarchy levels (Section III). Then, we solve the problem for each subgraph independently (Section IV) assuming that nodes with high priorities compute control laws for nodes with lower priorities. Finally, in Section V we propose a class of hierarchical RHC control schemes that guarantee stability and feasibility.

### III. Hierarchical Decomposition

Consider the nodes of the graph $\mathcal{G}$ of the original centralized problem description and divide it into $n_g$ intersecting subgraphs $\mathcal{G}_i = \{A_i, V_i\}$, with $A = \bigcup_{i=1}^{n_g} A_i$, $V = \bigcup_{i=1}^{n_g} V_i$ and $\forall i \exists j \ V_i \cap V_j \neq \emptyset$. We assign to each subgraph $G_i$ a certain priority $k$, $p(G_i) = k$. The decomposition into subgraphs follows a hierarchical scheme where $G_i$ has the
The nodes of each subgraph $G_j$ are partitioned into two groups: the group of predecessor nodes $V_j^P$ and the group of successor nodes $V_j^S, V_j = \{V_j^P, V_j^S\}$. The decomposition has to satisfy the following property: for a given subgraph $G_i$, with $i \geq 2$, the predecessor group composed of nodes $V_i^P$ is a subset of the group of successor nodes of $G_j$, i.e., $\forall j < i | V_i^P \subseteq V_j^S$. We will denote by $Pr$ the predecessor function, i.e., the function that associates to each subgraph with index $j$, the index $i$ of the subgraph which contains the predecessor nodes of $G_j$, i.e., $i = Pr(j) \iff x \in V_j^P \rightarrow x \in V_i^S$. The predecessor nodes of $G_i$ are those that are not elements of any other subgraph, i.e., $i \in V_i^P \iff i \notin V_j$ for $j = 2, \ldots, n_g$.

Figures 1-2 depict examples for both valid and invalid decompositions. Figure 1(a) shows a decomposition into three subgraphs $G = \bigcup_{i=1}^{3} G_i$. The definition of predecessor and successor nodes are the following:

- $G_1^P = \{0\}$, $G_1^S = \{1, 2, 3\}$
- $G_2^P = \{2\}$, $G_2^S = \{4, 5\}$
- $G_3^P = \{5\}$, $G_3^S = \{6\}$

Similarly, for the decomposition in Fig. 1(b) we have

- $G_1^P = \{0\}$, $G_1^S = \{1, 2\}$
- $G_2^P = \{2\}$, $G_2^S = \{3\}$
- $G_3^P = \{2\}$, $G_3^S = \{4\}$

According to the notation used in the previous sections we use the symbol $n^{i,P,S}$ to identify the number of states of all the nodes of the subgraph $G_i$, $n^{i,S}$ for the number of states of all the successor nodes of subgraph $G_i$, $n^{i,P}$ for the number of states of all the predecessor nodes of subgraph $G_i$. We denote by $\hat{x}^{i,P,S}_k \in \mathbb{R}^{n^{i,P,S}}$ the states of the nodes of graph $G_i$ at time $k$, $\hat{x}^{i,P}_k \in \mathbb{R}^{n^{i,P}}$ and $\hat{x}^{i,S}_k \in \mathbb{R}^{n^{i,S}}$ are predecessor and successors states of the nodes in $G_i$ at time $k$, respectively.

IV. SIMPLE HIERARCHICAL SCHEME ASSUMING COMMUNICATION DELAYS

Consider the systems (1) and the interconnection graph $G$ decomposed as discussed in Section III. Consider the following finite time optimal control problem for subgraph $G_v$.

$$\begin{align*}
(P_v): \quad & J_N^v(x_0^{v,S}, x_1^{v,P}) = \min_{U_{t}^{v,S}, U_{t}^{v,P}} \sum_{t=0}^{N-1} l^{v,S,P}(x_t^{v,S}, u_t^{v,S}, x_{t+1}^{v,P}, u_{t+1}^{v,P}) \quad (13a) \\
\text{subject to} & \begin{cases} 
\forall i, x_{k+1,t}^{i} = f(x_{k,t}^{i}, u_{k,t}^{i}), \\
\forall k, \forall i, u_{k,t}^{i} \in U^i, \\
\forall k, \forall i, \forall j, g^{i,j}(x_{k,t}^{i}, x_{k,t}^{j}) \leq 0, \\
\forall k, \forall i, \forall j, i \in V_v, j \in V_v, \\
x_{N,t}^{i} = x_{0,t}^{i}, \forall i \in V_v 
\end{cases} 
\quad (13b)
\end{align*}$$

where $\hat{U}_{t}^{v,S} \triangleq \{u_{0,t}^{i}, \ldots, u_{t,N-1}^{i} | i \in V_v^S\}$, $\hat{U}_{t}^{v,P} \triangleq \{u_{0,t}^{i}, \ldots, u_{N-1,t}^{i} | i \in V_v^P\}$ denotes the optimization vectors, i.e., the inputs to all the nodes of the subgraph $G_v$ grouped into successors and predecessors. Analogously, we define $\hat{U}_{t}^{v,S} \triangleq \{u_{0,t}^{i}, \ldots, u_{t,N-1}^{i} | i \in V_v^S\}$, $\hat{U}_{t}^{v,P} \triangleq \{u_{0,t}^{i}, \ldots, u_{N-1,t}^{i} | i \in V_v^P\}$, $\hat{X}_{t}^{i} \triangleq \{x_{0,t}^{i}, \ldots, x_{t,N-1}^{i}\}$, where $x_{k,t}^{i}$ denotes the state vector of the $i$-th node predicted at time $t + k$ obtained by starting from the state $x_{0,t}^{i}$ and applying to the $i$-th system (1) the input sequence $u_{0,t}^{i}, \ldots, u_{t,k-1}^{i}$. $X_{t}^{i} \subseteq \mathbb{R}^{n_i}$ is a terminal constraint for the $i$-th node. The
cost $l^{v,S,P}$ is defined as

$$l^{v,S,P}(x_{k,t}^{v}, u_{k,t}^{v}, x_{k,t}^{P}, u_{k,t}^{P}) = \sum_{(i,j) \in A_v} l^{i,j}(x^i, u^i, x^j, u^j)$$

(14)

Note that the summation is done only over the nodes and edges contained in the subgraph $G_v$.

We propose the following distributed strategy. Consider a subgraph $G_v$ and the associated problem $P_v$ (13). The optimal control sequence $\hat{U}_t^{v,P}$ for the predecessor nodes in group $v$ are also calculated as solutions for the successor nodes in the preceding group $Pr(v)$. Let us modify the problem of group $v$ by formulating and solving a finite time optimal control problem similar to (13) but constrained to use the control sequence calculated for the predecessor nodes by the preceding group. The only degree of freedom left is to obtain control sequences $\hat{U}_t^{v,S}$ for the successor nodes. This means that nodes in $V_v^{P}$ will be implementing a control sequence $\hat{U}_t^{v,P}$ received from the preceding group $\hat{U}_t^{Pr(v),S}$ and the successor nodes will be implementing the optimal control solutions $\hat{U}_t^{v,S}$ calculated by group $v$ assuming a one time step communication delay. These optimal control solutions for the successor nodes in $G_v$ have to respect predecessor-successor constraints represented by $g_{k}^{i,j}$ in (13b). This strategy can be summarized by the following algorithm.

Algorithm 1: Algorithm for propagating RHC solutions

1) For all $t \geq 0$
2) For all $v = 1, \ldots, n_g$
   a) Measure the states of all the nodes in the group $G_v$.
   b) Solve $P_1$ if $v = 1$, otherwise if $v > 1$ augment and solve problem $P_v$ (13) with the following constraints

$$\hat{U}_k^{v,P} = \hat{U}_k^{Pr(v),S}, k = 1, \ldots, N - 1.$$  (15)

c) Transmit and implement the solution $u_{0,t}^v$ on the $i$-th node for all $i \in V_v^{S}$.

Figure 3 illustrates the propagation of the solution in a simple case where the nodes are connected as a string, following one after another.
V. HIERARCHICAL DECOMPOSITION SCHEME WITH STABILITY AND FEASIBILITY GUARANTEES

The scheme presented in the previous section does not guarantee constraint fulfillment. In this section, we propose a decentralized scheme for which feasibility at time zero guarantees feasibility at all time instants $t > 0$ as well as stability. The main algorithm is described next.

**Algorithm 2:** Feasible Set Projection Algorithm

1) Consider the team system (3) and the interconnection graph $G$ (4) decomposed as discussed in Section III.
2) Consider problem $P_1$ in (13) with $X^i_1 = x^i_1, \forall i \in V_1$ and compute the set $X^{i,1,p,s} \subseteq \mathbb{R}^{n^{1,p,s}}$ of feasible initial predecessor and successor states $x^{1,p}, x^{1,s}$ for problem $P_1$.
3) Compute $X^{1,s}$ as the projection of the set $X^{i,1,p,s}$ on the successor space $\mathbb{R}^{n^{1,s}}, \text{i.e., } X^{1,s} = \{x^{1,s} \in \mathbb{R}^{n^{1,s}} : \exists x^{1,p} \in \mathbb{R}^{n^{1,p}}, (x^{1,s}, x^{1,p}) \in X^{i,1,p,s}\}$.
4) For $v = 2, \ldots, n_g$,
   a) Define a new RHC problem $P_v$ by augmenting problem $P_v$ (13) with the constraints
      
      $g^{i,j}(x^{i}_{1}, x^{j}_{1}) \leq 0,$
      $k = 1, \ldots, N - 1, (i,j) \in A_v,$
      \(\forall x^{v,p} \in \mathcal{X}^{v,p,v}\).
      
   b) Compute $X^{v,s}$ as the projection of the set $X^{v,p,s}$ on the successor space $\mathbb{R}^{n^{v,s}}$.

The algorithm presented above uses two main concepts. First, it requires the computation of the feasible domains of the RHC problems $P_v$ in an hierarchical increasing order. Second, it transforms the original RHC problems into new problems $P_v$ where constraints between predecessor and successors nodes are robustly enforced for all the successor nodes belonging to the feasibility domains of the hierarchically preceding subproblems. The RHC control policy for the team system is defined as follows.

**Algorithm 3:** RHC Control Policy

1) For all $t \geq 0$
2) For all $v = 1, \ldots, n_g$
   a) Measure the states of all the nodes in the group $G_v$.
   b) Solve $P_v$ if $v > 1$ or $P_1$ if $v = 1$.
   c) For all $i \in V^G_i$, implement $u^{i,v}_{t}$ on the $i$-th node.

The following theorem can be stated on the feasibility of the scheme presented above.

**Theorem 1:** Consider the team system (3), the equilibrium pair $(\bar{x}_e, \bar{u}_e)$, and the constraints (2). Assume a given interconnection graph (4) and the interconnection constraints (5) using the RHC control policy described in Algorithm 3, where the subproblems $P_v$ have been defined in Algorithm 2. If problems $P_1$ and $P_v (v = 2, \ldots, n_g)$ are feasible at time $t = 0$, then the RHC policy described in Algorithm 3 stabilizes the team system (3), in that

\[
\lim_{t \to \infty} \bar{x}(t) = \bar{x}_e \quad \text{and} \quad \lim_{t \to \infty} \bar{u}(t) = \bar{u}_e
\]

while fulfilling the state, input and interconnection constraints.

**Proof:** The proof follows by standard Lyapunov arguments and is omitted in this version of the paper.

**Remark 2:** The proof of Theorem 1 can be extended when terminal invariant sets and control Lyapunov functions are used instead of terminal point set constraint.

**Remark 4:** In Algorithm 2 the step of computing the feasibility domains of problem $P_v$ is not a trivial one. Algorithms are available when problems $P_v$ can be casted as linear, quadratic, mixed-integer linear and mixed-integer quadratic programs [11].

A. Graphical Illustration of the Method

In this section we describe the approach presented in the previous section through a simple example. Figure 5 shows two subgraphs $G_1$ and $G_2$, with two nodes in each of them. Node 2 is the successor of group $G_1$ and the predecessor of group $G_2$. $G_1$ has a higher priority than $G_2$. Denote by $x_1 \in \mathbb{R}, x_2 \in \mathbb{R}, x_3 \in \mathbb{R}$ the states of node 1, 2 and 3 respectively.

The RHC problem associated with $G_1$ and $G_2$ will be denoted by $P_1$ and $P_2$, respectively. The state feedback solution $P_1$ is computed first and it is a function of $x_1$ and $x_2$. We denote by $u_1$ and $u_2$ the state feedback solution of problem $P_1$ for node 1 and 2, respectively: $u_1 = f_1(x_1, x_2), u_2 = f_2(x_1, x_2)$. $X^{1,p,s} \subseteq \mathbb{R}^2$ is the domain of the functions $f_1$ and $f_2$. When solving problem $P_2$, the knowledge of $X^{1,p,s}$ allow us to guarantee that the inputs computed for $x_3$ will be feasible for the closed-loop behavior of node 2. We follow the steps described in Algorithm 2. We compute the projection of $X^{1,p,s}$ on the successor space, and obtain a set in 1 dimension denoted by $X^{1,s}$. Consider now problem $P_2$ and the interconnection constraints

\[
g^{2,3}(x^2_k, x^3_k) \leq 0, k \geq 0 \tag{17}
\]

We satisfy constraint (17) for all the states $x_2$ which are feasible at the higher level. That is, we construct a new problem $P_2$ where constraint (17) is substituted with

\[
g^{2,3}(x^2_k, x^3_k) \leq 0, \forall x^2_k \in X^{1,s}, k \geq 0 \tag{18}
\]

Problem $P_2$ is still function of the state $x_2$ because it enters the cost function.
Clearly the constraint (18) can be infeasible and such approach might be conservative. However the knowledge of the sets $X_{1,S}$ allows us to reduce the degree of conservativeness with respect to other approaches which consider all possible behavior of the states $x_2$. Also, note that only the knowledge of the feasible sets is required and not of the state-feedback laws $f_1$ and $f_2$.

Consider the example above and the three cases depicted in Figure 4. We assume to have solved $P_1$ and to have computed $X_{1,S}$ (the bold lines in the $x_2$ space). The shaded areas depict three possible feasible sets described by the interconnection constraints $g^{2,3}(x^2, x^3) \leq 0$. A quick glance at Figure 4 tells that the conservativeness of the proposed method depends on the shape of the feasible region described through the interconnection constraints.

In the first case, problem $P_2$ is always feasible, in the second case $P_2$ is infeasible and in the third case $P_2$ is feasible over two disconnected sets.

A possible way of reducing conservativeness is to introduce communication of the admissible feasible regions between the hierarchical levels, the work in [12] is a preliminary study in this direction. We are currently looking at different protocols for exchanging bounds on the feasible domains.

VI. EXAMPLES

In the final version of the paper we will introduce two examples where we compare the simple scheme without guarantees presented in Section IV with the scheme presented in Section V for the formation flying scenario [1]. The technique presented in this paper can be very effective for such applications. In fact, even if the feasible domains $X^{v,P,S}$ are non-convex in general (because of the non-convexity of the collision avoidance constraints), their projections $X^{v,S}$ on the successors are convex. Thus, robust constraint fulfillment can be achieved without additional computational effort.

REFERENCES