

# Receding Horizon Control of Multi-Vehicle Formations: A Distributed Implementation<sup>1</sup>

William B. Dunbar<sup>†</sup>, Richard M. Murray

Control and Dynamical Systems 107-81  
California Institute of Technology  
Pasadena, CA 91125

<sup>†</sup>Corresponding Author: [dunbar@cds.caltech.edu](mailto:dunbar@cds.caltech.edu)

**Abstract:** *We consider the control of dynamically decoupled subsystems whose state vectors are coupled in the cost function of a finite horizon optimal control problem. For a given cost structure, we generate distributed optimal control problems for each subsystem and establish that a distributed receding horizon implementation is asymptotically stabilizing. The communication requirements at each receding horizon update include the exchange of the previous optimal control trajectory between subsystems with coupling in the cost function. The key requirements for stability are that each distributed optimal control not deviate too far from the previous one, and that the receding horizon updates happen sufficiently fast. A simulation example of multi-vehicle formation stabilization is provided.*

**Keywords:** cooperative control, predictive control for linear systems, constrained control.

## 1 Introduction

We are interested in the control of a set of dynamically decoupled subsystems that are required to perform a cooperative task. An example of such a situation is a group of vehicles cooperatively converging to a desired formation, as explored by Dunbar and Murray [7] and Ren and Beard [15]. In [7], the cooperative objective is accommodated by an appropriate cost function in an optimal control problem that is implemented using receding horizon control. A recent survey of receding horizon control, or model predictive control, is given by Mayne *et al* [12]. Henceforth, we refer to each subsystem as an *agent* and any two agents that are cooperating are referred to as *neighbors*. In the formulation here, neighbors have a term coupling their states in a single cost function of an optimal control problem. Generally, receding horizon control is warranted when the individual subsystems are required to satisfy state and control constraints, as in the case of vehicles.

A drawback of the receding horizon control approach to

our problem is that currently only a centralized solution and implementation can guarantee asymptotic stability theoretically. A distributed solution to the problem is desirable, for autonomy of the individual subsystems and for potential scalability and improved tractability of the approach. Previous work on distributed receding horizon control include studies by Jia and Krogh [9], Motee and Sanyar-Rodsari [14] and Acar [1]. In all of these papers, the cost is separable while the dynamics are coupled. Further, state and input constraints are not included, aside from a stability constraint in [9]. In another work, Jia and Krogh [10] solve a min-max problem for each agent, where again coupling comes in the dynamics and the neighboring agent states are treated as bounded disturbances. Stability is obtained by contracting each agents state at every sample period, until the objective set is reached. As such, stability does not depend on information updates with neighboring agents, although such updates may improve performance. More recently, Keviczky *et al* [11] have formulated a distributed model predictive scheme where each agent optimizes locally for itself and for every neighbor at each update. By this formulation, feasibility becomes difficult to ensure when coupling constraints are present, and no proof of stability is provided.

The problem formulation and stability results here are based somewhat on those given by Chen and Allgöwer [4]. For a particular structure in the centralized cost and by appropriate decomposition in defining the distributed integrated costs, asymptotic stability is proven under stated conditions. Key requirements for stability are that the receding horizon updates happen sufficiently fast, and each distributed optimal control trajectory is constrained to not deviate too far from the remainder of the previous optimal trajectory. No communication is required between agents while the distributed optimal control problems are being solved.

Section 2 defines an optimal control problem, with an integrated cost function relevant for multi-vehicle formation stabilization, and defines the centralized receding horizon control law. Section 3 defines the distributed receding horizon implementation and the results used to prove asymp-

<sup>1</sup>Partial support for this work was provided by the DARPA SEC program under grant number F33615-98-C-3613 and by AFOSR grant number F49620-01-1-0361.

otic stability. Simulation results of a multi-vehicle formation are then given in Section 4. Finally, Section 5 gives conclusions, extensions and applications of the theory.

## 2 Receding Horizon Control

In this section, we pose a single finite horizon optimal control problem relevant for multi-vehicle formation stabilization and define the centralized receding horizon control law. The coupling of states of the vehicles occurs in the cost function. To simplify analysis, we consider only linear homogeneous dynamics here. Nonlinear heterogeneous subsystem dynamics, with a more generic quadratic coupling cost function and coupling state constraints, is treated elsewhere [6].

We wish to stabilize a group of agents toward a common objective in a cooperative way using receding horizon control. For each agent  $i \in \{1, \dots, N_a\}$ , the state and control are  $z_i(t) = (q_i(t), \dot{q}_i(t)) \in \mathbb{R}^{2n}$  and  $u_i(t) \in \mathbb{R}^n$ , respectively, and the decoupled dynamics are given by

$$\ddot{q}_i(t) = u_i(t), \quad i = 1, \dots, N_a. \quad (1)$$

Each agent  $i$  is also subject to the decoupled input and state constraints

$$u_i(t) \in \mathcal{U}, \quad z_i(t) \in \mathcal{Z}, \quad t \geq 0.$$

The set  $\mathcal{Z}^N$  is the  $N$ -times Cartesian product  $\mathcal{Z} \times \dots \times \mathcal{Z}$ . The concatenated vectors are denoted  $q = (q_1, \dots, q_{N_a})$ ,  $\dot{q} = (\dot{q}_1, \dots, \dot{q}_{N_a})$ ,  $z = (z_1, \dots, z_{N_a}) \in \mathcal{Z}^{N_a}$  and  $u = (u_1, \dots, u_{N_a}) \in \mathcal{U}^{N_a}$ .

**Definition 1** *The control objective is to cooperatively asymptotically stabilize all agents to  $z^c = (z_1^c, \dots, z_{N_a}^c) \in \mathcal{Z}^{N_a}$ , an equilibrium point of equation (1), with equilibrium control equal to zero.*

The position values at  $z^c$  are denoted  $q^c = (q_1^c, \dots, q_{N_a}^c)$ , and the equilibrium velocity is clearly zero from equation (1). The cooperation is achieved by the minimization of the integrated cost function

$$L(z, u) = \sum_{(i,j) \in \mathcal{E}_0} \omega \|q_i - q_j + d_{ij}\|^2 + \omega \|q_{\Sigma} - q_d\|^2 + \nu \|\dot{q}\|^2 + \mu \|u\|^2,$$

with positive weighting constants  $\omega, \nu$  and  $\mu$ . We refer to the term  $\omega \|q_{\Sigma} - q_d\|^2$  as the *tracking cost*, where

$$q_{\Sigma} = (q_1 + q_2 + q_3)/3, \quad q_d = (q_1^c + q_2^c + q_3^c)/3.$$

Every desired relative vectors  $d_{ij}$  between neighboring agents  $i$  and  $j$  satisfies  $q_i^c + d_{ij} = q_j^c$ . The set  $\mathcal{E}_0$  denotes the set of edges  $(i, j)$  in the formation graph defined in [8]. Assuming the formation graph is connected, which means that the position state  $q_k$  of every agent  $k \in \{1, \dots, N_a\}$  occurs

in at least one of the terms  $\|q_i - q_j + d_{ij}\|^2$  in  $L(z, u)$ , we can write the integrated cost equivalently as

$$L(z, u) = \|z - z^c\|_Q^2 + \mu \|u\|^2, \quad (2)$$

where  $Q = Q^T > 0$  [Proposition 1 in [8]].

**Assumption 1** *The following holds: (i) The set  $\mathcal{U} \subset \mathbb{R}^n$  is compact, convex and contains the origin in its interior, and the set  $\mathcal{Z} \subseteq \mathbb{R}^{2n}$  is convex, connected and contains  $z_i^c$  in its interior, for every  $i \in \{1, \dots, N_a\}$ ; (ii) All states are measurable and computational time is negligible compared to the evolution of the closed-loop dynamics.*

We do not incorporate any type of collision avoidance in this paper, although this can be done by an appropriate choice of cost function or constraints. At any time  $t$ , given  $z(t)$  and fixed horizon time  $T$ , the centralized open-loop optimal control problem is

**Problem 1** *Find*

$$J^*(z(t), T) = \min_{u(\cdot)} \int_t^{t+T} L(z(\tau), u(\tau)) d\tau + G(z(t+T)),$$

*subject to*

$$\ddot{q}(s) = u(s), \quad u(s) \in \mathcal{U}^{N_a}, \quad z(s) \in \mathcal{Z}^{N_a}, \quad \forall s \in [t, t+T], \\ z(t+T; z(t)) \in \Omega(\alpha),$$

where  $G(z) = \|z - z^c\|_P^2$ ,  $P = P^T > 0$  and

$$\Omega(\alpha) := \{z \in \mathbb{R}^{2nN_a} : G(z) \leq \alpha, \alpha \geq 0\}.$$

Let the first optimal control problem be initialized at some time  $t_0 \in \mathbb{R}$  and let  $\delta$  denote the *receding horizon update period*. The closed-loop system is

$$\ddot{q}(\tau) = u_{\text{cent}}^*(\tau), \quad \tau \geq t_0, \quad (3)$$

where the *centralized receding horizon control law* is

$$u_{\text{cent}}^*(\tau) = u_{\text{cent}}^*(\tau; z(t)), \quad \tau \in [t, t+\delta], \quad 0 < \delta < T,$$

and  $u_{\text{cent}}^*(s; z(t))$ ,  $s \in [t, t+T]$ , is the optimal open-loop solution (assumed to exist) to Problem 1 with initial state  $z(t)$ . The receding horizon control law is defined for all  $t \geq t_0$  by applying the open-loop optimal solution until each new initial state update  $z(t) \leftarrow z(t+\delta)$  is available. The notation above shows the implicit dependence of the optimal open-loop control  $u_{\text{cent}}^*(\cdot)$  on the initial state  $z(t)$  through the optimal control problem. Receding horizon control is not optimal and not necessarily stabilizing. Sufficient conditions for asymptotic stability are established in [4] and involve appropriate choice of terminal weighting  $P$  and terminal constraint set parameter  $\alpha$ . In the next section,  $N_a$  optimal control problems are defined for a distributed receding horizon implementation. Under stated conditions on the update parameter  $\delta$ , the implementation is asymptotically stabilizing.

### 3 Distributed Receding Horizon Control

In this section, a distributed receding horizon control law is defined. We first introduce some notation and define  $N_a$  separate optimal control problems, that are solved and implemented in a distributed receding horizon fashion. Next, we analyze the stability of the closed-loop system.

#### 3.1 Distributed Optimal Control Problems

In the centralized integrated cost, the non-separable terms  $\|q_i - q_j + d_{ij}\|^2$ , for all  $(i, j) \in \mathcal{E}_0$ , as well as the tracking term  $\|q_\Sigma - q_d\|^2$ , couple the states of neighboring agents. The set of neighbors of each agent  $i$  is denoted  $\mathcal{N}_i$ . Let  $z_{-i} = (z_{j_1}, \dots, z_{j_{|\mathcal{N}_i|}})$  denote the vector of states of the neighbors of  $i$ , i.e.,  $j_k \in \mathcal{N}_i$ ,  $k = 1, \dots, |\mathcal{N}_i|$ , where the ordering of the states is arbitrary but fixed. Also, let  $u_{-i} = (u_{j_1}, \dots, u_{j_{|\mathcal{N}_i|}})$ , where the ordering is consistent with  $z_{-i}$ .

**Definition 2** *The distributed integrated cost in the optimal control problem for any agent  $i \in \{1, \dots, N_a\}$  is defined as  $L_i(z_i, z_{-i}, u_i) = L_i^z(z_i, z_{-i}) + \gamma\mu\|u_i\|^2$ , where  $\gamma > 1$  is constant and  $L_i^z(z_i, z_{-i})$  is*

$$\gamma \left[ \sum_{j \in \mathcal{N}_i} \left\{ \frac{\omega}{2} \|q_i - q_j + d_{ij}\|^2 \right\} + \nu \|\dot{q}_i\|^2 + L_d(i) \right],$$

$$\text{with } L^d(i) = \begin{cases} \frac{\omega}{3} \|q_\Sigma - q_d\|^2, & i = 1, 2, 3 \\ 0, & \text{otherwise.} \end{cases}$$

Thus,  $\sum_{i=1}^{N_a} L_i(z_i, z_{-i}, u_i) = \gamma L(z, u)$ .

Agents 1, 2 and 3 are neighbors by virtue of the tracking term; however, the summation over  $\mathcal{N}_i$  in  $L_i^z$  for  $i \in \{1, 2, 3\}$  is understood to include only the  $\|q_i - q_j + d_{ij}\|^2$  terms in  $\mathcal{E}_0$ , which may or may not exist between 1, 2 and 3. In the proof of stability, the key structure is that the distributed integrates costs sum to be the centralized cost multiplied by a constant larger than 1. For any other cost, if a similar decomposition structure can be chosen, and if the other stated assumptions hold, the stability results that follow still hold. Moreover, the stability results that follow do not depend on equal weighting of terms between neighboring agents. The weighting will of course affect the performance of the closed-loop system, so making the weights lop-sided would result in one agent reacting more to the term than the corresponding neighbor. In the limit that one agent takes all of a coupling term, the result corresponds to a directed graph, or leader-follower situation relative to that term.

**Definition 3 (Distributed Implementation Logic)** *At each update of the distributed receding horizon control laws, every agent: (1) senses its own current state and senses or receives the current state of its neighbors, and (2) computes the optimal control trajectory, comparing it to an **assumed***

*control trajectory and based on some **assumed** control trajectories for its neighbors. Prior to the next receding horizon update, every agent: (1) implements the current optimal control trajectory, (2) computes the next assumed control trajectory, to be used at the next update, and (3) transmits the assumed trajectory to all of its neighbors and receives the assumed control trajectories from each neighbor.*

Implicit in the procedure above is that the assumed control for each agent  $i$  is consistent in every optimization problem that it occurs, i.e., in the optimal control problem for agent  $i$  and for each neighbor  $j \in \mathcal{N}_i$ . Before defining the computation for the optimal and assumed control trajectories, we introduce some notation.

**Definition 4** *For every agent  $i = 1, \dots, N_a$ , we denote:*

$$u_i(\cdot) : \text{applied control}, \quad \hat{u}_i(\cdot) : \text{assumed control}.$$

*The applied control is being optimized and applied to the system. While being optimized, the applied control is also compared to the assumed control, which all neighbors assume  $i$  is employing over the interval.*

The state trajectories corresponding to the applied and assumed controls are denoted  $z_i(\cdot)$  and  $\hat{z}_i(\cdot)$ , respectively. For each agent  $i$ , given the current state  $z_j(t)$  and assumed control  $\hat{u}_j(s)$ ,  $s \in [t, t+T]$ , of any neighbor  $j \in \mathcal{N}_i$ , the assumed state trajectory  $\hat{z}_j(s)$ ,  $s \in [t, t+T]$ , is computed using the model for that agent. Consequently, the initial condition of every assumed state trajectory is equal to that of the actual state trajectory of the corresponding agent, i.e.,  $\hat{z}_i(t) = z_i(t)$ , for initial time  $t$ . To be consistent with the notation  $z_{-i}$ , let  $\hat{z}_{-i}(\cdot)$  and  $\hat{u}_{-i}(\cdot)$  be the vector of assumed neighbor states and controls, respectively, of agent  $i$ .

Denote the receding horizon update times as  $t_k = t_0 + \delta k$ , where  $k \in \mathbb{N} = \{0, 1, 2, \dots\}$ . Common to each problem, we are given the constant  $\gamma \in (1, \infty)$  from Definition 2, update parameter  $\delta \in (0, T)$  and fixed horizon time  $T$ . Conditions will be placed on the update parameter  $\delta$  in the next section to guarantee stability of the closed-loop system. The collection of distributed open-loop optimal control problems is now defined.

**Problem 2** *For every agent  $i \in \{1, \dots, N_a\}$  and at any update time  $t_k$ ,  $k \in \mathbb{N}$ , given initial conditions  $z_i(t_k)$ ,  $z_{-i}(t_k)$ , and assumed controls  $\hat{u}_i(s)$  and  $\hat{u}_{-i}(s)$ , for all  $s \in [t_k, t_k + T]$ , find  $J_i^*(z_i(t_k), z_{-i}(t_k), T)$ , equal to*

$$\min_{u_i(\cdot)} \int_{t_k}^{t_k+T} L_i(z_i(\tau), \hat{z}_{-i}(\tau), u_i(\tau)) d\tau + \gamma G_i(z_i(t_k + T)),$$

*subject to*

$$\left. \begin{aligned} \ddot{q}_i(s) &= u_i(s), \quad \ddot{q}_j(s) = \hat{u}_j(s), \quad \forall j \in \mathcal{N}_i, \\ u_i(s) &\in \mathcal{U}, \quad z_i(s) \in \mathcal{Z} \\ \|u_i(s) - \hat{u}_i(s)\| &\leq \delta^2 \kappa \end{aligned} \right\} s \in [t_k, t_k + T],$$

$$z_i(t_k + T) \in \Omega_i(\varepsilon_i), \quad (4)$$

given constant  $\kappa \in (0, \infty)$ ,  $G_i(z_i) = \|z_i - z_i^c\|_{P_i}^2$ ,  $P_i = P_i^T > 0$  and where

$$\Omega_i(\varepsilon_i) := \{z \in \mathbb{R}^{2n} : G_i(z) \leq \varepsilon_i, \varepsilon_i \geq 0\}.$$

As part of the optimal control problem, the applied control for  $i$  is constrained to be at most a distance of  $\delta^2\kappa$  from the assumed control. The assumed control for each agent, as well as each terminal cost weighting  $P_i$ , will be mathematically defined below. As discussed in [8], it may be desirable to compute and transmit the needed assumed states, rather than assumed controls, since that is what each distributed optimal control problem depends upon in this case. This would, for example, remove the need for the differential equation of each neighbor in each local optimization problem. The optimal solution to each distributed optimal control problem, assumed to exist, is denoted

$$u_{di}^*(\tau; z_i(t_k)), \tau \in [t_k, t_k + T].$$

The closed-loop system, for which stability is to be guaranteed, is

$$\ddot{q}(\tau) = u_{\text{dist}}^*(\tau), \quad \tau \geq t_0 \quad (5)$$

where the *distributed receding horizon control law* is

$$u_{\text{dist}}^*(\tau; z(t_k)) = (u_{d1}^*(\tau; z_1(t_k)), \dots, u_{dN_a}^*(\tau; z_{N_a}(t_k))),$$

for  $\tau \in [t_k, t_k + \delta]$ ,  $0 < \delta < T$  and  $k \in \mathbb{N}$ . As before, the receding horizon control law is updated when each new initial state update  $z(t_k) \leftarrow z(t_{k+1})$  is available. The optimal state for agent  $i$  is denoted  $z_{di}^*(\tau; z_i(t_k))$  and the concatenated vector of the distributed optimal states is denoted

$$z_{\text{dist}}^*(\tau; z(t_k)) = (z_{d1}^*(\tau; z_1(t_k)), \dots, z_{dN_a}^*(\tau; z_{N_a}(t_k))),$$

for all  $\tau \in [t_k, t_k + T]$ . Although we denote the optimal control for agent  $i$  as  $u_{di}^*(\tau; z_i(t_k))$ , it is understood that this control is implicitly dependent on the initial state  $z_i(t_k)$  and the initial states of the neighbors  $z_{-i}(t_k)$ .

**Assumption 2** *The following holds for every  $i = 1, \dots, N_a$ :*  
(i) *The positive constant  $\varepsilon_i > 0$  is chosen such that  $\Omega_i(\varepsilon_i) \subseteq \mathcal{Z}$  and such that for all  $z_i \in \Omega_i(\varepsilon_i)$ , there is an asymptotically stabilizing feedback  $u_i = K_i(z_i - z_i^c)$  that is feasible and the weighting matrix  $P_i = P_i^T > 0$  satisfies*

$$(A_i + B_i K_i)^T P_i + P_i (A_i + B_i K_i) = -(Q_i + \mu K_i^T K_i)$$

*Moreover,  $Q_i$  is chosen such that  $\widehat{Q} = \text{diag}(Q_1, \dots, Q_{N_a})$  satisfies  $\widehat{Q} \geq Q$ , where  $Q$  is defined in equation (2); (ii) *At any receding horizon update time, the collection of open-loop optimal control problems in Problem 2 are solved globally synchronously; (iii) *The communication of control trajectories between neighboring agents is lossless.***

The receding horizon control law is employed for all time after the initialization and the decoupled linear feedbacks

$K_i$  need not be employed, even after agent  $i$  enters  $\Omega_i(\varepsilon_i)$ . Due to condition (ii) above, the distributed receding horizon control laws are not technically decentralized, since a globally synchronous implementation requires centralized clock keeping [3]. However, a locally synchronous, and consequently decentralized, version is also currently being constructed [6]. Also, one choice for  $Q_i$  that would satisfy  $\widehat{Q} \geq Q$  is  $Q_i = \lambda_{\max}(Q)I_{(2n)}$ , where  $\lambda_{\max}(M)$  is the maximum eigenvalue of the square matrix  $M$ . We now define the initialization procedure for the distributed receding horizon control law, and the assumed control for each agent at each update time.

**Definition 5 (Initialization)** *Denote time  $t_{-1} = t_0 - \delta$ . Solve Problem 2 with initial state  $z(t_{-1})$ , setting  $\hat{u}_i(\tau) = 0$  for all  $\tau \in [t_{-1}, t_{-1} + T]$  and every  $i = 1, \dots, N_a$ , and also setting  $\kappa = +\infty$ . The optimal trajectories are denoted  $u_{di}^*(\tau; z_i(t_{-1}))$  and  $z_{di}^*(\tau; z_i(t_{-1}))$ , for every  $i = 1, \dots, N_a$ . The optimal control  $u_{\text{dist}}^*(\tau; z(t_{-1}))$  is applied for  $\tau \in [t_{-1}, t_0]$ .*

At initialization, the control deviation constraint is removed by setting  $\kappa$  to a large number. The assumed controls at initialization will naturally have an impact on closed-loop performance. If instead the centralized problem were solved at time  $t_{-1}$ , and the solution disseminated to the agents, the closed-loop performance may be closed to that of the centralized case. Also, initialization is considerably more complicated if there are coupling constraints between neighboring agents, as intimated in [11]. The formulation in [6] guarantees that if an initially feasible solution can be found with coupling constraints, with appropriate but not substantial added conservatism, subsequent feasibility is ensured.

**Definition 6 (Assumed Control)** *For each agent  $i = 1, \dots, N_a$  and for any  $k \in \mathbb{N}$ , the assumed control  $\hat{u}_i(\cdot) : [t_k, t_k + T] \rightarrow \mathcal{U}$  is defined as follows:*

$$\text{if } z(t_k) = z^c, \text{ then } \hat{u}_i(\cdot) = 0. \quad \text{Otherwise}$$

$$\hat{u}_i(\tau) = \begin{cases} u_{di}^*(\tau; z_i(t_{k-1})), & \tau \in [t_k, t_{k-1} + T] \\ K_i(z_i^k(\tau) - z_i^c), & \tau \in [t_{k-1} + T, t_k + T] \end{cases},$$

where  $z_i^k(s)$  is generated by the closed-loop solution to  $\ddot{q}_i^k(s) = K_i(z_i^k(s) - z_i^c)$ . The assumed control for agent  $i$  at initial time  $t_k$  is generated and transmitted to each neighbor  $j \in \mathcal{N}_i$  in the time window  $[t_{k-1}, t_k]$ .

To state Definition 6 in words, in Problem 2 every agent is assuming all neighbors will continue along their previous optimal path, finishing with the decoupled linear control laws defined in Assumption 2, *unless* the control objective is met at any finite update time after initialization. In the latter case, neighbors are assumed to do nothing, i.e., apply zero control. Notice that the communication of control trajectories between neighboring agents is not required to happen instantaneously, but over each receding horizon update time interval.

The test of whether  $z(t_k) = z^c$  in generating the assumed control is a centralized test. The reason for the test is its use in the proof of Proposition 3.1, stated in the next section. We note that the asymptotic stability result in the next section guarantees that only in the limit as  $t_k \rightarrow \infty$  do we have  $z(t_k) \rightarrow z^c$ . Practically then, one could assume  $z(t_k) \neq z^c$ , which is true for any finite  $k$  when  $z(t_{-1}) \neq z^c$ , and ignore the test completely. Also, if dual-mode receding horizon control is used, the test can be removed, since Proposition 3.1 is not used to prove asymptotic stability in that case. A dual-mode version is provided in [8].

If  $J_i^*(z_i(t_{-1}), z_{-i}(t_{-1}), T) = 0$  for any agent  $i$ , then it can be shown that  $z_i(t_{-1}) = z_i^c$  and  $z_j(t_{-1}) = z_j^c$ , for each neighbor  $j \in \mathcal{N}_i$ , is the unique feasible solution, i.e., the local objective has been met. However, even if  $J_i^*(z_i(t_{-1}), z_{-i}(t_{-1}), T) = 0$ , it may not remain zero for all  $k \in \mathbb{N}$ . An example is where  $i$  and all neighbors  $j \in \mathcal{N}_i$  are initialized meeting their objective, but some  $l \in \mathcal{N}_j$  has not met its objective. Thus, in the subsequent optimizations,  $j$  will react to  $l$ , followed by  $i$  reacting to  $j$ , since the coupling cost terms become nontrivial. Consequently, we can not guarantee that each distributed optimal value function  $J_i^*(z_i(t_k), z_{-i}(t_k), T)$  will decrease with each receding horizon update. Instead, we show in the next section that the *sum of the distributed optimal value functions* is a Lyapunov function that does decrease at each update, enabling a proof that the distributed receding horizon control laws *collectively* meet the control objective.

### 3.2 Stability Analysis

We now proceed with analyzing the distributed receding horizon control laws. The proofs of the results stated in this section are provided in [8], available on-line. At any time  $t_k$ ,  $k \in \mathbb{N}$ , the sum of the optimal distributed value functions is denoted as

$$J_{\Sigma}^*(z(t_k), T) = \sum_{i=1}^{N_a} J_i^*(z_i(t_k), z_{-i}(t_k), T).$$

For stability of the distributed receding horizon control laws, we investigate  $J_{\Sigma}^*(z(t_k), T)$  as a Lyapunov function. One has to be careful not to assume that  $J_{\Sigma}^*(z(t_k), T)$  is itself optimal. Problem 2 is *feasible* at time  $t_k$  if for every  $i = 1, \dots, N_a$ , there exists a control  $u_i(\cdot) : [t_k, t_k + T] \rightarrow \mathcal{U}$  such that all the constraints are satisfied and the value function  $J_i(z_i(t_k), z_{-i}(t_k), u_i(\cdot), T)$  is bounded. Let  $Z_{\Sigma} \subset \mathcal{Z}^{N_a}$  denote the *set of initial states for which Problem 2 is feasible at initialization* (time  $t = t_{-1}$ ), as defined in Definition 5.

**Lemma 1** *Under Assumptions 1 and 2,  $Z_{\Sigma}$  is a positively invariant set with respect to the closed-loop system (5) setting  $u_{di}^*(\cdot; z_i(t_k)) = \hat{u}_i(\cdot)$  for every  $i = 1, \dots, N_a$  and for  $k \in \mathbb{N}$ . Thus, feasibility at initialization implies subsequent feasibility.*

The proof follows immediately from Definitions 5 and 6. Note that the assumed control  $\hat{u}_i$  is exactly the feasible con-

trol trajectory used in Lemma 2 of [4] to show initial feasibility implies subsequent feasibility of the on-line optimization problem in the centralized case. Clearly,  $z^c$  is in the set  $Z_{\Sigma}$ .

Since we will be exploring the closed-loop behavior for initial states that start in  $Z_{\Sigma}$ , we can immediately infer that any closed-loop state trajectory will remain bounded. Specifically, if an initial state can be driven to the compact terminal constraint set in finite time using bounded control ( $\mathcal{U}$  is compact), then the optimal trajectory from that state will remain bounded. In the bounding argument for the proof of stability, we will make use of the notation

$$\|z_{di}^*(\tau; t_k) - z_i^c\| \leq R, \quad (6)$$

for all  $\tau \in [t_k, t_k + T]$ , any  $k \in \mathbb{N}$  and for all  $i = 1, \dots, N_a$ . Moreover, let  $U_{\max} > 0$  be a positive scalar denoting the maximum-norm value over all feasible controls.

**Assumption 3** *The optimal solution to Problem 2 exists and is numerically obtainable for any  $z(t_k) \in Z_{\Sigma}$ .*

**Lemma 2** *Under Assumptions 1-3,  $Z_{\Sigma}$  is a positively invariant set with respect to the closed-loop system (5). Thus, if  $z(t_{-1}) \in Z_{\Sigma}$ ,  $z_{dist}^*(\tau) \in Z_{\Sigma}$  for all  $\tau \geq t_{-1}$ .*

The next result says that the net objective of the distributed receding horizon control laws is consistent with the control objective.

**Proposition 3.1** *Under Assumptions 1-3, for a given fixed horizon time  $T > 0$  and at any time  $t_k$ ,  $k \in \mathbb{N}$ ,*

1.  $J_{\Sigma}^*(z(t_k), T) \geq 0$  for any  $z(t_k) \in Z_{\Sigma}$ , and  $J_{\Sigma}^*(z(t_k), T) = 0$  if and only if  $z(t_k) = z^c$ ,
2.  $J_{\Sigma}^*(z(t_k), T)$  is continuous at  $z(t_k) = z^c$ .

Our objective is to show the distributed receding horizon control law achieves the control objective for sufficiently small  $\delta$ . We begin with three lemmas that are used to bound the Lyapunov function candidate  $J_{\Sigma}^*(z(t_k), T)$ . The first lemma gives a bounding result on the decrease in  $J_{\Sigma}^*(\cdot, T)$  from one update to the next.

**Lemma 3** *Under Assumptions 1-3, for a given fixed horizon time  $T > 0$ , we have that*

$$\begin{aligned} & J_{\Sigma}^*(z(t_{k+1}), T) - J_{\Sigma}^*(z(t_k), T) \\ & \leq - \int_{t_k}^{t_k + \delta} \sum_{i=1}^{N_a} L_i^z(z_{di}^*(\tau; z_i(t_k)), \hat{z}_{-i}(\tau)) d\tau + \delta^2 \xi, \end{aligned}$$

for any  $\delta \in (0, T]$  and for any  $z(t_k) \in Z_{\Sigma}$ ,  $k \in \mathbb{N}$ , where

$$\xi = \gamma \omega \kappa T^3 (2|\mathcal{E}_0| + 2/3) [3R + \kappa T^4].$$

Ultimately, we want to show that  $J_{\Sigma}^*(\cdot, T)$  decreases from one update to the next along the actual closed-loop trajectories. The next two lemmas show that, for sufficiently small  $\delta$ , the bounding expression above can be bounded by a negative-definite function of the closed-loop trajectories.

**Lemma 4** *Under Assumptions 1-3, for any  $z(t_k) \in Z_{\Sigma}$ ,  $k \in \mathbb{N}$ , such that at least one agent  $i$  satisfies  $z_i(t_k) \neq z_i^c$ , and for any positive constant  $\xi$ , there exists a  $\delta(z(t_k)) > 0$  such that*

$$\begin{aligned} & - \int_{t_k}^{t_k+\delta} \sum_{i=1}^{N_a} L_i^z(z_{di}^*(\tau; z_i(t_k)), \hat{z}_{-i}(\tau)) d\tau + \delta^2 \xi \\ & \leq - \int_{t_k}^{t_k+\delta} \|z_{dist}^*(\tau; z(t_k)) - z^c\|_Q^2 d\tau, \quad (7) \end{aligned}$$

for any  $\delta \in (0, \delta(z(t_k)))$ . If  $z(t_k) = z^c$ , then the equation above holds with  $\delta(z(t_k)) = 0$ .

By making the following assumption, we are able to obtain an analytic bound on the update period from the integral expression above.

**Assumption 4** *The interval of integration  $[t_k, t_k+\delta]$  for the expressions in equation (7) is sufficiently small that first-order Taylor series approximations of the integrands is a valid approximation for any  $z(t_k) \in Z_{\Sigma}$ .*

**Lemma 5** *The margin in Lemma 4 is attained with*

$$\delta(z(t_k)) = \frac{(\gamma - 1) \|z(t_k) - z^c\|_Q^2}{\xi + \gamma \lambda_{\max}(Q) (R^2 + U_{\max}^2)}, \quad (8)$$

given the state and control bounds  $R$  and  $U_{\max}$ , respectively.

Since  $\delta(z(t_k))$  depends on  $\|z(t_k) - z^c\|_Q^2$ , a centralized computation is required to generate equation (8) at each receding horizon update. Otherwise, a distributed consensus algorithm could be run in parallel to determine  $\|z(t_k) - z^c\|_Q^2$ , or a suitable lower bound on it. In the dual-mode version [8], no such centralized computation is required online, since a fixed bound on the update period is computed off-line and applied for every receding horizon update. The main theorem of this paper is now given.

**Theorem 1** *Under Assumptions 1-4, for a given fixed horizon time  $T > 0$  and for any state  $z(t_{-1}) \in Z_{\Sigma}$  at initialization, if the update time  $\delta$  satisfies  $\delta \in (0, \delta(z(t_k)))$ ,  $k \in \mathbb{N}$ , where  $\delta(z(t_k))$  is defined in equation (8), then  $z^c$  is an asymptotically stable equilibrium point of the closed-loop system (5) with region of attraction  $Z_{\Sigma}$ , an open and connected set.*

After applying the previous lemmas,  $J_{\Sigma}^*(z(t_k), T)$  is shown to be a Lyapunov function for the closed-loop system and

the remainder of the proof follows closely along the lines of the proof of Theorem 1 in [4]. From equation (8), we observe that  $\delta(z(t_k)) \rightarrow 0$  as  $z(t_k) \rightarrow z^c$ . As a consequence, the control comparison constraint gets tighter, and the communication between neighboring agents must happen with increasing bandwidth, as the agents approach their control objective. To mitigate these problems, a small fixed upper bound on  $\delta$  is provided in [8] that guarantees all agents have reached their terminal constraint sets via the distributed receding horizon control, making it safe to henceforth apply the decoupled linear feedbacks. The result is called dual-mode distributed receding horizon control.

In the next section, formations of vehicles are stabilized using the centralized and distributed receding horizon controllers defined in this section. In the simulations, it is observed that for a fixed, small value for the update parameter  $\delta$ , convergence is obtained with good accuracy. The distributed receding horizon controller is applied for all time and switching to the decoupled feedbacks is not employed.

## 4 Formation Stabilization Example

A simulation of a four vehicle formation is presented in this section. The objective is a finger-tip formation that tracks the reference trajectory  $(q_{\text{ref}}(t), \dot{q}_{\text{ref}}(t)) \in \mathbb{R}^4$ , defined as

$$q_{\text{ref}}(t) = \begin{cases} (t, 0.0), & t \in [0.0, 10.0) \\ (10.0, 10.0 - t), & t \in [10.0, \infty) \end{cases}, \quad (9)$$

where  $t_0 = 0.0$  in the notation of the previous sections. The error system for any agent  $i$  has state  $(q_i - q_{\text{ref}}, \dot{q}_i - \dot{q}_{\text{ref}})$  and dynamics  $\dot{q}_i = u_i$ . The jump in the reference velocity at time  $t = 10.0$  serves to examine how well the error dynamics are stabilized for two different legs of the reference trajectory. The state and control constraint sets are defined as  $\mathcal{Z} = \mathbb{R}^4$  and

$$\mathcal{U} = \{(u_1, u_2) \in \mathbb{R}^2 : -1 \leq u_j \leq 1, j = 1, 2\}.$$

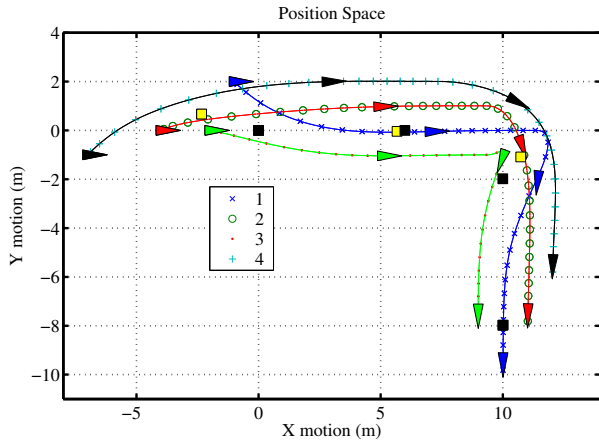
To eliminate any offset between the center of geometry of the formation and the reference trajectory, we set the formation path to  $q_d(t) = (0.0, 0.0)$  for all  $t \geq 0.0$ . In terms of the error dynamics, the tracking cost thus becomes  $\|(q_1 + q_2 + q_3)/3 - q_{\text{ref}}\|^2$ . The vector formation graph is defined by vertices  $\mathcal{V} = \{1, 2, 3, 4\}$  and relative vectors  $\mathcal{E} = \{(1, 2), (1, 3), (2, 4)\}$ . The desired relative vectors are defined for the two legs of the reference trajectory as

$$\begin{aligned} d_{12} = d_{24} &= \begin{cases} (-2, 1), & t \in [0, 10) \\ (1, 2), & t \in [10, \infty) \end{cases}, \\ d_{13} &= \begin{cases} (-2, -1), & t \in [0, 10) \\ (-1, 2), & t \in [10, \infty) \end{cases}. \end{aligned}$$

The common rotation in the vectors at time  $t = 10$  is match the heading of the fingertip formation with the heading of the reference trajectory. The initial conditions for each agent are given as  $q_1(0.0) = (-1, 2)$ ,  $q_2(0.0) = (-4, 0)$ ,  $q_3(0.0) = (-2, 0)$  and  $q_4(0.0) = (-7, -1)$ , with  $\dot{q}_i(0.0) = (0, 0)$  for each agent  $i \in \mathcal{V}$ . In both centralized and distributed receding horizon implementations, a horizon time

of  $T = 5.0$  and update parameter of  $\delta = 0.5$  are used. Also, the following weighting parameter values are consistent in both implementations:  $\omega = 2.0$ ,  $\nu = 1.0$  and  $\mu = 2.0$ .

To solve the optimal control problems numerically, we employ the Nonlinear Trajectory Generation (NTG) software developed at Caltech. A detailed description of NTG as a real-time trajectory generation package for constrained mechanical systems is given in [13]. For the centralized receding horizon control law, parameter values in the optimal control problem satisfy sufficient conditions for stability [8]. The finger-tip formation response is shown in Figure 1. The four closed-loop position trajectories of the vehicles



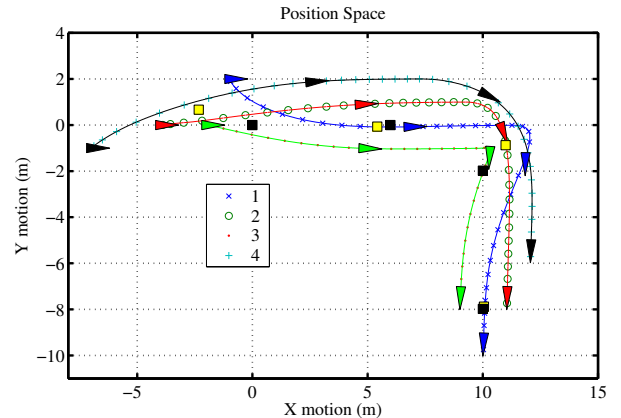
**Figure 1:** Four vehicle formation using centralized receding horizon control.

are shown, with each vehicle depicted by a triangle. The heading of any triangle shows the direction of the corresponding velocity vector. The symbols along each trajectory mark the points at which the receding horizon updates occur. The legend identifies a symbol with a vehicle number for each trajectory. The vehicles are shown at the snapshots of time 0.0, 6.0, 12.0 and 18.0 seconds. Also shown at these instants of time are the reference trajectory position  $q_{\text{ref}}(t)$ , identified by the black square, and the average position of the core vehicles  $q_{\Sigma}(t)$ , identified by the yellow square. The tracking part of the cooperative objective is achieved when  $q_{\Sigma}(t) = q_{\text{ref}}(t)$ , i.e., when the two squares are perfectly overlapping.

At time 6.0, the vehicles are close to the desired formation, and the squares are nearly overlapped, indicating that the tracking objective is being reached. At time 12.0, the snapshot shows the formation reconfiguring to the change in heading of the reference trajectory which occurred at time 10.0. At time 18.0, the objective has again been met with good numerical precision.

For the distributed receding horizon implementation, the initial state at time 0.0 is used for initialization ( $t_{-1} = 0.0$ ), as described in Definition 5. Regarding the conditions in Assumption 2, we first choose  $Q_i = \lambda_{\max}(Q)I_{(4)}$ , where

$\lambda_{\max}(Q) \approx 6.85$ . As in the centralized case,  $K_i$  is defined as the linear quadratic regulator and  $P_i$  the corresponding stable solution to the algebraic Riccati equation. Following the steps in [8], we can show that  $\alpha_i = 0.33$  guarantees that the conditions in the assumption will hold. Finally, we set  $\gamma = 2$  in the cost functions of the distributed optimal control problems. After initialization, the control comparison constraint is enforced, setting  $\kappa = 2$ . The finger-tip formation response is shown in Figure 2. The performance



**Figure 2:** Four vehicle formation using distributed receding horizon control.

is close to that of the centralized implementation. At snapshot time 6.0, the formation is slightly lagging the reference, compared to the centralized version. Also, vehicles 1 and 3 in particular slightly overshoot, in comparison to their centralized counterparts, when the reference changes. The overshoot can be attributed to the assumption that neighbors are reacting to the reference  $q_{\text{ref}} = (t, 0)$  for  $t \in [10, 15]$ , while in reality neighbors are reacting to the actual  $q_{\text{ref}} = (10, 10 - t)$  for  $t \in [10, 15]$ . After the next update, i.e., for time  $t \geq 10.5$ , performance improves as assumed information contains the influence of the actual reference heading. Note that the overshoot is eliminated if the initialization procedure is redone at time 10.0. At time 18.0, the formation objective is close to being met, and for slightly more time the same precision as the centralized implementation is achieved.

A more naive approach is when neighbors are assumed to have *zero* control and the control comparison constraint is not enforced ( $\kappa = +\infty$ ). This was explored in simulations in a previous paper [7], as well as in [8] where the response is characterized by overshoot, as agents believe neighbors will continue along vectors tangent the path over the entire optimization horizon at each update. If the horizon time  $T$  is shortened, overall performance improves, as the assumption becomes more valid. The reason is that a straight line approximation is generally a valid approximation locally, and shrinking  $T$  means the assumption should hold over a more local domain, relative to larger values of  $T$ . In the formulation in [11] a similar effect is observed.

Since agents are relying on the assumption that neighbors keep doing what they *were* doing, and the control comparison constraint ensures that the assumption is not too far off, stability is ensured. In fact, if the comparison constraint is removed, stability is observed in the simulation for the chosen parameter values above. The sensitivity to horizon time when neighbors are assumed to continue along straight-line paths, and as observed in the formulation in [11], is no longer present.

Regarding the communication requirements of transmitting assumed controls to neighboring agents, in the NTG formulation corresponding to the simulations above, 14 B-spline coefficients specified the two-dimensional assumed control trajectory of each agent. In comparison, when agents assume neighbors continue along straight lines, 4 numbers much be communicated at each update, representing the initial condition of the state at the update time. Polynomial representations of trajectories in the optimization problem, when valid, can aid in keeping the communication requirements closer to that of traditional decentralized schemes.

## 5 Conclusions and Extensions

A centralized optimal control problem, whose cost couples the states of a set of dynamically decoupled subsystems, is decomposed into a set of distributed optimal control problems for a distributed receding horizon implementation. The implementation requires an additional constraint in the local optimal control problems, namely a constraint ensuring that assumed and applied control trajectories not deviate too far from one another. Asymptotic stability is proven in the absence of uncertainty and for sufficiently fast receding horizon updates.

In the generalization of the theory, heterogeneous nonlinear dynamics and coupling state constraints between neighboring agents are possible [6]. The dimension of each distributed optimal control problem is equal to that of an optimal control problem of the single corresponding agent, so the implementation is scalable. Thus, there is considerable improvement in tractability over the centralized problem, particularly when the number of agents  $N_a$  is large. Additionally, no particular communications topology is required, aside from unidirectional links. If the trajectories are known to be sufficiently smooth, and polynomial-based approximations are valid, the communication requirements need not be substantially worse than that of other decentralized schemes.

We should also emphasize that the multi-vehicle formation stabilization problem is a venue. In other problems where the performance objective, specifically the integrated cost, is decomposable in such a way that the summation recovers the centralized cost (multiplied by a factor) the approach is applicable. The theory will ultimately be applied to the Caltech Multi-Vehicle Wireless Testbed [5]. Other venues for application of the theory may exist, for example, in dynamic formulations of resource allocation problems in networks,

or in dynamic game theoretic settings. For instance, the approach by Baglietto *et al* [2], which involves stochastic approximations, for distributed dynamic routing in a network could be compared to a discrete-time version of the our distributed receding horizon control law.

## References

- [1] L. Acar. Boundaries of the receding horizon control for interconnected systems. *Journal of Optimization Theory and Applications*, 84(2), 1995.
- [2] M. Baglietto, T. Parisini, and R. Zoppoli. Neural approximations and team theory for dynamic routing: A receding horizon approach. In *Proceedings of the IEEE Conference on Decision and Control*, Phoenix, AZ, 1999.
- [3] D. P. Bertsekas and J. N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, 1997.
- [4] H. Chen and F. Allgöwer. A quasi-infinite horizon nonlinear model predictive scheme with guaranteed stability. *Automatica*, 14(10):1205–1217, 1998.
- [5] L. Cremean, W.B. Dunbar, D. van Gogh, J. Meltzer, R.M. Murray, E. Klavins, and J. Hickey. The Caltech multi-vehicle wireless testbed. In *Proceedings of the Conference on Decision and Control*, Las Vegas, NV, 2002.
- [6] W. B. Dunbar. *Distributed Receding Horizon Control of Multi-Agent Systems: Theoretical and Numerical Algorithms (In Preparation)*. PhD thesis, California Institute of Technology, 2004.
- [7] W. B. Dunbar and R. M. Murray. Model predictive control of coordinated multi-vehicle formations. In *Proceedings of the IEEE Conference on Decision and Control*, Las Vegas, NV, 2002.
- [8] W. B. Dunbar and R. M. Murray. Distributed receding horizon control with application to multi-vehicle formation stabilization. Technical Report 04-001, Control and Dynamical Systems, California Institute of Technology, 2004. Available online: [http://www.cds.caltech.edu/~dunbar/reports/DRHC\\_dunbar.pdf](http://www.cds.caltech.edu/~dunbar/reports/DRHC_dunbar.pdf). Submitted to *Automatica*, January, 2004.
- [9] D. Jia and B. H. Krogh. Distributed model predictive control. In *Proceedings of the American Control Conference*, 2001.
- [10] D. Jia and B. H. Krogh. Min-max feedback model predictive control for distributed control with communication. In *Proceedings of the American Control Conference*, 2002.
- [11] T. Keviczky, F. Borrelli, and G. J. Balas. Model predictive control for decoupled systems: A study of decentralized schemes. In *Submitted to the American Control Conference*, Boston, MA, 2004.
- [12] D.Q. Mayne, J.B. Rawlings, C.V. Rao, and P.O.M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36:789–814, 2000.
- [13] M. B. Milam, K. Mushambi, and R. M. Murray. A new computational approach to real-time trajectory generation for constrained mechanical systems. In *Proceedings of the Conference on Decision and Control*, 2000.
- [14] N. Motee and B. Sayyar-Rodsari. Optimal partitioning in distributed model predictive control. In *Proceedings of the American Control Conference*, 2003.
- [15] W. Ren and R.W. Beard. A decentralized scheme for spacecraft formation flying via the virtual structure approach. *Journal of Guidance, Control and Dynamics*, 27(1):73–82, January 2004.