Article:

Trodden, P. and Richards, A. (2013) Cooperative distributed MPC of linear systems with coupled constraints. Automatica, 49 (2). 479 - 487. ISSN 0005-1098

https://doi.org/10.1016/j.automatica.2012.11.007

Cooperative distributed MPC of linear systems with coupled constraints[☆]

Paul Trodden[a,*], Arthur Richards[b]

[a]School of Mathematics, University of Edinburgh, James Clerk Maxwell Building, King's Buildings, Mayfield Road, Edinburgh EH9 3JZ, United Kingdom
[b]Department of Aerospace Engineering, University of Bristol, Queen's Building, University Walk, Bristol BS8 1TR, United Kingdom

Abstract

This paper develops a cooperative, distributed form of MPC for linear systems subject to persistent, bounded disturbances. The distributed control agents make decisions locally and communicate plans with each other. Cooperation is promoted by consideration of a greater portion of the system-wide objective by each local agent; specifically, a local agent designs hypothetical plans for other agents, sacrificing local performance for the benefit of system-wide performance. These hypothetical plans are never communicated and no negotiation takes place. The method guarantees robust feasibility by permitting only one agent to optimize per time step, while 'freezing' the plans of others, and sufficient conditions are given for robust stability. These properties hold for all structures of cooperation between agents. Thus, a key feature is that coupled constraint satisfaction is compatible with inter-agent cooperation.

Keywords: control of constrained systems; predictive control; decentralization; time-invariant; multi-agent systems

1. Introduction

Model Predictive Control (MPC) has attracted much attention over the last few decades, and theoretical foundations, such as closed-loop stability results, are well established [1, 2]. Recently, research has focused on *distributed* or *decentralized* forms of MPC [3], in which decision making is distributed among agents corresponding to different subsystems making up the whole. The primary challenge is how to coordinate efforts to achieve system-wide feasibility and stability, and numerous strategies have been proposed; see [4] for a comprehensive survey.

An important further problem is that of achieving good system-wide performance. With some degree of *cooperation* between agents, 'greedy' behaviour can be avoided and system-wide performance may improve [5, 6]. In the presence of coupling constraints, however, closed-loop performance is coupled even if the objective and dynamics are decoupled, and applied controls can be severely sub-optimal, despite inter-agent iterations [7]. Thus, the presence of such constraints has been identified as a key open research problem [7]. Approaches to system-wide cooperation for this problem include a hybrid logic rule-based approach [8], dual decomposition [9, 10], and bargaining or repeated exchange and refinement of solutions [5, 11–13]. In [14], agents solve their respective problems independently and simultaneously; though consideration is given to a neighbour's objective, coupling constraint satisfaction is not guaranteed.

In a recent paper [15], we proposed a robust form of distributed MPC, in which each agent designs a local plan that—based on the *tube MPC* method [16] for robustness—consists of a 'tube' for the subsystem to follow rather than a single trajectory; that is, a sequence of robust invariant sets centered on a trajectory for the nominal (*i.e.* disturbance-free) dynamics. The method permits a single agent to optimize per time step. Use of a local feedback law ensures that future states remain within the tube for all possible disturbance realizations, yet without the need for further communication; exchange of information with other agents is only required after an agent optimizes for a new tube, which is not necessarily at each time step.

In this paper, we extend the tube DMPC method [15] to promote inter-agent cooperation. Cooperation with respect to system-wide performance is promoted by including in the local optimizations a consideration of the objectives of other subsystems in a cooperating set of the updating subsystem. A local agent designs not only its own tube, but also *hypothetical* tubes for these agents. Coupled constraint satisfaction is achieved as before by permitting only one agent to optimize per time step, while other agents 'freeze' their plans.

The contribution of this paper, then, is a cooperative robust DMPC method that pairs robust coupled constraint satisfaction and stability with inter-agent cooperation, yet requires no inter-agent iterations or bargaining. The approach to cooperation—of a local agent designing hypothetical plans for others—is similar to that of [14], yet here (i) the choice of other agents with whom to cooperate is unrestricted, and (ii) the inclusion in the optimization of two representations of a neighbour's plan and extra coupling constraints guarantees coupled constraint satisfaction and stability. The approach may be seen as either an extension of the tube DMPC method [15] to promote cooperation while retaining robust feasibility—via a modification to the cost function—or as a constraint modification to [14] in order to guarantee feasibility. The single-update formulation, unrestricted choice of cooperating sets and absence of negotiation or bargaining leads to more flexible communications than other

[☆]This paper was not presented at any IFAC meeting. Research supported by the Engineering and Physical Sciences Research Council, UK, and BAE Systems.

[*]Corresponding author. Tel. +44-131-650-5055. Fax +44-131-650-6553.
Email addresses: paul.trodden@ed.ac.uk (Paul Trodden), arthur.richards@bris.ac.uk (Arthur Richards)

methods, *e.g.* the iterative cooperative schemes of [5, 6, 11–13], which require multiple and repeated information exchanges.

The outline of this paper is as follows. The next section outlines preliminary details. In Section 3 the cooperative DMPC algorithm is developed. Results on robust feasibility and stability are established in Section 4. Inter-agent communication requirements are provided in Section 5, while numerical simulations using the new method are presented in Section 6. Finally, conclusions are drawn in Section 7.

Notation: The matrix mapping of a set is defined as $A\mathcal{B} \triangleq \{c : \exists b \in \mathcal{B}, c = Ab\}$. The operator '$\sim$' denotes the Pontryagin difference [17], a set-shrinking operation defined as $\mathcal{A} \sim \mathcal{B} \triangleq \{a : a + b \in \mathcal{A}, \forall b \in \mathcal{B}\}$. The operator '$\oplus$' denotes the Minkowski sum, defined as $\mathcal{A} \oplus \mathcal{B} \triangleq \{a + b : a \in \mathcal{A}, b \in \mathcal{B}\}$. The double subscript notation $(k + j|k)$ indicates a prediction of a variable j steps ahead from time k. Let $\mathbb{N} \triangleq \{0, 1, 2, \ldots\}$.

2. Preliminaries

We consider a system of N_p linear time-invariant, discrete-time subsystems, the set of which is denoted $\mathcal{P} = \{1, \ldots, N_p\}$, described by the state equations

$$x_p(k + 1) = A_p x_p(k) + B_p u_p(k) + w_p(k), \forall p \in \mathcal{P}, k \in \mathbb{N}, \tag{1}$$

where $x_p \in \mathbb{R}^{N_{x,p}}$, $u_p \in \mathbb{R}^{N_{u,p}}$ and $w_p \in \mathbb{R}^{N_{x,p}}$ are, respectively, the state vector, control input vector, and disturbance acting on subsystem p. Assume that each system (A_p, B_p) is controllable, and that the complete state x_p is available to agent p at each sampling instant. The disturbances are unknown *a priori*, but are assumed to lie in known independent, bounded, compact sets that contain the origin:

$$w_p(k) \in \mathcal{W}_p \subset \mathbb{R}^{N_{x,p}}, \forall p \in \mathcal{P}, k \in \mathbb{N}.$$

Each subsystem $p \in \mathcal{P}$ is subject to local constraints on an output $y_p(k) \in \mathbb{R}^{N_{y,p}}$:

$$y_p(k) = C_p x_p(k) + D_p u_p(k) \in \mathcal{Y}_p,$$

where the set $\mathcal{Y}_p$ is closed. In addition, N_c coupling constraints exist across multiple subsystems. Each coupling constraint $c \in C = \{1, \ldots, N_c\}$ applies to coupling outputs $z_{cp} \in \mathbb{R}^{N_{z,c}}$, the sum of which must lie in a closed set $\mathcal{Z}_c$:

$$z_{cp}(k) = E_{cp} x_p(k) + F_{cp} u_p(k), \text{ and } \sum_{p=1}^{N_p} z_{cp}(k) \in \mathcal{Z}_c.$$

The following definitions identify structure in the coupling, and are used later to determine the requirements for communication. Define $\mathcal{P}_c$ as the set of subsystems involved in constraint c, and C_p as the set of constraints involving subsystem p:

$$\mathcal{P}_c \triangleq \left\{p \in \mathcal{P} : [E_{cp}\ F_{cp}] \neq 0\right\}, \tag{2}$$

$$C_p \triangleq \left\{c \in C : [E_{cp}\ F_{cp}] \neq 0\right\}. \tag{3}$$

Then the set of all other subsystems coupled to p is

$$Q_p = \left(\bigcup_{c \in C_p} \mathcal{P}_c\right) \backslash \{p\}. \tag{4}$$

Assumption 1 (Robust positively-invariant set)**.** *There exists a stabilizing controller K_p for each subsystem (A_p, B_p) and a corresponding robust positively-invariant (RPI) set $\mathcal{R}_p$, satisfying*

$$(A_p + B_p K_p) x_p + w_p \in \mathcal{R}_p, \forall x_p \in \mathcal{R}_p, w_p \in \mathcal{W}_p,$$
$$(C_p + D_p K_p)\mathcal{R}_p \subseteq \mathcal{Y}_p,$$
$$\bigoplus_{p=1}^{N_p} (E_{cp} + F_{cp} K_p)\mathcal{R}_p \subseteq \mathcal{Z}_c, \forall c \in C.$$

Since each (A_p, B_p) is controllable, the existence of K_p is assured. The latter part of the assumption requires that the disturbance set is not too 'large'; a mild assumption for many practical constraints and disturbances [17].

3. Cooperative distributed MPC

Consider the distributed control problem faced by a local control agent at some time step. With the system at a state $x(k) = \{x_1(k), \ldots, x_{N_p}(k)\}$, the tube DMPC method [15] has a sole optimizing agent p devise a plan consisting of initial state and a sequence of future controls

$$\mathbf{u}_p(k) \triangleq \{\bar{x}_p(k|k), \bar{u}_p(k|k), \ldots, \bar{u}_p(k + N - 1|k)\},$$

where $(\bar{x}_p, \bar{u}_p)$ are the state and input of the nominal model $\bar{x}_p(k + 1) = A_p \bar{x}_p(k) + B_p \bar{u}_p(k)$. Meanwhile all other agents $r \neq p$ simply adopt the tails of their respective previous plans, $\mathbf{u}_r^*(k)$, the collection of which is denoted $\mathbf{u}_{-p}^*(k)$. The new plan $\mathbf{u}_p(k)$ is obtained by agent p minimizing a *local* cost function $J_p(\mathbf{u}_p)$ subject to local constraints on $\mathbf{u}_p(k)$ and coupling constraints on $(\mathbf{u}_p(k), \mathbf{u}_{-p}^*(k))$.

In the cooperative form developed in this paper, a local agent p additionally designs *hypothetical* plans for others in some cooperating set $\mathcal{N}_p$. Such a plan for an agent $q \in \mathcal{N}_p$ is denoted $\hat{\mathbf{u}}_q$. The local optimization problem is to minimize a weighted sum of local costs

$$J_p(\mathbf{u}_p) + \sum_{q \in \mathcal{N}_p} \alpha_{pq} J_p(\hat{\mathbf{u}}_q),$$

subject to satisfaction of local constraints on $\mathbf{u}_p$ and each $\hat{\mathbf{u}}_q$, and satisfaction of coupling constraints by $\mathbf{u}_p$ with (i) fixed $\mathbf{u}_{-p}^*$, and (ii) the set of $\hat{\mathbf{u}}_q$ and $\mathbf{u}_r^*$ for all $r \notin \{p, \mathcal{N}_p\}$. The additional decision variables $\{\hat{\mathbf{u}}_q\}_{q \in \mathcal{N}_p}$ are internal to p's local decision making and will not be communicated to other agents. Following the optimization, p communicates information about only its own plan $\mathbf{u}_p^{\text{opt}}$. Moreover, there is no obligation for a cooperating set subsystem $q \in \mathcal{N}_p$ to itself optimize at the next step or indeed ever adopt the plan $\hat{\mathbf{u}}_q$. The main point is that the optimizing subsystem p determines its own plan by considering what others may be able to achieve.

This approach has similarities with that of [14], in which a local agent additionally designs plans for other subsystems. However, in that work, the choice of cooperating set $\mathcal{N}_p$ is restricted to the set of coupled agents, Q_p, while here it is unrestricted. Moreover, the representation of constraints is not sufficient to provide coupled constraint satisfaction. The presence of *two* sets of coupling constraints in the optimization is crucial in the development here. Effectively, two different representations of a plan for a cooperating subsystem $q \in \mathcal{N}_p$ appear in the local optimization for p: firstly, a previously published plan, $\mathbf{u}_q^*$, originating from the last time that q optimized, and the plan that subsystem is currently following; secondly, a hypothetical plan, $\hat{\mathbf{u}}_q$, designed locally by agent p. This leads to a key feature of the method; that of promoting inter-agent cooperation yet maintaining robust feasibility of all local decisions.

The cooperative distributed optimization is now formally described. With the system at a state $\{x_1(k), \ldots, x_{N_p}(k)\}$, the optimization problem $\mathbb{P}_p^{\mathcal{N}_p(k)}(x_p(k); Z_p^*(k))$ for an agent p is

$$\min_{\{\mathbf{u}_p(k), \hat{\mathbf{u}}_{\mathcal{N}_p(k)}\}} J_p(\mathbf{u}_p(k)) + \sum_{q \in \mathcal{N}_p(k)} \alpha_{pq} J_q(\hat{\mathbf{u}}_q(k)) \tag{5}$$

subject to $\forall j \in \{0, \ldots, N-1\}$:

$$\bar{x}_p(k+j+1|k) = A_p \bar{x}_p(k+j|k) + B_p \bar{u}_p(k+j|k), \tag{6a}$$

$$x_p(k) - \bar{x}_p(k|k) \in \mathcal{R}_p, \tag{6b}$$

$$\bar{x}_p(k+N|k) \in \mathcal{X}_{F_p}, \tag{6c}$$

$$\bar{y}_p(k+j|k) = C_p \bar{x}_p(k+j|k) + D_p \bar{u}_p(k+j|k), \tag{6d}$$

$$\bar{y}_p(k+j|k) \in \tilde{\mathcal{Y}}_p, \tag{6e}$$

$$\forall c \in C_p : \bar{z}_{cp}(k+j|k) = E_{cp} \bar{x}_p(k+j|k) + F_{cp} \bar{u}_p(k+j|k), \tag{6f}$$

$$\bar{z}_{cp}(k+j|k) + \sum_{q \in \mathcal{P}_c \setminus \{p\}} \bar{z}_{cq}^*(k+j|k) \in \tilde{\mathcal{Z}}_c, \tag{6g}$$

$$\forall q \in \mathcal{N}_p(k) : \hat{x}_q(k+j+1|k) = A_q \hat{x}_q(k+j|k) + B_q \hat{u}_q(k+j|k), \tag{6h}$$

$$\hat{x}_q(k|k) = \bar{x}_q^*(k|k-1), \tag{6i}$$

$$\hat{u}_q(k|k) = \bar{u}_q^*(k|k-1), \tag{6j}$$

$$\hat{x}_q(k+N|k) \in \mathcal{X}_{F_q}, \tag{6k}$$

$$\hat{y}_q(k+j|k) = C_q \hat{x}_q(k+j|k) + D_q \hat{u}_q(k+j|k), \tag{6l}$$

$$\hat{y}_q(k+j|k) \in \tilde{\mathcal{Y}}_q, \tag{6m}$$

$$\forall c \in C_q : \hat{z}_{cq}(k+j|k) = E_{cq} \hat{x}_q(k+j|k) + F_{cq} \hat{u}_q(k+j|k), \tag{6n}$$

and $\forall c \in C_{\mathcal{N}_p(k)} \triangleq \bigcup_{i \in \mathcal{N}_p(k)} C_i$:

$$\bar{z}_{cp}(k+j|k) + \sum_{q \in \mathcal{N}_p(k)} \hat{z}_{cq}(k+j|k) + \sum_{r \in \mathcal{P}_c \setminus \{p, \mathcal{N}_p(k)\}} \bar{z}_{cr}^*(k+j|k) \in \tilde{\mathcal{Z}}_c. \tag{6o}$$

In this optimization, the cost function is defined as

$$J_p(\mathbf{u}_p(k)) \triangleq F_p(\bar{x}_p(k+N|k)) + \sum_{j=0}^{N-1} l_p(\bar{x}_p(k+j|k), \bar{u}_p(k+j|k)), \tag{7}$$

where the stage cost $l_p : \mathbb{R}^{N_{x,p}} \times \mathbb{R}^{N_{u,p}} \mapsto \mathbb{R}_{0+}$. The terminal cost $F_p : \mathbb{R}^{N_{x,p}} \mapsto \mathbb{R}_{0+}$, is some cost-to-go beyond the end of the horizon. The sets $\tilde{\mathcal{Y}}_p, \tilde{\mathcal{Z}}_c$ represent the sets $\mathcal{Y}_p, \mathcal{Z}_c$ tightened by margins to allow for uncertainty:

$$\tilde{\mathcal{Y}}_p = \mathcal{Y}_p \sim (C_p + D_p K_p)\mathcal{R}_p,$$

$$\tilde{\mathcal{Z}}_c = \mathcal{Z}_c \sim \bigoplus_{p=1}^{N_p} (E_{cp} + F_{cp} K_p)\mathcal{R}_p.$$

These sets are non-empty by Assumption 1. The sets $\mathcal{R}_p$ are ‘cross-sections’ of the tubes, so that the tubes themselves are given by $\{\bar{x}_p(k|k) \oplus \mathcal{R}_p, \bar{x}_p(k+1|k) \oplus \mathcal{R}_p, \ldots, \bar{x}_p(k+N|k) \oplus \mathcal{R}_p\}$. The sets $\mathcal{X}_{F_p}$ are terminal sets, to which the following applies.

Assumption 2 (Admissible control invariant terminal set)**.** *There exist terminal sets $\mathcal{X}_{F_p}$, and terminal control laws $u_p = \kappa_{F_p}(x_p)$, $\forall p \in \mathcal{P}$, so that $\forall x_p \in \mathcal{X}_{F_p}, A_p x_p + B_p \kappa_{F_p}(x_p) \in \mathcal{X}_{F_p}$, $C_p x_p + D_p \kappa_{F_p}(x_p) \in \tilde{\mathcal{Y}}_p$ and $\sum_{p=1}^{N_p} E_{cp} x_p + F_{cp} \kappa_{F_p}(x_p) \in \tilde{\mathcal{Z}}_c, \forall c \in C$.*

The initial constraints (6i) and (6j) provide the starting point of the hypothetical trajectory $\hat{\mathbf{u}}_q$ for each $q \in \mathcal{N}_p(k)$. It is assumed that any cooperating subsystem q can not optimize its own plan until, at the earliest, the next time step $k+1$. Hence, these predicted trajectories shall only begin to diverge from the previously published trajectories at the $k+1$ prediction step.

Precise details and implications of the coupling constraints applied will be discussed in Section 5. For now, $Z_p^*(k)$ denotes the collection of information about other subsystems' plans that the control agent requires to evaluate the optimization.

This problem is solved in the following Algorithm. It is assumed that the information $Z_p^*(k)$ is known and sufficient; in Section 5 the communication requirements to obtain $Z_p^*(k)$ are identified. We also assume that stabilizing controllers K_p and κ_{F_p}, and sets $\mathcal{R}_p$, $\mathcal{X}_{F_p}$, $\mathcal{Y}_p$, $\mathcal{Z}_c$ are available to each agent. Note that tools and methods are available for computing invariant sets—or approximations to them—and their corresponding control laws, *e.g.* [18, 19].

Algorithm 1 (Cooperative DMPC for a subsystem p)**.**

1. *Set $k = 0$. Wait for feasible solution $\mathbf{u}_p^*(0)$ and information $Z_p^*(0)$ from central initialization agent.*
2. *Sample current state $x_p(k)$.*
3. *Update plan. If $p_k = p$*
   (a) *Choose cooperating set $\mathcal{N}_p(k)$ and weightings α_{pq} for each $q \in \mathcal{N}_p(k)$.*
   (b) *Obtain new plan $\mathbf{u}_p(k) = \mathbf{u}_p^{\text{opt}}(k)$ as solution to problem $\mathbb{P}_p^{\mathcal{N}_p(k)}(x_p(k); Z_p^*(k))$.*
   (c) *Transmit new plan $\mathbf{u}_p(k)$ to other agents.*
   else
   (a) *Renew current plan: $\mathbf{u}_p(k) = \tilde{\mathbf{u}}_p(k)$.*
4. *Apply control $u_p(k) = \bar{u}_p(k|k) + K_p(x_p(k) - \bar{x}_p(k|k))$. Wait one time step, increment k, go to step 2.*

Though the algorithm is executed by all agents in parallel, only a sole agent p_k optimizes at a time step k. All other agents $p \neq p_k$ renew their current plan, by shifting in time the tail of the previous, feasible solution and augmenting with a step of terminal control, the result of which is denoted $\tilde{\mathbf{u}}_p(k)$. The order in which subsystems' plans are optimized is determined by the *update sequence*, $\{p_1, \ldots, p_k, p_{k+1}, \ldots\}$. This is to be chosen by the designer, and may be a static (*i.e.* pre-determined) or dynamic sequence, and may include steps of zero update .

The cooperating set $\mathcal{N}_p(k)$ and the scalar weightings α_{pq} are essentially tuning parameters for the level of cooperation. The parameter $\alpha_{pq} \geq 0$ is the weighting applied to the local objective J_q for $q \in \mathcal{N}_p(k)$; smaller values ($\alpha_{pq} < 1$) place more emphasis on p's own objective and self interest, whilst larger values ($\alpha_{pq} > 1$) have the opposite effect. The size of the cooperating set maps to what portion of the system-wide objective is considered in the local optimization. If $\mathcal{N}_p(k)$ is empty, the objective reverts simply to the function $J_p(\mathbf{u}_p(k))$ of the non-cooperative form. Conversely, as $\mathcal{N}_p(k) \to \mathcal{P}\backslash\{p\}$, the local optimization problem more closely resembles the system-wide, centralized problem, but with modified constraints.

Detailed investigation of the choices of update sequence, cooperating sets and weightings is beyond the scope of this paper; the key point is that the choices are unrestricted and results developed hold for all choices. In [20], a method is proposed for choosing the cooperating sets on-line, based on the structure of the (active) coupling constraints.

The distributed algorithm requires that a feasible initial plan be made available to each control agent, and this is a common assumption of DMPC methods; for example, see [21, 22]. Note this does not imply a centralized optimization must be solved; often a simple feasible solution is available, such as all subsystems remaining stationary [23]. A further requirement is that the terminal set $\mathcal{X}_{F_p}$ for the local optimization be made available centrally, since coupling constraints must be satisfied therein. However, note that no further centralized processing is required from that point on. Following optimization, the agent p_k transmits its new plan to some other agents; precisely which agents is identified in the Section 5.

4. Robust feasibility and stability

Under Assumptions 1 and 2, the system controlled by Algorithm 1 has the properties of robust constraint satisfaction and robust feasibility.

Proposition 1 (Robust feasibility). *Suppose the sequence of controls $\mathbf{u}_p^*(k_0) = \{\bar{x}_p^*(k_0|k_0), \bar{u}_p^*(k_0|k_0), \ldots, \bar{u}_p^*(k_0 + N - 1|k_0)\}$ exists and, for each $p \in \mathcal{P}$, is a feasible (but not necessarily optimal) solution to $\mathbb{P}_p^{\mathcal{N}_p}(x_p(k_0); Z_p^*(k_0))$ at some time step k_0 with $\mathcal{N}_p(k_0) = \emptyset$. Then, (i) $\{\mathbf{u}_p^*(k_0), \mathbf{u}_{\mathcal{N}_p}^*(k_0)\}$, where $\mathbf{u}_{\mathcal{N}_p}^*(k_0) = \{\mathbf{u}_q^*(k_0)\}_{q\in\mathcal{N}_p(k_0)}$, is a feasible solution to $\mathbb{P}_p^{\mathcal{N}_p}(x_p(k_0); Z_p^*(k_0))$ for any $\mathcal{N}_p(k_0) \subseteq \mathcal{P}\backslash\{p\}$; and, (ii) for all $x_p(k_0 + 1) \in A_p x_p(k_0) + B_p u_p(k_0) \oplus \mathcal{W}_p, \forall p \in \mathcal{P}$, where $u_p(k_0) = \bar{u}_p^*(k_0|k_0) + K_p(x_p(k_0) - \bar{x}_p^*(k_0|k_0))$, the candidate solution $\{\tilde{\mathbf{u}}_p(k_0 + 1), \tilde{\mathbf{u}}_{\mathcal{N}_p}(k_0 + 1)\}$ is a feasible solution to $\mathbb{P}_p^{\mathcal{N}_p}(x_p(k_0 + 1); Z_p^*(k_0 + 1))$, where*

$$\tilde{\mathbf{u}}_p(k_0 + 1) = \Big\{\bar{x}_p^*(k_0 + 1|k_0), \bar{u}_p^*(k_0 + 1|k_0), \ldots, \bar{u}_p^*(k_0 + N - 1|k_0), \kappa_{F_p}(\bar{x}_p^*(k_0 + N|k_0))\Big\}, \quad (8)$$

and $\tilde{\mathbf{u}}_{\mathcal{N}_p}(k_0 + 1) = \{\tilde{\mathbf{u}}_q(k_0 + 1)\}_{q\in\mathcal{N}_p(k_0+1)}$, for any $\mathcal{N}_p(k_0 + 1)$. Subsequently, (iii) the resulting closed-loop system controlled by Algorithm 1 is robustly feasible for any update sequence.

Proof. For (i), feasibility of $\mathbf{u}_p^*(k_0)$ for $\mathbb{P}_p^{\mathcal{N}_p}(x_p(k_0); Z_p^*(k_0))$ with $\mathcal{N}_p(k_0) = \emptyset$ implies satisfaction of constraints (6a)–(6g). Note that satisfaction of (6g) implies

$$\sum_{p=1}^{N_p} \bar{z}_{cp}^*(k_0 + j|k_0) \in \tilde{\mathcal{Z}}_c, \forall c \in \mathcal{C}, \quad (9)$$

because, by definitions (2) and (3), $\bar{z}_{cp} = 0$ for all $c \notin \mathcal{C}_p$ and, for any $c \in \mathcal{C}$, $\bar{z}_{cr} = 0$ for all $r \notin \mathcal{P}_c$.

Examining $\mathbb{P}_p^{\mathcal{N}_p}(x_p(k_0); Z_p^*(k_0))$ with any $\mathcal{N}_p(k_0) \subseteq \mathcal{P}\backslash\{p\}$, the solution $\{\mathbf{u}_p^*(k_0), \mathbf{u}_{\mathcal{N}_p}^*(k_0)\}$ is feasible if and only if constraints (6h)–(6o) are satisfied. Consider some $q \in \mathcal{N}_p(k_0)$. Satisfaction of (6h)–(6n) by $\hat{\mathbf{u}}_q(k_0) = \mathbf{u}_q^*(k_0)$ follows immediately from feasibility of $\mathbf{u}_q^*(k_0)$ for $\mathbb{P}_q^{\mathcal{N}_q}(x_q(k_0); Z_q^*(k_0))$ with $\mathcal{N}_q(k_0) = \emptyset$. Constraint (6o), with $\bar{z}_{cp}(\cdot|k_0) = \bar{z}_{cp}^*(\cdot|k_0)$ and $\hat{z}_{cq}(\cdot|k_0) = \bar{z}_{cq}^*(\cdot|k_0)$, becomes identical to (9), and the result is established.

Now consider problem $\mathbb{P}_p^{\mathcal{N}_p}(x_p(k_0 + 1); Z_p^*(k_0 + 1))$ at time $k_0 + 1$, for any $p \in \mathcal{P}$ and with $\mathcal{N}_p(k_0 + 1) = \emptyset$. The candidate plan $\tilde{\mathbf{u}}_p(k_0 + 1)$ satisfies (6a) by construction. For the initial constraint (6b),

$$\begin{aligned} x_p(k_0 + 1) - \bar{x}_p(k_0 + 1|k_0 + 1) &= x_p(k_0 + 1) - \bar{x}_p^*(k_0 + 1|k_0) \\ &= A_{K,p}(x_p(k_0) - \bar{x}_p^*(k_0|k_0)) + w_p(k_0) \\ &\in \mathcal{R}_p, \forall w_p(k_0) \in \mathcal{W}_p, \end{aligned}$$

where $A_{K,p} \triangleq A_p + B_p K_p$, because $x_p(k_0) - \bar{x}_p^*(k_0|k_0) \in \mathcal{R}_p$, and $\mathcal{R}_p$ satisfies Assumption 1. For the terminal constraint (6c),

$$\begin{aligned} \bar{x}_p(k_0 + N + 1|k_0 + 1) &= \bar{x}_p^*(k_0 + N + 1|k_0) \\ &= A_p \bar{x}_p^*(k_0 + N|k_0) + B_p \kappa_{F_p}(\bar{x}_p^*(k_0 + N|k_0)) \\ &\in \mathcal{X}_{F_p} \end{aligned}$$

by Assumption 2. Satisfaction of (6d) is by construction, while (6e) is satisfied by $\bar{y}_p(k_0 + 1 + j|k_0 + 1) = \bar{y}_p^*(k_0 + j + 1|k_0)$ for $j \in \{0, \ldots, N - 2\}$ and satisfaction for $j = N - 1$ is by admissibility of the terminal set:

$$\begin{aligned} \bar{y}_p(k_0 + N|k_0 + 1) &= C_p \bar{x}_p^*(k_0 + N|k_0) + D_p \kappa_{F_p}(\bar{x}_p^*(k_0 + N|k_0)) \\ &\in \tilde{\mathcal{Y}}_p. \end{aligned}$$

For the coupling constraints, (6f) is satisfied by construction, while $\bar{z}_{cp}(k_0 + 1 + j|k_0 + 1) = \bar{z}_{cp}^*(k_0 + j + 1|k_0)$ satisfies (6g) for $j \in \{0, \ldots, N - 2\}$. Satisfaction for $j = N - 1$ is again by admissibility of the terminal set. Consequently, $\tilde{\mathbf{u}}_p(k_0 + 1)$ is a feasible solution to $\mathbb{P}_p^{\mathcal{N}_p}(x_p(k_0+1); Z_p^*(k_0+1))$ with $\mathcal{N}_p(k_0+1) = \emptyset$. Finally, to establish part (ii), the result of part (i) is applied at time step $k_0 + 1$, so that $\{\tilde{\mathbf{u}}_p(k_0 + 1), \tilde{\mathbf{u}}_{\mathcal{N}_p}(k_0 + 1)\}$ is a feasible solution to $\mathbb{P}_p^{\mathcal{N}_p}(x_p(k_0 + 1); Z_p^*(k_0 + 1))$ with any $\mathcal{N}_p(k_0 + 1)$.

Part (iii) follows by applying recursion to (ii). A collection of feasible solutions to each problem $\mathbb{P}_p^{\mathcal{N}_p}(x_p(0); Z_p^*(0))$ implies all subsequent optimizations $\mathbb{P}_p^{\mathcal{N}_p}(x_p(k); Z_p^*(k))$, $k > 0$, are feasible, regardless of update sequence and cooperating sets. □

In order to consider closed-loop stability, first define the global cost as the summation of local costs, including only local, published decision variables $\mathbf{u}_p(k)$—the plans the subsystems are following—and not hypothetical decision variables $\hat{\mathbf{u}}_{\mathcal{N}_p}$:

$$\mathcal{J}(k) \triangleq \sum_{p=1}^{N_p} J_p(\mathbf{u}_p(k)) \quad (10)$$

Then, under the further, following assumptions, Proposition 2 guarantees asymptotic convergence of the states of the controlled system to a neighbourhood of the origin.

Assumption 3 (Terminal cost is local Lyapunov function)**.** *For all $x_p \in \mathcal{X}_{F_p}$ and $p \in \mathcal{P}$,*

$$F_p(A_p x_p + B_p \kappa_{F_p}(x_p)) - F_p(x_p) \le -l_p(x_p, \kappa_{F_p}(x_p)).$$

Together with Assumption 2, these assumptions represent a specific case of the standard assumptions A1–A4 in [1] or equivalently A1 and A2 in [16].

Assumption 4 (Bounded local costs)**.** *The local cost of an adopted plan $\mathbf{u}_p^*(k)$ for any agent $p = p_k$ updating at k satisfies*

$$J_p(\mathbf{u}_p^*(k)) \le J_p(\tilde{\mathbf{u}}_p(k)) + \sum_{i=1}^{N_p} \epsilon_i l_i(\bar{x}_i^*(k-1|k-1), \bar{u}_i^*(k-1|k-1)) \quad (11)$$

for some chosen $0 \le \epsilon_i < 1$, $\forall i \in \{1, \ldots, N_p\}$, where $\tilde{\mathbf{u}}_p(k)$ is the candidate plan for time k, defined by (8)*.*

Lemma 1. *Suppose the solution $\mathbf{u}_p^*(k_0)$ to $\mathbb{P}_p^{\mathcal{N}_p}(x_p(k_0); Z_p^*(k_0))$ at some time step k_0 with $\mathcal{N}_p(k_0) = \emptyset$ exists for all $p \in \mathcal{P}$. Then the solution $\{\tilde{\mathbf{u}}_p(k_0+1), \tilde{\mathbf{u}}_{\mathcal{N}_p}(k_0+1)\}$ to $\mathbb{P}_p^{\mathcal{N}_p}(x_p(k_0+1); Z_p^*(k_0+1))$, defined in Proposition 1, satisfies Assumption 4 for any $p \in \mathcal{P}$ and $\mathcal{N}_p(k_0+1) \subseteq \mathcal{P}\backslash\{p\}$.*

Proof. By Proposition 1, the solution $\{\tilde{\mathbf{u}}_p(k_0+1), \tilde{\mathbf{u}}_{\mathcal{N}_p}(k_0+1)\}$ to $\mathbb{P}_p^{\mathcal{N}_p}(x_p(k_0+1); Z_p^*(k_0+1))$ exists for any $p \in \mathcal{P}$ and $\mathcal{N}_p(k_0+1) \subseteq \mathcal{P}\backslash\{p\}$. For that solution, $J_p(\mathbf{u}_p(k_0+1)) - J_p(\tilde{\mathbf{u}}_p(k_0+1)) = 0$ for any p, which trivially satisfies (11) for all $\epsilon \in [0, 1)$. □

Proposition 2 (Robust asymptotic convergence to $\mathcal{R}_p$)**.** *Suppose the sequence of controls $\mathbf{u}_p^*(k_0) = \{\bar{x}_p^*(k_0|k_0), \bar{u}_p^*(k_0|k_0), \ldots, \bar{u}_p^*(k_0+N-1|k_0)\}$ exists and, for each $p \in \mathcal{P}$, is a feasible (but not necessarily optimal) solution to $\mathbb{P}_p^{\mathcal{N}_p}(x_p(k_0); Z_p^*(k_0))$, at some time step k_0 with $\mathcal{N}_p(k_0) = \emptyset$. Then, for all $x_p(k_0+1) \in A_p x_p(k_0) + B_p u_p(k_0) \oplus \mathcal{W}_p, \forall p \in \mathcal{P}$, where $u_p(k_0) = \bar{u}_p^*(k_0|k_0) + K_p(x_p(k_0) - \bar{x}_p^*(k_0|k_0))$, if Assumption 4 holds the global cost decreases monotonically:*

$$\mathcal{J}^*(k_0+1) \le \mathcal{J}^*(k_0) - \gamma \sum_{i=1}^{N_p} l_i(\bar{x}_i^*(k_0|k_0), \bar{u}_i^*(k_0|k_0)),$$

where $\gamma > 0$, Furthermore, if $l_p(x_p, u_p) \ge c||(x_p, u_p)||$ for some $c > 0$ and $l_p(0, 0) = 0$ then, for each p, $x_p(k) \to \mathcal{R}_p$ and $u_p(k) \to K_p x_p(k)$ as $k \to \infty$.

Proof. $\{\mathbf{u}_p^*(k_0), \mathbf{u}_{\mathcal{N}_p}^*(k_0)\}$, where $\mathbf{u}_{\mathcal{N}_p}^*(k_0) = \{\mathbf{u}_q^*(k_0)\}_{q \in \mathcal{N}_p(k_0)}$, is a feasible solution to $\mathbb{P}_p^{\mathcal{N}_p}(x_p(k_0); Z_p^*(k_0))$ for any $\mathcal{N}_p(k_0) \subseteq \mathcal{P}\backslash\{p\}$. The global cost associated with each agent adopting the local part $\mathbf{u}_p^*(k_0)$ of this solutions is $\mathcal{J}^*(k_0) \triangleq \sum_{i=1}^{N_p} J_i(\mathbf{u}_i^*(k_0))$. By Proposition 1, the candidate $\{\tilde{\mathbf{u}}_p(k_0+1), \tilde{\mathbf{u}}_{\mathcal{N}_p}(k_0+1)\}$, as defined by (8), is a feasible solution to $\mathbb{P}_p^{\mathcal{N}_p}(x_p(k_0+1); Z_p^*(k_0+1))$ for any $\mathcal{N}_p(k_0+1) \subseteq \mathcal{P}\backslash\{p\}$ and any p. The global cost were each p to implement the $\tilde{\mathbf{u}}_p(k_0+1)$ part of this solution is

$$\tilde{\mathcal{J}}(k_0+1) \triangleq \sum_{i=1}^{N_p} J_i(\tilde{\mathbf{u}}_i(k_0+1)) \le \mathcal{J}^*(k_0) - \sum_{i=1}^{N_p} l_i(\bar{x}_i^*(k_0|k_0), \bar{u}_i^*(k_0|k_0)).$$

This is constructed in the standard way (see *e.g.* [1, Sec. 3.3]) by evaluating $\mathcal{J}$ in (10) at k_0 and k_0+1, using the definition of J_i in (7), and applying the inequality in Assumption 3.

However, at this step k_0+1, one agent, $p = p_{k_0+1}$, optimizes while all $r \ne p$ adopt their respective candidate plans. Supposing the cooperating set for p is $\mathcal{N}_p(k_0+1)$, the optimization produces some (not necessarily optimal) solution $\{\mathbf{u}_p^*(k_0+1), \hat{\mathbf{u}}_{\mathcal{N}_{p(k_0+1)}}(k_0+1)\}$—where $\hat{}$ denotes a hypothetical plan—with an optimization cost (5) less than or equal to that for taking the candidate solution for itself and all in its cooperating set. Thus, p adopts the $\mathbf{u}_p^*(k_0+1)$ part of this solution as its adopted plan at time k_0+1, while all non-optimizing $r \ne p$ adopt their respective candidate plans $\tilde{\mathbf{u}}_r(k_0+1)$, with global cost

$$\begin{aligned}\mathcal{J}^*(k_0+1) &= J_p(\mathbf{u}_p^*(k_0+1)) + \sum_{r \ne p} J_r(\tilde{\mathbf{u}}_r(k_0+1)) \\ &= \tilde{\mathcal{J}}(k_0+1) + \left[J_p(\mathbf{u}_p^*(k_0+1)) - J_p(\tilde{\mathbf{u}}_p(k_0+1))\right].\end{aligned}$$

It follows that

$$\mathcal{J}^*(k_0+1) \le \mathcal{J}^*(k_0) - \left(\sum_{i=1}^{N_p} l_i(\bar{x}_i^*(k_0|k_0), \bar{u}_i^*(k_0|k_0)) - \left[J_p(\mathbf{u}_p^*(k_0+1)) - J_p(\tilde{\mathbf{u}}_p(k_0+1))\right]\right).$$

Subsequently, if the $\mathbf{u}_p^*(k_0+1)$ part of the solution to the problem $\mathbb{P}_p^{\mathcal{N}_p}(x_p(k_0+1); Z_p^*(k_0+1))$ at k_0+1, for any p with any $\mathcal{N}_p(k_0+1) \subseteq \mathcal{P}\backslash\{p\}$, satisfies (11) in Assumption 4 for some chosen $\epsilon_i \in [0, 1), \forall i \in \{1, \ldots, N_p\}$,

$$J_p(\mathbf{u}_p^*(k_0+1)) \le J_p(\tilde{\mathbf{u}}_p(k_0+1)) + \sum_{i=1}^{N_p} \epsilon_i l_i(\bar{x}_i^*(k_0|k_0), \bar{u}_i^*(k_0|k_0))$$

then $\mathcal{J}^*(k_0+1) \le \mathcal{J}^*(k_0) - \gamma \sum_{i=1}^{N_p} l_i(\bar{x}_i^*(k_0|k_0), \bar{u}_i^*(k_0|k_0))$, where $\gamma \ge \sum_{i=1}^{N_p}(1-\epsilon_i) > 0$, as $\epsilon_i < 1, \forall i$, and the result is established.

For asymptotic convergence, $l_p(\cdot,\cdot) \ge 0, F_p(\cdot) \ge 0$, hence $\mathcal{J}_p(k) \ge 0$ for all $p \in \mathcal{P}$. Then $\mathcal{J}^*(k+1) - \mathcal{J}_p^*(k) \to 0$ as $k \to \infty$. It follows that $\lim_{k\to\infty} \gamma \sum_{i=1}^{N_p} l_i(\bar{x}_i^*(k|k), \bar{u}_i^*(k|k)) = 0$. Then, because $\gamma > 0$, $\sum_{i=1}^{N_p} l_i(\bar{x}_i^*(k|k), \bar{u}_i^*(k|k)) \to 0$. Furthermore, because each $l_i(x_i, u_i) \ge c\left\|(x_i, u_i)\right\| \ge 0$, for some $c > 0$, then each $l_i(\bar{x}_i^*(k|k), \bar{u}_i^*(k|k)) \to 0$. This implies that, for any p, $\bar{x}_p^*(k|k) \to 0$ and $\bar{u}_p^*(k|k) \to 0$; as $x_p(k) - \bar{x}_p^*(k|k) \in \mathcal{R}_p$ and $u_p(k) = \bar{u}_p(k|k) + K_p(x_p - \bar{x}_p^*(k|k))$, then $x_p(k) \to \mathcal{R}_p$ and $u_p(k) \to K_p x_p(k)$ as $k \to \infty$. □

The condition (11) bounds the value of the *local* cost J_p in the optimization. Thus, the bound sets a limit on the amount by which the local cost J_p of an adopted local plan $\mathbf{u}_p^*$ is permitted to *increase* over the local cost of the candidate plan $\tilde{\mathbf{u}}_p$ in order to benefit other agents. Intuitively, an unbounded increase may lead to instability if repeated by many agents over time.

For an implementation that guarantees stability, (11) may be included in the local optimization as a constraint with $0 \le \epsilon_i < 1$ chosen by the designer. While the summation in the right-hand side of (11) involves all agents in the problem, note that $\epsilon_i = 0$ is permitted for any i. Therefore, no additional communication to that identified in the next section is required to constrain for stability, since any agent's information may be omitted in the constraint by setting $\epsilon_i = 0$. This is similar to the stability constraint approach [24], where performance may be traded with feasibility by tuning the different ϵ_i. Note that although the presence of such a constraint may restrict optimality, a feasible solution always exists by Lemma 1.

Where (11) is not included as a constraint in the local optimization, note that in many cases the optimization will tend to satisfy the constraint anyway, since the left-hand side of the inequality is related to what is being minimized in the objective. In particular, as the cost weightings α_{pq} in (5) take on low values and approach zero, the objective weights most heavily the local cost for p, which appears in the left-hand side of (11).

5. Communication requirements

It remains to evaluate exactly what information, denoted $Z_p^*(k)$, is required in the local optimization for p. The following standing assumption shall apply to the subsequent analysis.

Assumption 5 (Construction from $\mathbf{u}_q(k)$)**.** *Each agent $p \in \mathcal{P}$ has* a priori *knowledge of static model parameters for all other subsystems, including dynamics (A_q, B_q), controller κ_{F_q}, and constraint sets $C_q, D_q, \tilde{\mathcal{Y}}, E_{cq}, F_{cq}, \tilde{\mathcal{Z}}_c$, so that, given the plan $\mathbf{u}_q(k)$, all predicted states and outputs may be constructed.*

Examining the local optimization, firstly we consider the initial constraints (6i) and (6j). To evaluate these constraints, the local agent p_k must have knowledge of $\bar{x}_q^*(k|k-1)$ and $\bar{u}_q^*(k|k-1)$ for each $q \in \mathcal{N}_{p_k}(k)$. Define $\hat{k}_p$ as the last time, before the current step k, a subsystem p optimized its plan:

$$\hat{k}_p(k) \triangleq \max_{k' \in \{k' < k | p_{k'} = p\}} k'.$$

Suppose that the plan $\mathbf{u}_q^*(\hat{k}_q)$ for a subsystem $q \in \mathcal{N}_{p_k}(k)$, from this latest update step, has been made available to the agent p_k:

$$\mathbf{u}_q^*(\hat{k}_q) = \{\bar{x}_q^*(\hat{k}_q|\hat{k}_q), \bar{u}_q^*(\hat{k}_q|\hat{k}_q), \bar{u}_q^*(\hat{k}_q+1|\hat{k}_q), \ldots, \bar{u}_q^*(\hat{k}_q+N-1|\hat{k}_q)\}.$$

Then p_k may construct values as required:

$$\bar{u}_q^*(k|k-1) = \bar{u}_q^*(k|\hat{k}_q),$$
$$\bar{x}_q^*(k|k-1) = A_q^{(k-\hat{k}_q)} \bar{x}_q^*(\hat{k}_q|\hat{k}_q) + \sum_{i=0}^{k-\hat{k}_q-1} A_q^i B_q \bar{u}_q^*(\hat{k}_q + i|\hat{k}_q),$$

for all $k \leq \hat{k}_q + N - 1$. For greater values of k, states and inputs may be constructed using the terminal control law $\kappa_{F_q}(x_q)$.

Now we turn to the coupling constraints, (6g) and (6o), in the optimization. The communication requirements for cooperative DMPC are higher than for non-cooperative DMPC [15] since the problem includes two sets of coupling constraints. Firstly, by constraint (6g), the coupling outputs $\bar{z}_{cp_k}(\cdot|k)$ of the optimizing subsystem p_k satisfy the coupling constraints $c \in C_{p_k}$ when taken with the previously published outputs $\bar{z}_{cq}^*(\cdot|k)$ of subsystems $\forall q \in \mathcal{P}_c \backslash \{p_k\}$, which may include some $q \in \mathcal{N}_{p_k}(k)$. (Even if p_k includes q in its cooperating set, any shared coupling constraints are still evaluated with q's *previously published* plan).

Secondly, by constraint (6o), the sum of hypothetical coupling outputs $\hat{z}_{cq}(\cdot|k)$ over all $q \in \mathcal{N}_{p_k}(k)$ must be consistent with the coupling outputs of p_k, and also with the previously published outputs of all other subsystems coupled to any $q \in \mathcal{N}_{p_k}(k)$. That is, the collection of $\bar{z}_{cr}^*(\cdot|k), \forall c \in C_{\mathcal{N}_{p_k}(k)}$, where

$$C_{\mathcal{N}_{p_k(k)}} \triangleq \bigcup_{i \in \mathcal{N}_{p_k}(k)} C_i,$$

is required from all r in the union

$$Q_{\{p_k, \mathcal{N}_{p_k}\}} \triangleq \bigcup_{i \in \mathcal{N}_{p_k}(k)} Q_i \backslash \{p_k, \mathcal{N}_{p_k}(k)\}. \quad (12)$$

Note that in each case, the structure in the coupling constraints, identified in (2) and (3), has been exploited. For example, for (6g), only constraints $c \in C_p$ are applied, as by definition (3), $\bar{z}_{cp}(k+j|k) = 0$ for all other constraints $c \notin C_p$, so these outputs do not affect the update of subsystem p. Furthermore, the summation in (6g), for each c, includes output terms from only those subsystems in $\mathcal{P}_c$; by definition (2), $\bar{z}_{cr}(k+j|k) = 0$ for all other subsystems $r \notin \mathcal{P}_c$. The coupling terms $\bar{z}_{cq}^*(k+j|k), \forall q \in \mathcal{P}_c \setminus \{p\}$ are not affected by the decision variables $\mathbf{u}_p(k)$, so they appear as fixed values in (6g), denoted by *—hence, values for $\bar{z}_{cq}^*(k+j|k), \forall c \in C_p$, are required from all other subsystems q in Q_p. A similar analysis for (6o) results in only the constraints $C_{\mathcal{N}_{p_k}(k)}$ being applied.

We note, therefore, that it is not necessary to obtain the whole plan $\mathbf{u}_q^*(k)$ from some other q. Instead, define a message vector from subsystem p regarding constraint c at time k as

$$\mathbf{m}_{cp}(k) \triangleq \begin{bmatrix} \bar{z}_{cp}^*(k|k)^{\mathrm{T}} & \ldots & \bar{z}_{cp}^*(k+N-1|k)^{\mathrm{T}} & \bar{x}_p^*(k+N|k)^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}},$$

which includes the coupling outputs and terminal state. Again, the * superscript denotes a feasible solution. At this point, it is worth noting the difference between a message and a plan. A message may be constructed from a plan given the system and constraint matrices. However, a plan—or the initial states and inputs—may not, in general, be reconstructed from a message. Often, a message is a smaller representation of a plan, and in a convenient format that aids direct evaluation of the coupling constraints without further matrix operations.

Next, to allow a local agent to form current coupling data based on previous information, define a propagation matrix,

$$\Pi_{cp} \triangleq \begin{bmatrix} 0 & I & 0 & \ldots & 0 \\ 0 & 0 & I & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & (E_{cp} + F_{cp}K_{F_p}) \\ 0 & 0 & 0 & \ldots & (A_p + B_pK_{F_p}) \end{bmatrix},$$

assuming a linear terminal control law, *i.e.* $\kappa_{F_p}(x_p) = K_{F_p}x_p$, so that $\mathbf{m}_{cp}(k) = \Pi_{cp}\mathbf{m}_{cp}(k-1)$ is the message at time k for a non-updating subsystem $p \neq p_k$. Then the message at k for a subsystem p that last optimized at $\hat{k}_p$ is $\mathbf{m}_{cp}(k) = \Pi_{cp}^{(k-\hat{k}_p)}\mathbf{m}_{cp}(\hat{k}_p)$.

Relating this back to the information that is required by p_k to evaluate (6g), this information is obtained as

$$\{\mathfrak{I}\mathbf{m}_{cq}(k)\}_{c\in C_{p_k},q\in Q_{p_k}} = \{\mathfrak{I}\Pi_{cq}^{(k-\hat{k}_q)}\mathbf{m}_{cq}(\hat{k}_q)\}_{c\in C_{p_k},q\in Q_{p_k}},$$

where the matrix operator $\mathfrak{I} \triangleq \mathrm{diag}(I, I, \ldots, 0)$ removes the terminal states. The inclusion of the terminal state $\bar{x}_p(k+N|k)$ in the message permits the correct propagation for steps $k > \hat{k}_q+N$. The information for (6o) is obtained in an identical manner.

Motivated by these requirements for information to evaluate initial and coupling constraints, the information requirement for evaluation of the constraints in the cooperative problem $\mathbb{P}_p^{\mathcal{N}_p}(x_p(k); Z_p^*(k))$ is now stated.

Requirement 1 (Information for $Z^*_{p_k}(k)$)**.** *At a time step k, the control agent for an optimizing subsystem p_k must have received*

1. *plans* $\mathbf{u}_q^*(\hat{k}_q)$ *from all* $q \in \mathcal{N}_{p_k}(k)$*;*
2. *messages* $\mathbf{m}_{ci}(\hat{k}_q), \forall c \in C_{p_k}$*, from all* $q \in Q_{p_k}$*;*
3. *messages* $\mathbf{m}_{cr}(\hat{k}_r), \forall c \in C_{\mathcal{N}_{p_k}(k)}$*, from all* $r \in Q_{\{p_k,\mathcal{N}_{p_k}\}}$*.*

The first part of the requirement ensures all that initial constraints can be evaluated. Satisfaction of the second part means p_k can evaluate all its coupling constraints, with respect to the previously published plans of coupled subsystems. The final part means that the coupling constraints for every cooperating subsystem may be evaluated, using the hypothetical plans for $q \in \mathcal{N}_p(k)$ with published plans for any coupled subsystems not in the cooperating set.

Note that if $\mathcal{N}_{p_k}(k)$ is empty, p_k does not require plans $\mathbf{u}_q^*(\hat{k}_q)$ from any other agent (part 1), and the union set $Q_{\{p_k,\mathcal{N}_{p_k}\}}$ (12) of part 3 becomes empty; hence, Requirement 1 reduces to requiring only coupling data from those $q \in Q_{p_k}$. Conversely, if $\mathcal{N}_{p_k}(k) = \mathcal{P}\backslash\{p_k\}$, p_k requires plans from *all* others, but each term $Q_i\backslash\{p_k, \mathcal{N}_{p_k}(k)\}$ of the union in part 3 is empty, and no previously published outputs appear in constraint (6o). In between these extremes, p_k requires information from (i) those in the cooperating set, (ii) those to whom p_k is coupled, and (iii) those to whom any $q \in \mathcal{N}_{p_k}(k)$ is coupled.

It is assumed that the communication availability is sufficient to meet the information requirement. Thus, the communication step in Algorithm 1 conservatively specifies transmission to all other subsystems following update. While this may seem significant, it should be noted that to meet the requirement it is sufficient for one agent to transmit its plan to others only after that plan has changed, *i.e.* as a result of optimization. Moreover, it is not necessary for an agent to update at every time step, and robust coupled constraint satisfaction and stability are guaranteed for any choices of update sequence and cooperating sets. Thus, data exchanges need not occur at every time step, and the cooperating set and update sequence may be tailored to exploit this flexibility, as has been shown for the latter in [15]. In comparison, the approaches to cooperation based on inter-agent iteration or bargaining [5, 6, 11–13] require multiple and repeated information exchanges at each time step in order to achieve constraint satisfaction and stability.

Finally, note that if the cooperating set is such that the local optimization is similar in size to the centralized problem, the performance will not match that of centralized, owing to the hypothetical plans not being communicated and the enforced delay in other agents' updates. However, even in this scenario the approach offers some advantages, namely some degree of robustness to agent failure or communication breakdown, since the other agents in the system are not reliant on an optimizing agent for new plans in order to maintain constraint satisfaction.

6. Numerical examples

6.1. Integrators with coupled inputs

The first example considers a set of six identical integrators, with dynamics $\dot{x}_p = u_p, \forall p \in \{1, \ldots, 6\}$, where $x_p \in \mathbb{R}, u_p \in \mathbb{R}$. To show more clearly the effects of cooperation, we assume no uncertainty or additive disturbances. The control objective is to regulate the system to the origin from an initial state $x_p(0) = 10, \forall p$, while minimizing the quadratic cost

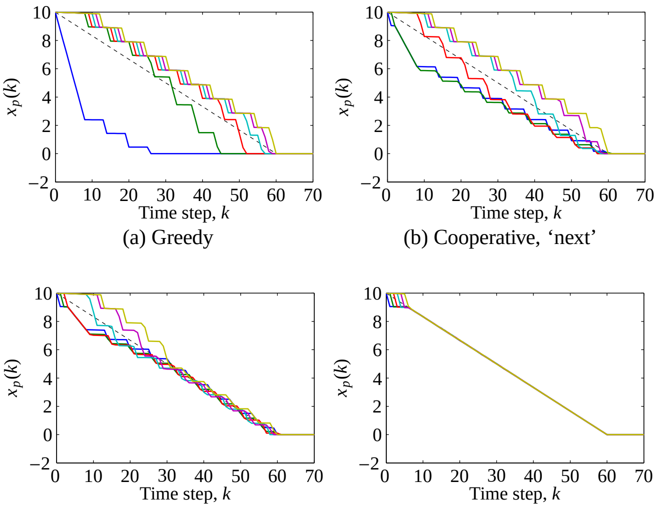
$$\sum_k \sum_{p=1}^{N_p} x_p^2(k) \tag{13}$$

The integrators are coupled via the input constraints; each input is constrained locally as $|u_p(k)| \leq 1$, and a limit is imposed on the sum of inputs over all subsystems: $\sum_{p=1}^{N_p}|u_p(k)| \leq 1$. State constraints are defined by $\mathcal{X}_p = \{x_p : |x_p| \leq 10\}, \forall p$.

For the controller, the dynamics are discretized assuming zero-order hold and a sampling period of 1 s. The prediction horizon is 8 steps. The terminal set is chosen to be equal to the state constraint set, *i.e.* $\mathcal{X}_{F_p} = \mathcal{X}_p$. Within this set, the terminal control law $\kappa_{F_p}(x_p) = K_{F_p}x_p = -0.001x_p$ satisfies Assumption 2. The local cost J_p, defined by (7), is a finite-horizon approximation to (13) with $l_p(\bar{x}_p, \bar{u}_p) = \bar{x}_p^2$ and $F_p(\bar{x}_p) = 1000\bar{x}_p^2$. This satisfies Assumption 3 for all $x_p \in \mathcal{X}_{F_p}$ with $\bar{u}_p = K_{F_p}\bar{x}_p$. The absence of uncertainty means that $\bar{x}_p(k|k) = x_p(k), \forall k$, no feedback controller K_p is required, and the local cost is zero only at the origin. The integrators are each initialized with the feasible plan $\bar{u}_p^*(j|0) = K_{F_p}\bar{x}_p(j|0), j \in \{0, \ldots, N-1\}$, placing each on a linearly decreasing trajectory to the origin.

Figure 1 shows the state evolution of the six integrators when controlled by distributed and centralized MPC (CMPC). When centralized control is used, an equal share of the available control input is allocated to each integrator, and this solution attains the lowest system-wide cost. With distributed control, the agents optimize in the simple alternating sequence, $\{1, 2, 3, \ldots\}$. Thus, since each agent has an initial plan using only a small fraction of the total control available, the first agent to optimize has the greatest control authority, in that it *may* elect to use all of the available shared control effort. Subsequent agents optimizing would then have zero control available until the first agent relinquishes control. This greedy behaviour occurs for the non-cooperative DMPC, *i.e.* Algorithm 1 with $\mathcal{N}_p = \emptyset$.

Figures 1(b)–(d) show trajectories for cooperative DMPC with three different schemes for choosing the cooperating set: 'next', where $\mathcal{N}_p = 1 + (p \bmod N_p)$; 'next two', where $\mathcal{N}_p = \{1 + (p \bmod N_p), 1 + (p + 1 \bmod N_p)\}$; and all $p \neq p_k$. For each



(a) Greedy (b) Cooperative, 'next' (c) Cooperative, 'next two' (d) Cooperative, all $p \neq p_k$

Figure 1: State evolution of controlled integrators. Trajectory of the system when controlled by centralized MPC shown dotted.

Table 1: Aggregate closed loop costs, as percentage increase over centralized cost, for different cooperation schemes.

Size of $\mathcal{N}_{p_k}$	0	1	2	3	4	5
Cost increase (%)	12.02	4.25	1.12	0.38	0.13	0.04

cooperative optimization, $\alpha_{pq} = 1$ in the objective. The trajectories become progressively 'fairer' as the size of the cooperating set increases, converging towards the CMPC state trajectory.

The stability constraint (11) was not included in the optimizations. Nevertheless, the value of the local cost was found to satisfy the condition at every time step for $\epsilon = 0$—the lowest upper bound—hence guaranteeing convergence.

Closed-loop costs are shown in Table 1, with 'next three' and 'next four' schemes shown in addition. The aggregate closed-loop cost decreases as the number of agents in the cooperating set increases. The largest decrease is seen moving from non-cooperative DMPC to cooperating with one other agent. Thus, immediate benefits can be obtained without having to solve a problem comparable in size to a centralized problem.

6.2. *Vehicle guidance*

We consider a pair of vehicles, each modelled by the point unit mass dynamics $\dot{r}_p = v_p$ and $\dot{v}_p = f_p + d_p$, where $r_p \in \mathbb{R}^2, v_p \in \mathbb{R}^2$ represent, respectively, the position and velocity of vehicle $p \in \{1, 2\}$, and d_p is an additive disturbance to the control force $f_p \in \mathbb{R}^2$. These dynamics are discretized with a time step of 1.5 seconds to provide the linear, state-space model (1), with state $x_p = [r_p^{\mathrm{T}} \quad v_p^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^4$. The output constraints take the form of local speed and applied force limits: $\|v_p\|_2 \leq V_{\max}$ and $\|f_p\|_2 \leq F_{\max}$ respectively, where $V_{\max} = 0.225$ m/s and $F_{\max} = 0.08$ N. These 2-norm constraints may be approximated by polyhedra [25], with only small errors introduced. The disturbance is limited to 10% of the maximum control force, *i.e.* $\|d_p\|_\infty \leq 0.01 F_{\max}$.

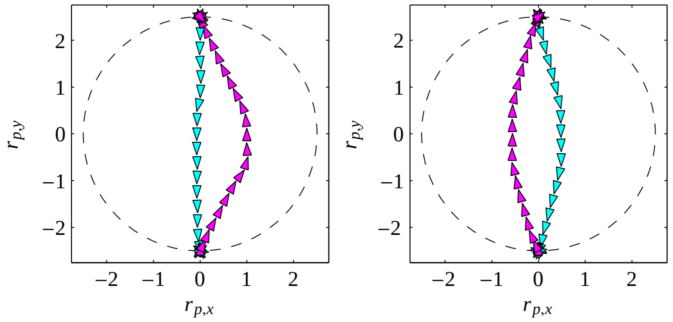


Figure 2: Position histories for two-vehicle problem; (left) non-cooperative DMPC, and (right) cooperative DMPC.

Coupling between vehicles arises from collision avoidance constraints, expressed as a minimum separation distance $L = 1$ m between each pair of vehicles: $\|r_1 - r_2\|_\infty \geq L$, resulting in a square exclusion region around each vehicle. These constraints are implemented as mixed integer linear inequalities using the 'big-M' approach [25, 26].

The feedback matrix K_p is the nilpotent controller for the system (A_p, B_p), and the set $\mathcal{R}_p$ is the corresponding minimal RPI set. Constraints are tightened accordingly, or in the case of the non-convex avoidance constraints, by enlargement of the excluded regions [23].

The two vehicles are required to traverse a 5 m diameter circle from opposing ends; a straight line path for both would lead to a collision. The objective for a vehicle p is to be steered close to a target state, a position $t_p = -r_p(0)$ where the velocity is zero. The *nominal* stage cost is $\|\bar{r}_p - t_p\|_2$. A polyhedral approximation to this 2-norm function is used [25], rendering the optimization objective (7) linear. The terminal set $\mathcal{X}_{F_p}$ for each vehicle is equal to the target state, and the terminal cost $F_p \triangleq 0$. Under these conditions, asymptotic convergence of the perturbed vehicles is guaranteed to the RPI set around the target state; that is, $[t_p^{\mathrm{T}} \quad 0^{\mathrm{T}}]^{\mathrm{T}} + \mathcal{R}_p$.

The initialization provided is sub-optimal, in that one of the vehicles is provided a straight-line plan, whilst for the other a deviated plan is formed to avoid collision. The update sequence subsequently employed is the simple, alternating sequence, so that vehicle agents optimize plans in sequence. Each vehicle is subjected to a sequence of random disturbances over the duration of the simulation. The horizon length is 25 steps.

Figure 2 shows the results. For non-cooperative DMPC, the vehicle travelling from North to South follows a desirable straight line path, leaving the other vehicle to deviate to avoid collision in the centre. The former has no incentive at any point to adopt a higher cost plan than the one it is following, or to make any allowances for the other vehicle. The cooperative control scheme delivers a more equal response: both vehicles deviate equally and oppositely to avoid collision. Thus, the initial sub-optimality is overcome.

7. Conclusions

In this paper, a cooperative distributed form of MPC has been presented. In the new algorithm, for LTI subsystems sharing coupling constraints, agents optimize plans locally and exchange information. A key finding is that coupled constraint satisfaction, achieved by permitting only one agent to optimize while 'freezing' the plans of others, is compatible with cooperation, achieved by considering wider objectives in the optimizations. Specifically, a local objective considers the weighted costs of other agents in the problem, and a local agent designs not only its own plan, but also hypothetical plans for other agents. The algorithm has been applied to example systems, including a vehicle guidance problem where 'greedy' behaviour leads to poor system-wide performance. It has been shown that the cooperative method has led to a more coordinated response and better global performance.

References

[1] D. Q. Mayne, J. B. Rawlings, C. V. Rao, P. O. M. Scokaert, Constrained model predictive control: Stability and optimality, Automatica 36 (2000) 789–814.

[2] J. M. Maciejowski, Predictive Control with Constraints, Prentice Hall, 2002.

[3] E. Camponogara, D. Jia, B. H. Krogh, S. Talukdar, Distributed model predictive control, IEEE Control Systems Magazine 22 (1) (2002) 44–52.

[4] R. Scattolini, Architectures for distributed and hierarchical model predictive control — a review, Journal of Process Control 19 (2009) 723–731.

[5] S. L. Waslander, G. Inalhan, C. J. Tomlin, Theory and Algorithms for Cooperative Systems, Vol. 4 of Series on Computers and Operations Research, World Scientific, 2004, Ch. Decentralized Optimization via Nash Bargaining, pp. 565–585, chapter 25.

[6] A. N. Venkat, I. A. Hiskens, J. B. Rawlings, S. J. Wright, Distributed MPC strategies with application to power system automatic generation control, IEEE Transactions on Control Systems Technology 16 (6) (2008) 1192–1206.

[7] J. B. Rawlings, B. T. Stewart, Coordinating multiple optimization-based controllers: New opportunities and challenges, Journal of Process Control 18 (2008) 839–845.

[8] T. Keviczky, B. Vanek, F. Borrelli, G. J. Balas, Hybrid decentralized receding horizon control of vehicle formations, in: Proceedings of the American Control Conference, 2006, pp. 3358–3363.

[9] R. L. Raffard, C. J. Tomlin, S. P. Boyd, Distributed optimization for cooperative agents: Application to formation flight, in: Proceedings of the 43rd IEEE Conference on Decision and Control, 2004, pp. 2453–2459.

[10] L. Pavel, An extension of duality to game-theoretic framework, Automatica 43 (2007) 226–237.

[11] G. Inalhan, D. M. Stipanović, C. J. Tomlin, Decentralized optimization, with application to multiple aircraft coordination, in: Proceedings of the 41st IEEE Conference on Decision and Control, 2002.

[12] B. T. Stewart, A. N. Venkat, J. B. Rawlings, S. J. Wright, G. Pannocchia, Cooperative distributed model predictive control, Systems & Control Letters 59 (2010) 460–469.

[13] Y. Kuwata, J. P. How, Cooperative distributed robust trajectory optimization using receding horizon MILP, IEEE Transactions on Control Systems Technology 19 (2) (2011) 423–431.

[14] T. Keviczky, F. Borrelli, G. J. Balas, Decentralized receding horizon control for large scale dynamically decoupled systems, Automatica 42 (12) (2006) 2105–2115.

[15] P. Trodden, A. Richards, Distributed model predictive control of linear systems with persistent disturbances, International Journal of Control 83 (8) (2010) 1653–1663.

[16] D. Q. Mayne, M. M. Seron, S. V. Raković, Robust model predictive control of constrained linear systems with bounded disturbances, Automatica 41 (2) (2005) 219–224.

[17] I. Kolmanovsky, E. G. Gilbert, Theory and computation of disturbance invariant sets for discrete-time linear systems, Mathematical Problems in Engineering 4 (1998) 317–367.

[18] M. Kvasnica, P. Grieder, M. Baotić, Multi-parametric toolbox (MPT), version 2.6.2 (December 2006).

[19] S. V. Raković, E. C. Kerrigan, K. I. Kouramas, D. Q. Mayne, Invariant approximations of the minimal robust positively invariant set, IEEE Transactions on Automatic Control 50 (4) (2005) 406–410.

[20] P. Trodden, A. Richards, Adaptive cooperation in robust distributed model predictive control, in: Proceedings of the IEEE Multi-conference on Systems and Control, 2009.

[21] A. Richards, J. P. How, Robust distributed model predictive control, International Journal of Control 80 (9) (2007) 1517–1531.

[22] W. B. Dunbar, Distributed receding horizon control of dynamically coupled nonlinear systems, IEEE Transactions on Automatic Control 52 (7) (2007) 1249–1263.

[23] Y. Kuwata, A. Richards, T. Schouwenaars, J. How, Distributed robust receding horizon control for multivehicle guidance, IEEE Transactions on Control Systems Technology 14 (4) (2007) 627–641.

[24] P. O. M. Scokaert, D. Q. Mayne, J. B. Rawlings, Suboptimal model predictive control (feasibility implies stability), IEEE Transactions on Automatic Control 44 (3) (1999) 648–654.

[25] A. Richards, J. P. How, Aircraft trajectory planning with collision avoidance using mixed integer linear programming, in: Proceedings of the American Control Conference, 2002, pp. 1936–1941.

[26] H. P. Williams, Logic and Integer Programming, International Series in Operations Research & Management Science, Springer, 2009.