2013 European Control Conference (ECC)
July 17-19, 2013, Zürich, Switzerland.

From Non-cooperative to Cooperative Distributed MPC: A Simplicial Approximation Perspective

Mathias Bürger, Giuseppe Notarstefano, and Frank Allgöwer

Abstract— **The paper deals with the coordination of dynamical systems by distributed model predictive control. We consider a set-up in which the subsystems dynamics are decoupled, while the subsystems outputs are coupled by some constraint. Starting from a well established non-iterative and non-cooperative architecture, we provide a novel interpretation for this non-cooperative scheme as a *simplicial approximation* of a convex program. Thanks to this novel interpretation, we are able to show why the existing algorithm, while guaranteeing feasibility, fails to compute an optimal solution to the centralized problem. Furthermore, by exploiting the simplicial approximation structure, we are able to propose a novel algorithm. The proposed algorithm inherits all the properties of the existing one, namely little communication, and feasibility. Furthermore, increasing the communication among the subsystems between two control updates improves the performance of the algorithm, regaining in the limit optimality in a cooperative sense.**

I. Introduction

A key challenge in modern control engineering is to control large-scale dynamical systems by taking control decisions only locally. In [1], a simple and elegant distributed model predictive control (MPC) algorithm for dynamical systems coupled through output constraints is proposed. The main difficulty is *"to ensure that the distributed decision making leads to actions that are consistent with the actions of others and satisfy the coupling constraints"* [1]. The algorithm proposed in [1] is *non-cooperative* and *non-iterative* in the sense that each agent solves a local optimal control problem and communicates with the next agent only once before each control update. In a sequential order the agents solve local optimal control problems given an estimate of the other outputs, and communicate the predicted trajectories to the next agent. The algorithmic idea proposed in [1], has been further investigated for problems with persistent disturbances [2] and for general cooperative control problems [3]. The algorithm in [1] is designed such that recursive feasibility is guaranteed. The main advantages are the low communication requirements and the compliance with the distributed problem structure. However, the algorithm is in general not able to obtain the performance of a centralized MPC. Cooperative MPC, e.g., [4], compute the central optimal solution, but require much communication and the coupling constraints are satisfied only in the limit.

In this paper, we first present an interpretation of the non-cooperative method of [1] as a simplicial approximation. We consider therefore the *dual* of the centralized optimal control problem, and show that the duals of the local sub-problem, solved by the algorithm of [1], can be interpreted as outer-approximation of the centralized dual problem. We show that storing one trajectory for each agent is not enough to compute a centralized solution. Based on these results we propose an alternative version of the algorithm, which keeps the advantageous distributed structure and can perform with the same communication requirements, but is additionally able to compute eventually the central optimal solution (if it is implemented in an iterative way). We allow the agents to store and exchange more than one trajectory per agent and adopt the structure of the local subproblems. Instead of considering the trajectories of other agents to be given, convex combinations of feasible output trajectories are considered. The novel MPC algorithm combines the advantageous properties of non-cooperative and cooperative algorithms: (i) it provides a feasible solution independent of how many local communication steps are performed (ii) and it will eventually converge to the central optimal solution.

The local sub-problems use simplicial approximations of the problem data, as known from *Dantzig-Wolfe* method [5], [6], see also [7]. However, in contrast to the original methods the novel approach has no master/sub-problem structure, but rather solves a sequence of optimization problems, where only parts of the problem are approximated. It is similar to the fully distributed polyhedral approximation method, which was recently proposed for general convex programs in [8] and previously for linear programs in [9].

The remainder of the paper is organized as follows. In Section II the non-iterative distributed MPC algorithm of [1] is reviewed. An alternative interpretation based on the dual problem is presented in Section III. In Section IV an extended version of the algorithm is proposed and the properties of the new algorithm are proven. Finally, some concluding remarks are given.

II. Problem Formulation

We consider a group of N dynamical systems, with each system $i \in \{1, \dots, N\}$ being governed by a linear time-invariant dynamics

$$x_i(t+1) = A_i x_i(t) + B_i u_i(t), \tag{1}$$

where $x_i(t) \in \mathbb{R}^{p_i}$, and $u_i(t) \in \mathbb{R}^{q_i}$. We assume that the states and the control inputs of the system are constrained to compact and convex sets containing the origin, i.e. $x_i(t) \in$

M. Bürger and F. Allgöwer thank the German Research Foundation (DFG) for financial support within the Cluster of Excellence in Simulation Technology (EXC 310/1) at the University of Stuttgart. The work of G. Notarstefano is partially supported by the project SOCIAL-ROBOTS under the program "5 per mille per la ricerca".

M. Bürger and F. Allgöwer are with the Institute for Systems Theory and Automatic Control, University of Stuttgart, Pfaffenwaldring 9, 70550 Stuttgart, Germany, {mathias.buerger, frank.allgower}@ist.uni-stuttgart.de.

Giuseppe Notarstefano is with the Department of Engineering, University of Lecce, Via per Monteroni, 73100 Lecce, Italy, giuseppe.notarstefano@unile.it.

$\mathcal{X}_i, \quad u_i(t) \in \mathcal{U}_i$ for all $t \geq 0$. In the rest of the paper we will denote the state and input trajectories over a finite time horizon $[t, t+T]$ with $\boldsymbol{x}_i = [x_i^\top(t), \ldots, x_i^\top(t+T)]^\top$, and $\boldsymbol{u}_i = [u_i^\top(t), \ldots, u_i^\top(t+T-1)]^\top$. Consistently, we will write $\boldsymbol{x}_i \in \mathcal{X}_i$ and $\boldsymbol{u}_i \in \mathcal{U}_i$ if we require the constraints to be satisfied at each time instant. We associate to each system an output

$$z_i(t) = C_i x_i(t) + D_i u_i(t), \tag{2}$$

with $z_i(t) \in \mathbb{R}^r$. The coupling among the systems is realized with the constraint

$$\sum_{i=1}^{N} z_i(t) \in \mathcal{S}_t, \quad \text{for all } t \geq 0, \tag{3}$$

where $\mathcal{S}_t$ is a convex set. For a time interval $[t, t+T]$, we will write in the following $\mathcal{S} = \mathcal{S}_t \times \cdots \times \mathcal{S}_{t+T-1}$ and $\boldsymbol{z}_i = [z_i^\top(t), \ldots, z_i^\top(t+T-1)]^\top$, the constraint can be rewritten in a more compact form as $\sum_{i=1}^{N} \boldsymbol{z}_i \in \mathcal{S}$.

Informally, the conceptual idea of model predictive control is to solve an optimal control problem over a finite horizon, and then to apply only the first piece of the computed input trajectory. Subsequently, a new optimization problem is formulated taking the realized system state as new initial condition. In the proposed distributed set-up, we assume each system to be assigned an infinite-horizon objective function penalizing states and inputs at each time step. Following the classical MPC idea, at each time t the cost functions are truncated at a finite time horizon $[t, t+T]$, leading to the *finite horizon* cost

$$\begin{aligned} F_i(x_i(t), \boldsymbol{u}_i) = \sum_{\tau=0}^{T-1} \ell_i(x_i(t+\tau|t), u_i(t+\tau|t)) \\ + V_i(x(t+T|t)), \end{aligned} \tag{4}$$

where $\ell_i(x_i(t+\tau|t), u_i(t+\tau|t))$ is called the stage-cost, and $V_i(x(t+T))$ the terminal cost. Taking a *centralized* perspective, the finite horizon optimal control problem, which involves the whole group of dynamical systems, is

$$\begin{aligned} \min \sum_{i=1}^{N} \Big(\sum_{\tau=0}^{T-1} \ell_i(x_i(t+\tau|t), u_i(t+\tau|t)) \\ + V_i(x(t+T|t)) \Big) \\ \sum_{i=1}^{N} z_i(t+\tau|t) \in \mathcal{S}_{t+\tau}, \\ x_i(t+\tau+1|t) = A_i x_i(t+\tau|t) + B_i u_i(t+\tau|t), \\ z_i(t+\tau|t) = C_i x_i(t+\tau|t) + D_i u_i(t+\tau|t), \\ x_i(t|t) = x_i(t), \ x_i(t+\tau|t) \in \mathcal{X}_i, \ u_i(t+\tau|t) \in \mathcal{U}_i \\ i \in \{1, \ldots, N\}, \ \tau \in \{0, \ldots, T-1\}. \end{aligned} \tag{5}$$

In a centralized set-up a central MPC controller would compute the solution to (5) at each time instant. However, if a central coordination is not applicable or is computationally too expensive, a distributed MPC scheme is required.

A. Sequential Distributed Model Predictive Control

A simple and elegant distributed MPC method was proposed in [1], and is reviewed here briefly. We assume that the agents are ordered sequentially according to their labels $i \in \{1, \ldots, N\}$ and are communicating according to this structure. The basic idea is that the centralized problem (5) is approximated by a sequence of local sub-problems. Each agent solves once per iteration a sub-problem, involving only its own control inputs as decision variables and fixing the other inputs to the estimates received by the former agent. We denote output trajectories predicted over the finite time horizon $[t, t+T]$ with $\bar{\boldsymbol{z}}_i = [\bar{z}_i^\top(t), \ldots, \bar{z}_i^\top(t+T-1)]^\top$. A set of feasible trajectories is denoted by $\mathbb{Z} = \{\bar{\boldsymbol{z}}_1, \ldots, \bar{\boldsymbol{z}}_N\}$.

Assumption 2.1: For any set $\mathbb{Z}$ of feasible output trajectories over the finite horizon $[t, t+T]$, a continuation of these trajectories $z_i(t+T)$, $i \in \{1, \ldots, N\}$ can be found, such that the new trajectories satisfy all systems constraints. □

Remark 2.2: There has been a major effort in the MPC literature to develop methods ensuring Assumption 2.1 to hold. We want to emphasize that this step is not in the focus of the present paper, and we refer the interested reader to the relevant literature, e.g., [1], [3], [10]. □

Now, at any time instant t, given a set of feasible predicted trajectories $\mathbb{Z}$, each system i can formulate a *local* optimization problem, $P_i^o(x_i(t), \boldsymbol{u}_i \,|\, \mathbb{Z})$, of the form

$$\begin{aligned} &\min_{\boldsymbol{u}_i, \boldsymbol{x}_i} \ F_i(x_i(t), \boldsymbol{u}_i) \\ &\text{subj. to } z_i(t+\tau|t) + \sum_{j \neq i} \bar{z}_j(t+\tau) \in \mathcal{S}_{t+\tau}, \\ &x_i(t+\tau+1|t) = A_i x_i(t+\tau|t) + B_i u_i(t+\tau|t), \\ &z_i(t+\tau|t) = C_i x_i(t+\tau|t) + D_i u_i(t+\tau|t), \\ &x_i(t|t) = x_i(t), \ x_i(t+\tau|t) \in \mathcal{X}_i, \ u_i(t+\tau|t) \in \mathcal{U}_i, \\ &\tau \in \{0, \ldots, T-1\}. \end{aligned} \tag{6}$$

Each problem $P_i^o(x_i(t), \boldsymbol{u}_i \,|\, \mathbb{Z})$ involves only the dynamic variables and the local constraint sets of agent i, whereas the trajectories of the other systems are fixed according to the set of feasible trajectories $\mathbb{Z}$. By construction, an optimal output trajectory $\boldsymbol{z}_i$ computed by $P_i^o(x_i(t), \boldsymbol{u}_i \,|\, \mathbb{Z})$, is such that set of trajectories $\mathbb{Z} = \{\bar{\boldsymbol{z}}_1, \ldots, \boldsymbol{z}_i, \ldots, \bar{\boldsymbol{z}}_N\}$ is compatible with the coupling constraint. The non-cooperative distributed MPC algorithm proposed in [1] is as follows:

Algorithm 1:

(i) *Set $t = 0$ and find a set of feasible trajectories $\mathbb{Z}$.*

(ii) *Apply the first element of the control inputs sequence corresponding to the output trajectories in $\mathbb{Z}$ for each subsystem and increment t,*

(iii) *For $i = 1, \ldots, N$*
Let agent i
 a) *receive the set of feasible output trajectories $\mathbb{Z}$;*
 b) *solve sub-problem $P_i^o(x_i(t), \boldsymbol{u}_i \,|\, \mathbb{Z})$ and compute the optimal output trajectory $\boldsymbol{z}_i$;*
 c) *update $\mathbb{Z}$ by replacing the $\bar{\boldsymbol{z}}_i$ with the newly computed $\boldsymbol{z}_i$.*

(iv) *Go to Step (ii).*

If Assumption 2.1 holds, i.e., a prediction method is known, this algorithm ensures recursive feasibility of all optimization problems and satisfaction of all constraints.

Concerning the distributed structure of the algorithm, the advantages of Algorithm 1 are that: (i) each agent has to solve only the sub-problem $P_i^o(x_i(t), \boldsymbol{u}_i \,|\, \mathbb{Z})$, involving only its own local constraint, and (ii) at every point in time a solution consistent with the constraints is known.

However, a drawback of Algorithm 1 is that in general it is not able to reproduce the performance of a centralized algorithm. In the next section we show that, even if we go from the sequential to an iterative implementation of the algorithm, the centralized optimal solution cannot be achieved. In other words, even if Step (iii) is performed repeatedly within one time step and agent N communicates the set of feasible trajectories to agent 1 such that the procedure can be repeated, one cannot expect this method to converge to the central optimal solution of (5). We say that the algorithm is not *consistent*.

III. Inner-Linearization Interpretation

We show next that sequential optimization can be interpreted as a *simplicial approximation method* of a semi-infinite convex program. For clarity of presentation we restrict the discussion to polyhedral coupling constraints

$$\sum_{i=1}^{N} \boldsymbol{z}_i \leq \mathrm{h} \tag{7}$$

for some $\mathrm{h} \in \mathbb{R}^{r \cdot T}$. We start with restating the centralized optimization problem (5) and the sub-problems (6) as convex programs. It is a standard result, that over a finite horizon the input trajectory of a linear (time-invariant) system determines the state and output trajectory in an affine manner, i.e., $\boldsymbol{x}_i = \Gamma_i \boldsymbol{u}_i + \beta_i x_{i0}$, and $\boldsymbol{z}_i = (C_i \Gamma_i + D_i)\boldsymbol{u}_i + C_i \beta_i x_{i0}$, where x_{i0} is the state of system i at initialization, and Γ_i, β_i are suitably defined. We can define the set of feasible output trajectories as

$$\mathcal{Z}_i := \Big\{ \boldsymbol{z}_i \in \mathbb{R}^{r \cdot T} \,|\, \exists \boldsymbol{u} \in \mathcal{U}_i, \text{s.t. } \Gamma_i \boldsymbol{u} + \beta_i x_{i0} \in \mathcal{X}_i, \\ \boldsymbol{z}_i = (C_i \Gamma_i + D_i)\boldsymbol{u} + C_i \beta_i \mathrm{x}_{i0} \Big\}.$$

Clearly, the sets $\mathcal{Z}_i$, $i \in \{1, \ldots, N\}$, are *convex*. In what follows, we consider the feasible output trajectories $\boldsymbol{z}_i$ as decision variables. We can associate to any feasible output trajectory $\boldsymbol{z}_i \in \mathcal{Z}_i$, at least one input trajectory $\boldsymbol{u}_i \in \mathcal{U}_i$ that generates it. In order to change the optimization variable, we associate to each output trajectory one input trajectory with minimal cost, and define the new objective function as $F_i(\boldsymbol{z}_i) = \min_{\boldsymbol{u}_i} F_i(x_{i0}, \boldsymbol{u}_i)$, under the constraint that $\boldsymbol{u}_i$ realizes the output trajectory $\boldsymbol{z}_i$. The function $F_i(\boldsymbol{z}_i)$ is clearly a convex function. The *centralized* finite horizon optimal control problem (5) can now be expressed as the convex program

$$\begin{aligned} P := \min_{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N} \; & \sum_{i=1}^{N} F_i(\boldsymbol{z}_i) \\ & \sum_{i=1}^{N} \boldsymbol{z}_i \leq \mathrm{h} \\ & \boldsymbol{z}_i \in \mathcal{Z}_i, \; i \in \{1, \ldots, N\}. \end{aligned} \tag{8}$$

The *local* sub-problems can be expressed in a similar way. Assuming an agent knows the set of feasible trajectories $\mathbb{Z} = \{\bar{\boldsymbol{z}}_1, \ldots, \bar{\boldsymbol{z}}_N\}$ with $\boldsymbol{z}_j \in \mathcal{Z}_j$, the local sub-problem (6) of agent i can be expressed as[1]

$$\begin{aligned} P_i^o(\mathbb{Z}) := \min_{\boldsymbol{z}_i} \quad & F_i(\boldsymbol{z}_i) + \sum_{j \neq i} F_j(\bar{\boldsymbol{z}}_j) \\ & \boldsymbol{z}_i + \sum_{j \neq i} \bar{\boldsymbol{z}}_j \leq \mathrm{h}, \; \boldsymbol{z}_i \in \mathcal{Z}_i, \end{aligned} \tag{9}$$

where we added here with respect to (6) the constants terms $\sum_{j \neq i} F_j(\bar{\boldsymbol{z}}_j)$ to the objective function. Presenting the sub-problem in the structure (9) will lead the way to an alternative interpretation of the algorithm.

Dual Interpretation of the Centralized Problem:

To gain more insights into the relation between the centralized problem and the local sub-problems, it is worthwhile to consider their *dual problems*. We assume that there is a set of feasible trajectories $\bar{\boldsymbol{z}}_1, \ldots, \bar{\boldsymbol{z}}_N$, each in the relative interior of $\mathcal{Z}_i$, $i \in \{1, \ldots, N\}$, such that the linear coupling constraint holds strictly. This reasonable assumption, is Slater's condition and ensures that strong duality holds.

Let $\boldsymbol{\mu} \in \mathbb{R}^{r \cdot T}$ be the Lagrange multiplier for the coupling constraint, then the dual problem to (8) can be expressed as

$$\begin{aligned} & \max_{\boldsymbol{\mu} \geq 0} \min_{\boldsymbol{z}_1 \in \mathcal{Z}_1, \ldots, \boldsymbol{z}_N \in \mathcal{Z}_N} \; \sum_{i=1}^{N} F_i(\boldsymbol{z}_i) + \boldsymbol{\mu}^\top \left(\sum_{i=1}^{N} \boldsymbol{z}_i - \mathrm{h} \right) \\ \Leftrightarrow \quad & \max_{\boldsymbol{\mu} \geq 0} \; -\mathrm{h}^\top \boldsymbol{\mu} + \sum_{i=1}^{N} \min_{\boldsymbol{z}_i \in \mathcal{Z}_i} \; \big(F_i(\boldsymbol{z}_i) + \boldsymbol{\mu}^\top \boldsymbol{z}_i \big) \\ \Leftrightarrow \quad & \max_{\boldsymbol{\mu} \geq 0, \nu_1, \ldots, \nu_N} \; -\mathrm{h}^\top \boldsymbol{\mu} + \sum_{i=1}^{N} \nu_i \\ & \quad \text{subj. to} \quad \nu_i \leq \min_{\boldsymbol{z}_i \in \mathcal{Z}_i} F_i(\boldsymbol{z}_i) + \boldsymbol{\mu}^\top \boldsymbol{z}_i \\ & \quad i \in \{1, \ldots, N\}. \end{aligned}$$

Equivalently, the dual problem may be stated as the *linear semi-infinite optimization problem*

$$\begin{aligned} D := \max_{\boldsymbol{\mu} \geq 0, \nu_1, \ldots, \nu_N} \; & -\mathrm{h}^\top \boldsymbol{\mu} + \sum_{i=1}^{N} \nu_i \\ \text{subj. to } \; & \nu_i \leq F_i(\boldsymbol{z}_i) + \boldsymbol{\mu}^\top \boldsymbol{z}_i, \; \text{for all } \boldsymbol{z}_i \in \mathcal{Z}_i \\ & i \in \{1, \ldots, N\}, \end{aligned} \tag{10}$$

We denote D the optimal value of (10). Since, by assumption, Slater's condition holds, we have strong duality and thus $D = P$. The dual problem (10) is a linear optimization problem with $\delta = N + r \cdot T$ *decision variables*, and N *constraints*, each parameterized by the feasible trajectories of the corresponding system. In fact, due to this parameterization, problem (10) has an infinite number of linear constraints. Given a set of feasible trajectories $\mathbb{Z} = \{\bar{\boldsymbol{z}}_1^1, \ldots, \bar{\boldsymbol{z}}_1^{L_1}, \ldots, \bar{\boldsymbol{z}}_N^1, \ldots, \bar{\boldsymbol{z}}_N^{L_N}\}$ we can formulate a

[1] We use from here on the short hand notation $P_i^o(\mathbb{Z})$ instead of $P_i^o(x_i(t), \boldsymbol{u}_i \,|\, \mathbb{Z})$

relaxed dual problem as

$$
\begin{aligned}
D(\mathbb{Z}) := \max_{\boldsymbol{\mu} \geq 0, \nu_1, \ldots, \nu_N} \; & -\mathrm{h}^\top \boldsymbol{\mu} + \sum_{i=1}^{N} \nu_i \\
\text{s.t. } & \nu_i \leq F_i(\bar{\boldsymbol{z}}_i^l) + \boldsymbol{\mu}^\top \bar{\boldsymbol{z}}_i^l, \\
& l \in \{1, \ldots, L_i\}, \;\; i \in \{1, \ldots, N\}.
\end{aligned} \tag{11}
$$

The relaxed dual problem is a finite-dimensional linear program with δ decision variables and one linear constraint for each element of $\mathbb{Z}$. The finite-dimensional constraints in (11) are a relaxation of the semi-infinite constraints in (10), and we have $D(\mathbb{Z}) \geq D = P$. Now, the following result is a direct consequence of Theorem 4.2 in [11].

Theorem 3.1 (Finite Approximation): There exists a set of trajectories $\mathbb{Z} = \{\boldsymbol{z}_1^1, \ldots, \boldsymbol{z}_1^{L_1}, \ldots, \boldsymbol{z}_N^1, \ldots, \boldsymbol{z}_N^{L_N}\}$ with $\sum_{i=1}^{N} L_i = \delta \; \left(\delta = N + r \cdot T\right)$ such that $D(\mathbb{Z}) = D$. □

Dual Interpretation of the Local Sub-problems

We are now ready to consider the dual problems to the *local* sub-problems (9). As for the centralized problem, we formulate the dual problem to the local sub-problem $P_i^o(\mathbb{Z})$, defined for a set of trajectories $\mathbb{Z}$, as

$$
\begin{aligned}
\max_{\boldsymbol{\mu} \geq 0} \min_{\boldsymbol{z}_i \in \mathcal{Z}_i} \; & F_i(\boldsymbol{z}_i) + \boldsymbol{\mu}^\top \boldsymbol{z}_i \\
& + \left(\sum_{j \neq i} F_j(\bar{\boldsymbol{z}}_j) + \boldsymbol{\mu}^\top \bar{\boldsymbol{z}}_j \right) - \mathrm{h}^\top \boldsymbol{\mu}.
\end{aligned}
$$

This dual problem can be rewritten as

$$
\begin{aligned}
D_i(\mathbb{Z}) := \max_{\boldsymbol{\mu} \geq 0, \nu_1, \ldots, \nu_n} \; & -\mathrm{h}^\top \boldsymbol{\mu} + \sum_{j=1}^{N} \nu_i \\
& \nu_i \leq F_i(\boldsymbol{z}_i) + \boldsymbol{\mu}^\top \boldsymbol{z}_i, \text{ for all } \boldsymbol{z}_i \in \mathcal{Z}_i \\
& \nu_j \leq F_j(\bar{\boldsymbol{z}}_j) + \boldsymbol{\mu}^\top \bar{\boldsymbol{z}}_j, \text{ for all } j \neq i.
\end{aligned} \tag{12}
$$

We are at a key point in our analysis. We have just shown that problem (12), which is the dual of the local sub-problem (9) and thus of (6), is a polyhedral outer approximation of (10), which is the dual of the centralized problem (8) and thus of (5). Only the i-th constraint is kept as a semi-infinite constraint, while all other constraints are approximated by a *single* linear constraint. We use the notational convention that $D_i(\mathbb{Z})$ denotes the dual sup-problem of agent i, having only one semi-infinite constraint plus a set of linear constraints, while $D(\mathbb{Z})$ denotes the approximation of the centralized dual problem with only finite constraints.

This optimization interpretation allows us to show why, in general, Algorithm 1 cannot recapture the performance of the centralized solver, even if implemented in an iterative way. Specifically, each set of N feasible trajectories $\mathbb{Z}$, leads to a relaxed dual problem (11) with N linear constraints. However, it is known from Theorem 3.1 that, in general, $\delta = N + r \cdot T$ constraints are needed to recover an optimal solution. This observation suggests also the trail to design a new distributed MPC scheme that recovers consistency when implemented in an iterative way. We start by exploiting the structure of the local sub-problems, $D_i(\mathbb{Z})$, if the set of feasible trajectories does not contain only N trajectories, but more generally $\mathbb{Z} = \{\bar{\boldsymbol{z}}_1^1, \ldots, \bar{\boldsymbol{z}}_1^{L_1}, \ldots, \bar{\boldsymbol{z}}_N^1, \ldots, \bar{\boldsymbol{z}}_N^{L_N}\}$. The local sub-problems $D_i(\mathbb{Z})$ are now in the form (12), with one semi-infinite constraint (i.e. the i-th constraint) and $\sum_{j \neq i} L_j$ linear constraints.

We can use again duality to understand the role of this problem. Using standard linear programming duality with Lagrange multipliers $\lambda_{jl} \in \mathbb{R}$ for each linear constraint leads directly to the following primal representation of the problem

$$
\begin{aligned}
P_i(\mathbb{Z}) := \min_{\boldsymbol{z}_i, \lambda_{jl} \geq 0} \; & F_i(\boldsymbol{z}_i) + \sum_{j \neq i} \sum_{l=1}^{L_j} F_j(\bar{\boldsymbol{z}}_j^l) \lambda_{jl} \\
& \boldsymbol{z}_i + \sum_{j \neq i} \sum_{l=1}^{L_j} \bar{\boldsymbol{z}}_j^l \lambda_{jl} \leq \mathrm{h} \\
& \boldsymbol{z}_i \in \mathcal{Z}_i, \; \sum_{l=1}^{L_j} \lambda_{jl} = 1, \text{ for all } j \neq i.
\end{aligned} \tag{13}
$$

Note that we denote $P_i(\mathbb{Z})$ the local approximate primal problem with an arbitrary number of trajectories, as opposed to the original local subproblem $P_i^o(\mathbb{Z})$. We can give now the following interpretation of the approximate sub-problems in the primal form $P_i(\mathbb{Z})$: all parts of the problem related to agent i are maintained as in their original representation, while a *simplicial approximation* is used for the problem data related to the other agents $j \neq i$.

IV. A Consistent Distributed MPC Algorithm

We can derive now a novel distributed MPC algorithm. The key difference with Algorithm 1 is that the set $\mathbb{Z}$, defining the local sub-problems, may contain more than N predicted trajectories (rather than exactly N). Also, we assume that, together with a predicted trajectory $\bar{z}_j^l$, the value of the corresponding cost, i.e., $F_j(\bar{\boldsymbol{z}}_j^l)$, is stored and exchanged. Thus, given a set of trajectories $\mathbb{Z}$, each system can formulate its *local* sub-problem $P_i(\mathbb{Z})$, in the explicit MPC formulation, as

$$
\begin{aligned}
& P_i(\mathbb{Z}) = \min_{\boldsymbol{u}_i, \boldsymbol{x}_i, \lambda_{jl}} \; F_i(x_i(t), \boldsymbol{u}_i) + \sum_{j \neq i} \sum_{l=1}^{L_j} F_j(\bar{\boldsymbol{z}}_j^l) \lambda_{jl} \\
& z_i(t+\tau|t) + \sum_{j \neq i} \sum_{l=1}^{L_j} \bar{z}_j^l(\tau) \lambda_{jl} \leq \mathrm{h}(\tau), \; \tau \in \{0, \ldots, T-1\} \\
& \sum_{l=1}^{L_j} \lambda_{jl} = 1, \; \lambda_{jl} \geq 0, \; j \neq i, \\
& x_i(t+\tau+1|t) = A_i x_i(t+\tau|t) + B_i u_i(t+\tau|t), \\
& z_i(t+\tau|t) = C_i x_i(t+\tau|t) + D_i u_i(t+\tau|t), \\
& x_i(t|t) = x_i(t), \quad x_i(t+\tau|t) \in \mathcal{X}_i, u_i(t+\tau|t) \in \mathcal{U}_i.
\end{aligned} \tag{14}
$$

Clearly, problem (14) is equivalent to problem (13). The local sub-problems (14) are finite-horizon optimal control problems over the control inputs of agent i, plus a linear component, with which agent i adjusts the predictions of the other agents. Since each agent i has no knowledge about the dynamic models or local constraints of the other agent, it considers only convex combinations of the feasible output trajectories of the other agents. We formalize this aspect, ensuring that the solutions to the sub-problems satisfy all systems constraints, i.e., the local and the coupling constraints. feasibility of the solutions to $P_i(\mathbb{Z})$.

Lemma 4.1 (Feasibility): Consider the sub-problem $P_i(\mathbb{Z})$ and let $\boldsymbol{z}_i$ and $\lambda_{jl}, l \in \{1, \ldots, L_j\}, j \neq i$, be an optimal solution. Define

$$\boldsymbol{z}_j = \sum_{l=1}^{L_j} \bar{\boldsymbol{z}}_j^l \lambda_{jl}, \; \forall j \neq i. \tag{15}$$

Then $\boldsymbol{z}_j \in \mathcal{Z}_j$, for all $j \in \{1, \ldots, N\}$, and $\sum_{j=1}^{N} \boldsymbol{z}_j \leq \mathrm{h}$.

Proof: The proof follows directly since all $\mathcal{Z}_j$ are convex sets and $\sum_{j=1}^{L_j} \lambda_{jl} = 1$, $\lambda_{jl} \geq 0$. ■

This result shows that we can always use (15) to *reconstruct a feasible solution* to the centralized problem (5) from the solution of the local sub-problem $P_i(\mathbb{Z})$. We propose now the following distributed MPC algorithm.

Algorithm 2:

(i) *Set $t = 0$ and find a set of feasible trajectories $\mathbb{Z}$. Set K to the allowed number of communications within each time step.*
(ii) *Apply the first element of the control inputs sequence corresponding to the output trajectories in $\mathbb{Z}$ for each subsystem and increment t.*
(iii) *Receive the set of feasible output trajectories $\mathbb{Z}$ and set $\hat{\mathbb{Z}}(0) := \mathbb{Z}$;*
(iv) *For $k = 0, 1, \ldots, K-1$.*
Set $i = \mathrm{mod}(k, N) + 1$ and let agent i:
  a) *receive $\hat{\mathbb{Z}}(k)$;*
  b) *solve sub-problem $P_i(\hat{\mathbb{Z}}(k))$ and compute an optimal trajectory $\boldsymbol{z}_i$;*
  c) *update $\hat{\mathbb{Z}}(k+1) := \hat{\mathbb{Z}}(k) \cup \{\boldsymbol{z}_i\}$.*
(v) *Reconstruct from $\hat{\mathbb{Z}}(K)$ the set of feasible trajectories $\mathbb{Z}$ and transmit it to all agents.*
(vi) *Go to Step (ii).*

This algorithm inherits the advantageous properties of the original sequential MPC algorithm: (i) each agent has to solve only a finite-horizon optimal control problem involving its local dynamics, (ii) at any iteration of the local computations a centrally feasible solution is known.

Proposition 4.2 (Recursive Feasibility): Let Assumption 2.1 hold and assume a set of feasible trajectories can be found at time $t = 0$. For any number of allowed local communications K, the closed-loop solutions realized by Algorithm 2 satisfy the constraints at all times.

Proof: By assumption, a feasible output and corresponding input trajectory is known for the time horizon $\tau = 0, \ldots, T$. The statement can now be derived by induction over t. Let at time t a feasible trajectory for the time horizon $\tau = t, \ldots, t+T$ be known. By Assumption 2.1, at time $t+1$ feasible trajectory for the time horizon $\tau = t+1, \ldots, t+T+1$ can be determined. Now, independent of the choice of K, the solution reconstructed from the local solutions in Step (v) is feasible for the centralized problem (Theorem 4.1). ■

Note that the recursive feasibility depends mainly on two properties: (i) the ability to predict a feasible trajectory at each time step (this is not in the scope of the present paper) and (ii) the central feasibility of the solutions to the local sub-problems (this is the main scope of the present paper). Algorithm 2 thus maintains the desired MPC properties of Algorithm 1, but is also *consistent*. That is, the algorithm recaptures the performance of the centralized MPC algorithm for sufficiently large K.

Next, we show that it is possible to compare the values of two subsequent sub-problems, although their objective functions are differently defined. From now on, we use the short hand notation $P_i(k)$ instead of $P_i(\hat{\mathbb{Z}}(k))$ and we implicitly assume that i refers always to the active agent at iteration k, i.e., $i = \mathrm{mod}(k, N) + 1$.

Theorem 4.3 (Monotonicity): At each time instant t, let P be the value of the centralized optimization problem. For all $k \geq 0$, it holds that $P_i(k) \geq P_i(k+1) \geq P$.

Proof: Consider an arbitrary iteration k and let i be the corresponding agent solving (14). Let $\boldsymbol{z}_i$, and $\lambda_{jl}, l \in \{1, \ldots, L_j\}, j \neq i$ be an optimal solution to (14). The next agent, say agent $i+1$, recieves $\hat{\mathbb{Z}}(k+1) = \hat{\mathbb{Z}}(k) \cup \{\boldsymbol{z}_i\}$. A feasible solution to the sub-problem (14) of agent $i+1$ is now as follows: all λ_{jl} for $j \neq i, i+1$, are chosen as computed by agent i in the previous step; for $j = i$ the single λ_{il} corresponding to the previously computed trajectory is chosen to be one, and all other λ_{il} are set to zero; for $j = i+1$ the decision variable $\boldsymbol{z}_{i+1} = \sum_{l=1}^{L_{i+1}} \bar{\boldsymbol{z}}_{i+1} \lambda_{i+1,l}$, with $\lambda_{i+1,l}$ as computed by agent i in the previous iteration.

By the convexity of the cost function, we have $\sum_{l=1}^{L_{i+1}} F_{i+1}(\bar{\boldsymbol{z}}_{i+1}) \lambda_{i+1,l} \geq F_{i+1}(\boldsymbol{z}_{i+1})$. Thus, there is a feasible solution to the sub-problem of agent $i+1$ with a cost not exceeding $P_i(k)$, and we can conclude $P_i(k) \geq P_i(k+1)$, for all $k \geq 0$. Since the constructed solution is feasible for the centralized problem, we we also have $P_i(k) \geq P$, for all $k \geq 0$. ■

The values of the local sub-problems are monotonically non-increasing and the longer the local computations can be performed, i.e., the larger K, the better will be the final cost.

Theorem 4.4 (Consistency): Consider an arbitrary time instant t. Let P be the optimal value of the centralized finite horizon optimal control problem (5). Suppose $K \to \infty$. Then, $\lim_{k \to \infty} P_i(k) \to P$.

Proof: We have $P_i(k) \geq P$ for all $k \geq 0$. Since $P_i(k)$ is monotonically non-increasing and bounded, it converges. Corresponding to each $P_i(k)$ we have can formulate the dual problem $D_i(k)$ with the dual solutions $\boldsymbol{\mu}(k)$ and $\nu_i(k)$. The dual solution is not necessarily unique, and we can select any of the possible candidates. For any dual solution $\boldsymbol{\mu}(k)$ to $D_i(k)$, we have

$$\sum_{i=1}^{N} \min_{\boldsymbol{z}_i \in \mathcal{Z}_i} \left(F_i(\boldsymbol{z}_i) + \boldsymbol{\mu}^\top(k) \boldsymbol{z}_i \right) - \mathrm{h}^\top \boldsymbol{\mu}(k) \leq P,$$

since $\boldsymbol{\mu}(k)$ is an infeasible solution to the central dual problem D. Strict duality has to hold also at each iteration, and we have

$$\sum_{i=1}^{N} \nu_i(k) - \mathrm{h}^\top \boldsymbol{\mu}(k) = P_i(k).$$

We can define now

$$\Delta(k) := \sum_{i=1}^{N} \min_{\boldsymbol{z}_i \in \mathcal{Z}_i} \left(F_i(\boldsymbol{z}_i) + \boldsymbol{\mu}^\top(k) \boldsymbol{z}_i - \nu_i(k) \right).$$

It follows directly that

$$\Delta(k) + P_i^\star(k) \le P^\star \le P_i^\star(k). \tag{16}$$

We show next that $\lim_{k\to\infty} \Delta(k) \to 0$. Any dual solution feasible for $D_i(k)$ must be such that

$$0 \le F_i(\boldsymbol{z}_j^l) + \boldsymbol{\mu}^\top(k)\boldsymbol{z}_j^l - \nu_i(k),$$

where $\boldsymbol{z}_j^l$, with $l \in \{1,\dots,L_j\}, j \in \{1,\dots,N\}$, are all trajectories computed up to iteration k. At each iteration, a finite constraint is added in the dual and all previous dual constraints are kept. Thus, it holds that $\Delta(k) \le \Delta(k+1)$. Now, at any iteration k with "active" agent i, let $\boldsymbol{z}_i^k$ be the minimizer of

$$\min_{z_i \in \mathcal{Z}_i} F_i(\boldsymbol{z}_i) + \boldsymbol{\mu}^\top(k)\boldsymbol{z}_i - \nu_i(k).$$

If

$$F_i(\boldsymbol{z}_i^k) + \boldsymbol{\mu}^\top(k)\boldsymbol{z}_i^k - \nu_i(k) < 0 \tag{17}$$

it must hold that

$$F_i(\boldsymbol{z}_i^k) + \boldsymbol{\mu}^\top(k')\boldsymbol{z}_i^k - \nu_i(k') \ge 0 \tag{18}$$

for all $k' > k$. Since $P_i(k)$ is converging, also $D_i(k)$ is converging. We have at each iteration k possibly several dual optimal solutions. However, since the dual solutions are never explicitly computed, we can select at each iteration one dual optimal solution such that the resulting sequence is converging. Now, as the sequence of dual solutions is converging, for sufficiently large k the difference between (17) and (18) must go to zero. Thus, for any $\epsilon > 0$, there is a k_ϵ such that for all $k \ge k_\epsilon$

$$-\epsilon \le \min_{z_i \in \mathcal{Z}_i} F_i(\boldsymbol{z}_i) + \boldsymbol{\mu}^\top(k)\boldsymbol{z}_i - \nu_i(k) \le 0, \tag{19}$$

holds for all $i \in \{1,\dots,N\}$. Since ϵ was chosen arbitrarily, we can conclude that $\lim_{k\to\infty} \Delta(k) \to 0$. Now, the last conclusion together with (16) proves the statement. ∎

It is worth to notice, that the proposed scheme is particularly efficient for polyhedral problems, where a finite number of communications suffices to obtain the central optimal solution. Note that this includes MPC problems with a stage cost defined by the 1-norm or the ∞-norm.

Theorem 4.5 (Polyhedral Systems): Assume all local constraint sets $\mathcal{X}_i$ and $\mathcal{U}_i$ are polytopes and all objective functions are piecewise linear. Then there is a finite number k^* such that $P_i(k^*) = P$.

Proof: Consider the centralized dual problem D in the form (10). The set of feasible dual solutions is defined by the N semi-infinite constraints. Since the sets $\mathcal{Z}_i$ are polytopic sets, the set of centralized feasible dual solution can be equivalently described by a *finite* number of linear constraints: one for each extreme point of $\mathcal{Z}_i,\ i \in \{1,\dots,N\}$ and one for each break point of the objective functions. Now, the local sub-problems are linear programs. The proof proceeds now as the proof of Theorem 4.4, noting additionally that after a finite number of iterations we have $\Delta(k) = 0$. ∎

The novel Algorithm 2 relies strongly on the simplicial approximation. Each agent uses this simplicial approximation to overcome its lack of knowledge about the other agents dynamics. The "breaking points", at which the approximation is defined, are exchanged between the agents and updated at each iteration. Such a simplicial approximation of convex optimization problems is well known from the classical *Dantzig-Wolfe (DW) method* for nonlinar problems [5], [6], [12]. However, the classical DW-method leads to a structure with one linear master program and several several nonlinear sub-problems. The nonlinear sub-problems are solved to add "breaking points" to the approximation. Contrasting this, the set-up here does not lead to a master/sub-problem structure, but uses a sequence of optimization problems, where only parts of the optimization problem are approximated. Thus, the iterative method proposed for the local computation can be seen as a novel, fully distributed optimization algorithm.

V. Conclusions

We presented a novel interpretation of the basic distributed MPC algorithm by Richards and How [1] as a simplicial approximation method. This interpretation provides an explanation, why the algorithm cannot derive the centralized optimal solution. Building upon this result, we proposed a novel version of the distributed MPC algorithm, which keeps the advantageous properties of the original non-cooperative algorithm, but is cooperative as it can compute the centralized solutions. The novel MPC algorithm ensures recursive feasibility and converges to the central optimal solution if the local computations are performed repeatedly. We used heavily the classical idea of simplicial approximation, and we believe that this conceptual idea opens ways for further advances on distributed model predictive control.

References

[1] A. Richards and J. P. How, "Robust distributed model predictive control," *International Journal of Control*, vol. 80, no. 9, pp. 1517 – 1531, 2007.

[2] P. Trodden and A. Richards, "Distributed model predictive control of linear systems with persistent distrubances," *International Journal of Control*, vol. 83, no. 8, 2010.

[3] M. A. Müller, M. Reble, and F. Allgöwer., "Cooperative control of dynamically decoupled systems via distributed model predictive control." *Int. J. Robust Nonlinear Control*, vol. 22, no. 12, pp. 1376–1397, 2012.

[4] P. Giselsson and A. Rantzer, "Distributed model predictive control with suboptimality and stability guarantees," in *IEEE Conference on Decision and Control*, Atlanta, USA, Dec. 2010, pp. 7272 – 7277.

[5] G. B. Dantzig, *Linear Programming and Extensions*. Princeton University Press, 1963.

[6] L. S. Lasdon, *Optimization Theory for Large Scale Systems*. Courier Dover Publications, 2002.

[7] D. P. Bertsekas and H. Yu, "A unifying polyhedral approximation framework for convex optimization," *SIAM Journal on Optimization*, vol. 21, no. 1, pp. 333–360, 2011.

[8] M. Bürger, G. Notarstefano, and F. Allgöwer, "A polyhedral approximation framework for convex and robust distributed optimization." *IEEE Transactions on Automatic Control*, 2012, conditionally accepted.

[9] ——, "Locally constrained decision making via two-stage distributed simplex," in *Proc. 50th IEEE Conf. Decision and Control, European Control Conf.*, Orlando, Florida, December 2011, pp. 5911 – 5916.

[10] L. Grüne and K. Worthmann, *Distributed Decision Making and Control*, ser. Lecture Notes in Control and Information Sciences. Springer, 2012, ch. A Distributed NMPC Scheme without Stabilizing Terminal Constraints, pp. 261–287.

[11] R. Hettich and K. Kortanek, "Semi-infinite programming: Theory, methods and applications," *SIAM Review*, vol. 35, no. 3, pp. 380 – 429, Sep. 1993.

[12] Y. Sekine, "Decentralized optimization of an interconnected system," *IEEE Transactions on Circuit Theory*, vol. 10, no. 2, pp. 161 – 168, 1963.