Design of plug-and-play model predictive control: an approach based on linear programming

Stefano Riverso, Graduate Student Member, IEEE, Marcello Farina and Giancarlo Ferrari-Trecate, Senior Member, IEEE

Abstract—In this paper we consider a linear system represented by subsystems coupled through states and present a distributed control scheme for guaranteeing asymptotic stability and satisfaction of constraints on system inputs and states. The design procedure enables Plug-and-Play (PnP) operations, meaning that (i) the addition or removal of subsystems triggers the synthesis of local controllers associated to successors to the subsystem only and (ii) the synthesis of a local controller for a subsystem requires information only from predecessors of the subsystem and it can be performed using only local computational resources. Our method, that is based on Model Predictive Control (MPC), advances the PnP design procedure proposed in [1] in several directions. Notably, we show how critical steps in the design of a local controller can be solved through linear programming.

I. INTRODUCTION

The ever-increasing complexity and size of process plants, manufacturing systems, transportation systems and power distribution networks, call for the development of decentralized and distributed control architectures, where decisions are taken in parallel by a number of local regulators that can be collocated with actuators. Decentralized and distributed control schemes have a long tradition [2] and are usually based on specific models requiring to split the overall plant into interacting subsystems. In this paper we assume subsystems are coupled through state variables only and we say that subsystem $j$ is a predecessor (respectively, a successor) of subsystem $i$ if its state variables directly influence the dynamics of $i$ (respectively, if its dynamics is directly influenced by the state variables of $i$).

In the last years, many Decentralized/Distributed control schemes based on MPC (De/DiMPC) have been proposed [3], [4], [5], [6], [7]. One of the main problems of existing De/DiMPC approaches is the need of a centralized off-line design phase, meaning that controller synthesis requires the knowledge of the whole system. In the context of large-scale systems, this can be a severe limitation because a global model of the system can be very hard or costly to obtain. Even worse, in several examples of cyber-physical systems [8] units can enter and leave a network over time making impractical to retune the overall controller in a centralized fashion. In these cases, decentralized design based on local computational resources is the only viable approach. Accordingly, in [1] a novel solution based on the PnP design paradigm [9] has been proposed. PnP design allows synthesis decentralization and it has the following features: when a subsystem is added to an existing plant (i) local controllers have to be designed only for the subsystem and its successors; (ii) the design of a local controller uses only information from the subsystem and its predecessors. Quite remarkably, these requirements allow local controllers to be computed using computational resources collocated with subsystems. Furthermore, the complexity of controller design and implementation, for a given subsystem, scales with the number of predecessor subsystems only. We highlight that, in our approach, addition and removal of subsystems are performed offline and they do not trigger hybrid dynamics.

As in [1], we propose a PnP design procedure hinging on the notion of tube MPC. In [1] the most critical step in the design of local MPC controllers requires to solve a nonlinear optimization problem. In this paper, using local tube MPC regulators based on Robust Control Invariant (RCI) sets [10], [11], we guarantee overall stability and constraints satisfaction solving Linear Programming (LP) problems only. Another difference is that in [1] stability requirements are fulfilled imposing an aggregate sufficient small-gain condition for networks. In the present paper, we resort instead to set-based conditions that are usually less conservative. Finally, while methods in [1] are tailored to decentralized control only, the new PnP-DeMPC scheme also admits a distributed implementation (see Section V of [12]). As for any decentralized synthesis procedure, our method involves some degree of conservativeness [13], i.e. it implicitly assumes that coupling between subsystems is not too strong.

The paper is structured as follows. The design of decentralized controllers is introduced in Section II. In Section III we discuss how to design local controllers by solving LP problems and we describe PnP operations. In Section IV we present an applicative example and Section V is devoted to concluding remarks.

Notation. We use $a : b$ for the set of integers $\{a, a + 1, \ldots, b\}$. The column vector with $s$ components $v_1, \ldots, v_s$ is $v = (v_1, \ldots, v_s)$. The function $\text{diag}(G_1, \ldots, G_s)$ denotes the block-diagonal matrix composed by $s$ block $G_i$, $i \in 1 : s$. The symbols $\oplus$ and $\ominus$ denote the Minkowski sum and difference, respectively, i.e. $A = B \oplus C$ if $A = \{a : a =
will use the notation \( x_1 \) is given by

Similarly, the input is composed by \( w \) and \( u \)

Consistently with (1), the dynamics of the \( i \)-th subsystem is given by

\[
\Sigma_i: \quad x^+_i = A_{ij} x_i + B_i u_i + w_i \\
w_i = \sum_{j \in N_i} A_{ij} x_j + u_j
\]

where \( A_{ij} \in \mathbb{R}^{n_i \times n_j}, i, j \in M, B_i \in \mathbb{R}^{n_i \times m_i} \) and \( N_i \) is the set of predecessors to subsystem \( i \) defined as \( N_i = \{ j \in M : A_{ij} \neq 0, i \neq j \} \).

**Assumption 1:** The matrix pair \( (A_{ii}, B_i) \) is controllable, \( \forall i \in M \).

We equip subsystems \( \Sigma_i, i \in M \) with the constraints \( x_i \in \mathcal{X}_i, u_i \in \mathcal{U}_i \), define the sets \( \mathcal{X} = \prod_{i \in M} \mathcal{X}_i, \mathcal{U} = \prod_{i \in M} \mathcal{U}_i \) and equip (1) with

\[
x \in \mathcal{X}, \ u \in \mathcal{U}.
\]

**Assumption 2:** Constraints \( \mathcal{X}_i \) and \( \mathcal{U}_i \) are compact and convex polytopes containing the origin in their nonempty interior.

Next, we propose a decentralized controller for (1) guaranteeing asymptotic stability of the origin and constraints satisfaction. In the spirit of tube-based control [10], we treat \( w_i \) as a disturbance and define the nominal system \( \hat{\Sigma}_i \) as

\[
\hat{\Sigma}_i: \quad \dot{x}^+_i = A_{ii} \hat{x}_i + B_i v_i
\]

where \( v_i \) is the input. Our goal is to relate inputs \( v_i \) in (5) to \( u_i \) in (2) and compute sets \( Z_i \subseteq \mathbb{R}^n, i \in M \) such that

\[
x_i(0) \in \hat{x}_i(0) + Z_i \Rightarrow x_i(t) \in \hat{x}_i(t) + Z_i, \ \forall t \geq 0.
\]

Assume there exists a nonempty RCI set \( Z_i \), for the constrained system (2), with respect to the disturbance \( w_i \in \mathcal{W}_i = \bigoplus_{j \in N_i} A_{ij} \mathcal{X}_j \). Therefore, if \( x_i \in Z_i \) then there exists \( u_i = \tilde{v}_i(x_i) : \mathcal{Z}_i \rightarrow \mathcal{U}_i \) such that \( x^+_i \in \mathcal{Z}_i, \ \forall u_i \in \mathcal{W}_i \).

Moreover, if \( x_i \in \hat{x}_i + Z_i \) and one uses the controller

\[
\hat{C}_i: \quad u_i = \tilde{v}_i + \tilde{v}_i(x_i - \hat{x}_i)
\]

then, for all \( v_i \in \mathcal{W}_i \), one has \( x^+_i \in \hat{x}^+_i + \mathcal{Z}_i \).

Following [10], the next goal is to compute tightened constraints \( \hat{x}_i \subseteq x_i \) and \( \mathcal{V}_i \subseteq u_i \) for \( \hat{x}_i(0) \) and \( v_i(0) \), respectively, guaranteeing (5) at all times. To this purpose, we introduce the following assumption.

**Assumption 3:** There exist \( \rho_{i,1} > 0, \rho_{i,2} > 0 \) such that

\[
Z_i \subseteq B_{i,1}(0) \subseteq \mathcal{X}_i \quad \text{and} \quad \mathcal{U}_i \subseteq B_{i,2}(0) \subseteq \mathcal{W}_i,
\]

where \( B_{i,1}(0) \subseteq \mathbb{R}^n, B_{i,2}(0) \subseteq \mathbb{R}^m \), and \( \mathcal{Z}_i = \tilde{v}_i(Z_i) \).

If Assumption 3 is verified, there are nonempty sets \( \hat{x}_i \) and \( \mathcal{V}_i, i \in \mathcal{M} \), that verify

\[
\hat{x}_i \subseteq \mathcal{X}_i, \mathcal{V}_i \subseteq \mathcal{U}_i, \mathcal{V}_i \cap \mathcal{X}_i \neq \emptyset.
\]

Under Assumptions 1-3, we set in (6)

\[
v_i(t) = v_i(0), \quad \hat{x}_i(t) = \hat{x}_i(0) + V_f t, \quad k = 0 \quad \text{to } \mathcal{W}_i \rightarrow \mathbb{R}^n.
\]

In (9), \( N_i \in \mathbb{N} \) is the prediction horizon and \( \ell_i(\hat{x}_i(k), v_i(k)) : \mathbb{R}^{n_i \times m_i} \rightarrow \mathbb{R}_+ \), is the stage cost. Moreover, \( V_{f} t, \mathcal{N}_i \) is the final cost and \( \mathcal{X}_f \) is the terminal set fulfilling the following assumption.

**Assumption 4:** For all \( i \in \mathcal{M} \), there exists an auxiliary control \( \kappa^u_{\mathcal{N}_i}(\hat{x}_i) \) and a \( \kappa_{\infty} \) function \( B_i \) such that:

(i) \( \ell_i(\hat{x}_i(k), v_i(k)) \geq B_i(\#(\hat{x}_i, v_i)) \) for all \( \hat{x}_i \in \mathbb{R}^n \), \( v_i \in \mathcal{V}_i \) and \( \ell_i(0) = 0 \);

(ii) \( \mathcal{X}_f \subseteq \mathcal{X}_i \) is an invariant set for \( \dot{\hat{x}}^+_i = A_{ii} \hat{x}_i + B_{i} \kappa^u_{\mathcal{N}_i}(\hat{x}_i) \);

(iii) \( \forall \hat{x}_i \in \mathcal{X}_f, \kappa^u_{\mathcal{N}_i}(\hat{x}_i) \in \mathcal{V}_i \);

(iv) \( \forall \hat{x}_i \in \mathcal{X}_f, V_f(\hat{x}^+_i - V_f(\hat{x}_i) \leq -\ell_i(\hat{x}_i, \kappa^u_{\mathcal{N}_i}(\hat{x}_i)) \).

We highlight that there are several methods, discussed e.g. in [14], for computing \( \ell_i(\cdot), V_f(\cdot) \) and \( \mathcal{X}_f \), verifying Assumption 4.

In summary, the controller \( C_i \) is given by (6), (8) and (9) and it is completely decentralized since it depends upon quantities of system \( \Sigma_i \) only. The main open problem is the following one.

**Problem \( \mathcal{P} \)**: Compute RCIs \( Z_i, i \in \mathcal{M} \) for (2) verifying Assumption 3.

In the next section, we show how to solve Problem \( \mathcal{P} \) in a distributed fashion with efficient computations, under
Assumptions 1 and 2. In this case, we will also show how sets $\bar{X}_i$ and $V_i$ verifying (7) can be readily computed.

### III. Decentralized synthesis of DeMPC

From Assumption 2, sets $X_i$ and $U_i$ can be written as

\begin{align}
X_i &= \{x_i \in \mathbb{R}^{n_i} : c_{x_i}^T x_i \leq d_{x_i}, \forall r \in 1 : g_i \} \\
U_i &= \{u_i \in \mathbb{R}^{n_i} : c_{u_i}^T u_i \leq d_{u_i}, \forall r \in 1 : l_i \}
\end{align}

where $c_{x_i}, d_{x_i}, c_{u_i}, d_{u_i} \in \mathbb{R}$.

Using the procedure proposed in [11] we can compute an RCI set $Z_i \subset X_i$ using an appropriate parametrization. As in Section VI of [11], we define the sets of variables $\Theta_i, i \in M$

$$\Theta_i = \{ \bar{z}_i^{(s,f)}(0,f) \in \mathbb{R}^{n_i}, \forall s \in 1 : k_i, \forall f \in 1 : q_i ; \}
\bar{u}_i^{(s,f)}(0,f) \in \mathbb{R}^{n_i}, \forall s \in 1 : k_i, \forall f \in 1 : q_i ;
\bar{r}_i^{(s,f)} \in \mathbb{R}, \forall f \in 1 : q_i, \forall r \in 1 : l_i ; \}
\bar{\psi}_i^{(r,s)} \in \mathbb{R}, \forall r \in 1 : l_i, \forall s \in 0 : k_i - 1 ;
\bar{\theta}_i \in \mathbb{R}, \forall r \in 1 : g_i, \forall s \in 0 : k_i - 1 ; \bar{\alpha}_i \in \mathbb{R} \}
$$

where $k_i, q_i \in \mathbb{N}$ are parameters of the procedure that can be chosen by the user as well as the set $Z_i^0 = \text{convh}(\bar{z}_i^{(0,f)}(0,f) \in \mathbb{R}^{n_i}, \forall f \in 1 : q_i)$, with $\bar{z}_i^{(0,1)}(0) = 0$. Let us define the sets $Z_i^* = \text{convh}(\bar{z}_i^{(s,f)}(0,f), \forall f \in 1 : q_i), \forall s \in 1 : k_i$ with $\bar{z}_i^{(s,1)} = 0$, and $\bar{Z}_i^* = \text{convh}(\bar{u}_i^{(s,f)}(0,f) \in \mathbb{R}^{n_i}, \forall f \in 1 : q_i), \forall s \in 0 : k_i - 1$ with $\bar{u}_i^{(s,1)} = 0$. The following assumption is needed to compute the RCI set $Z_i$.

**Assumption 5:** The set $Z_i^0$ is such that there is $\omega_i > 0$ verifying $W_i + B_{\omega_i}(0) \subset Z_i^0$.

We highlight that, in view of Assumption 2, the set $W_i$ contains the origin in its nonempty relative interior. Hence, under Assumption 5, the set $Z_i^0$ also contains the origin in its nonempty interior.

Consider the following set of affine constraints on the decision variable $\theta_i$

$$\Theta_i = \{ \theta_i : \alpha_i < 1 ; - \alpha_i \leq 0 ;
\bar{z}_i^{(k_1,1)}(0,f) \leq \sum_{f_2=1}^{q_i} \bar{r}_i^{(f_2)} \bar{z}_i^{(0,f_2)} = 0, \forall f_1 \in 1 : q_i ;
- \alpha_i + \sum_{f_2=1}^{q_i} \bar{r}_i^{(f_2)} \leq 0, \forall f_1 \in 1 : q_i ;
- \bar{r}_i^{(f_1)}(f_2) \leq 0, \forall f_2 \in 1 : q_i ;
\sum_{s=0}^{k_1-1} \bar{\psi}_i \bar{u}_i^{(r,s)} + d_{u_i}, \alpha_i \leq d_{u_i}, \forall r \in 1 : l_i ;
\bar{c}_{x_i}^T \bar{u}_i^{(s,f)} - \bar{\psi}_i \bar{\theta}_i \leq 0, \forall r \in 1 : l_i, \forall s \in 0 : k_i - 1, \forall f \in 1 : q_i ;
\sum_{s=0}^{k_1-1} \bar{\gamma}_i \bar{u}_i^{(r,s)} + d_{x_i}, \alpha_i \leq d_{x_i}, \forall r \in 1 : g_i ;
\bar{c}_{x_i}^T \bar{z}_i^{(s,1)} - \bar{\gamma}_i \bar{\theta}_i \leq 0, \forall r \in 1 : g_i, \forall s \in 0 : k_i - 1, \forall f \in 1 : q_i ;
\bar{z}_i^{(s,1),f} = A_i \bar{z}_i^{(s,f)} + B_i \bar{u}_i^{(s,f)}, \forall r \in 1 : g_i, \forall s \in 0 : k_i - 1, \forall f \in 1 : q_i \}\}
$$

The relation between elements of $\Theta_i$ and the RCI sets in Problem $P$ is established in the next proposition.

**Proposition 1 (Section 3 of [11]):** Let Assumptions 1 and 5 hold, sets $\bar{X}_i$ and $U_i$ be defined as in (10) and (11) respectively, and $k_i$ be a positive integer. If there exists an admissible $\theta_i \in \Theta_i$, then

$$Z_i = (1 - \alpha_i)^{-1} \bigoplus_{s=0}^{k_i-1} \bar{Z}_i^s \subset X_i$$

is an RCI set and the corresponding set $U_{\bar{z}_i}$ is given by

$$U_{\bar{z}_i} = (1 - \alpha_i)^{-1} \bigoplus_{s=0}^{k_i-1} \bigoplus_{s=0}^{k_i} U_i^s \subset U_i.$$

**Remark 1:** Under Assumption 2 the feasibility problem (13) is an LP problem, since the constraints in $\Theta_i$ are affine. Moreover we solve it while minimizing $\alpha_i$, since this corresponds to the minimization of the size of the set $Z_i$. We also note that the inclusion of 0 in the definition of sets $Z_i^s$, $\forall s \in 0 : k_i$, ensures that $Z_i^0$ contains the origin and hence, under Assumption 5, $Z_i$ contains the origin in its nonempty interior.

We highlight that the set of constraints $\Theta_i$ depends only upon local fixed parameters $\{A_{ij}, B_i, X_i, U_i\}$, fixed parameters $\{A_{ij}, X_j\}_{j \in \mathbb{N}_i}$ of predecessors of $X_i$ (because, from Assumption 5, it has to be checked that the set $Z_i^0$ verifies $Z_i^0 \supset \bigoplus_{j \in \mathbb{N}_i} \bigoplus_{j \in \mathbb{N}_i} U_i X_j$) and local tunable parameters $\theta_i$ (the decision variables (12)). However, $\Theta_i$ does not depend on tunable parameters of predecessors. This implies that the computation of sets $Z_i$ and $U_{\bar{z}_i}$ in (14) and (15) does not influence the choice of $Z_i$ and $U_{\bar{z}_i}$, $j \neq i$ and therefore Problem $P$ is decomposed in the following $M$ independent LP problems.

**Problem $P_i$:** Solve the feasibility LP problem $\theta_i \in \Theta_i$.

We note that if Problem $P_i$ is solved, then we can compute sets $\bar{X}_i$ and $V_i$ in (9d) and (9e) as

$$\bar{X}_i = X_i \cap Z_i, \quad V_i = U_i \cap U_{\bar{z}_i}.$$ (16)

The overall procedure for the decentralized synthesis of local controllers $C[i], i \in M$ is summarized in Algorithm 1 that has the following features.

**Proposition 2:** Under Assumption 1 and 2 if, for all $i \in \mathcal{M}$, controllers $C[i]$ are designed according to Algorithm 1, then also Assumptions 3, 4 and 5 are verified.

Note that if the LP problem in Step 2 is infeasible, we can restart the Algorithm with a different $k_i$. However, the existence of a parameter $k_i$ such that the LP problem is feasible is not guaranteed [11]. Steps 3, 4 and 5 of Algorithm 1, that provide constraints appearing in the MPC-$i$ problem (9), are the most computationally expensive ones because they involve Minkowski sums and differences of polytopic sets. Next, we show how to avoid burdensome computations.

### A. Implicit representation of sets $Z_i$ and $U_{\bar{z}_i}$

In this section we show how to rewrite constraint (9b) by exploiting the implicit representation of RCI sets proposed in Section VI.B of [11]. We have

$$\bar{z}_i^s = \bar{z}_i^{(s,f)} \text{ if } \forall f \in 1 : q_i, \exists \beta_i^{(s,f)} \geq 0 \text{ such that } \sum_{f=1}^{q_i} \beta_i^{(s,f)} = 1, \bar{z}_i^s = \sum_{f=1}^{q_i} \beta_i^{(s,f)} \bar{z}_i^{(s,f)}$$

where $\bar{z}_i^{(s,f)}$ is a RCI set and $\bar{z}_i^{(s,f)}$ is the corresponding set $U_i$.
Hence, $x_i[t] - \hat{x}_i[0]/t \in Z_i$ if and only if $\forall f \in 1 : q_i, \forall s \in 0 : k_i - 1$ there exist $\beta_{i}^{(s,f)} \in \mathbb{R}$ such that

$$\beta_{i}^{(s,f)} \geq 0, \sum_{f=1}^{q_i} \beta_{i}^{(s,f)} = 1$$

$$x_i[t] - \hat{x}_i[0]/t = (1 - \alpha_i)^{-1} \sum_{s=0}^{k_i-1} \sum_{f=1}^{q_i} \beta_{i}^{(s,f)} \hat{z}_{i}^{(s,f)}$$

In other words, we add to the optimization problem (9) the variables $\beta_{i}^{(s,f)}$ and replace (9b) with constraints (17).

With similar arguments, we can also provide an implicit representation of sets $\mathcal{U}_{z_i}$. In particular, we have that $u_{z_i}[\in \mathcal{U}_{z_i}$ if and only if $\forall f \in 1 : q_i, \forall s \in 0 : k_i - 1$ there exist $\phi_{i}^{(s,f)} \in \mathbb{R}$ such that

$$\phi_{i}^{(s,f)} \geq 0, \sum_{f=1}^{q_i} \phi_{i}^{(s,f)} = 1$$

$$u_{z_i} = (1 - \alpha_i)^{-1} \sum_{s=0}^{k_i-1} \sum_{f=1}^{q_i} \phi_{i}^{(s,f)} \hat{u}_{i}^{(s,f)}$$

### B. Computation of sets $\tilde{X}$ and $V_i$

Using (14) we obtain $\tilde{X} = X \cap (1 - \alpha_1)^{-1} \bigoplus_{s=0}^{k_i-1} \mathcal{Z}_s$. Recalling that $\mathcal{Z}_s, \forall s \in 0 : k_i - 1$ are defined as the convex hull of points $\hat{z}_{i}^{(s,f)}$, $f \in 1 : q_i$, we can compute the set $\tilde{X}_i$ using Algorithm 2. In particular, the operation in Step (IIii) amounts to solve suitable LP problems. We can compute $V_i$ using the implicit representation of $\mathcal{U}_{z_i}$ in a similar way. Indeed, it suffices to use Algorithm 2 replacing $\tilde{X}_i$ with $\mathcal{U}_i$ defined in (11) and points $\hat{z}_{i}^{(s,f)}$ with $\hat{u}_{i}^{(s,f)}$, $\forall s \in 0 : k_i - 1, \forall f \in 1 : q_i$.

### C. Computation of control law $\tilde{r}_{i}^{(\cdot)}$

Concerning equation (9), the control law $\tilde{r}_{i}^{(z_i)} \in U_{z_i}$, with $\hat{z}_i = x_i[t] - \hat{x}_i[0]/t$, guarantees that if $x_i[t] - \hat{x}_i[0]/t \in Z_i$ (i.e. MPC-i problem (9) is feasible) then there is a $\lambda_i > 0$ such that $x_i[t+1] - \hat{x}_i[1]/t \in \lambda_i Z_i$. To compute the control law $\tilde{r}_{i}^{(z_i)}$ one can use the methods proposed in [15] or in [11].

In [15] the authors propose to solve an LP problem in order to maximize the contractivity parameter $\lambda_i$, i.e. for a given $z_i$ one computes $\tilde{r}_{i}(z_i) \in U_{z_i}$ by minimizing the scalar $\lambda_i$ such that $A_i x_i + B_i \tilde{r}_{i}(z_i) \in \lambda Z_i \cap \mathcal{W}$. In [11] the authors propose an implicit representation of controller $\tilde{r}_{i}(z_i)$ based on the implicit representation (17) of set $\mathcal{Z}_i$.

We want to take advantage of both approaches and compute the control law $\tilde{r}_{i}(\cdot)$ solving the following LP problem

$$\tilde{r}_{i}(z_i) : \min_{\mu, \beta_{i}^{(s,f)}} \mu$$

$$\beta_{i}^{(s,f)} \geq 0, \forall f \in 1 : q_i, \forall s \in 0 : k_i - 1$$

$$\sum_{f=1}^{q_i} \beta_{i}^{(s,f)} = \mu, \forall s \in 0 : k_i - 1$$

$$\mu \geq 0$$

$$z_i = (1 - \alpha_i)^{-1} \sum_{s=0}^{k_i-1} \sum_{f=1}^{q_i} \beta_{i}^{(s,f)} \hat{z}_{i}^{(s,f)}$$

and setting $\tilde{r}_{i}(z_i) = (1 - \alpha_i)^{-1} \sum_{s=0}^{k_i-1} \beta_{i}^{(s,f)} \hat{u}_{i}^{(s,f)}$.

Concerning equation (9), the control law $\tilde{r}_{i}(z_i) \in U_{z_i}$, with $\hat{z}_i = x_i[t] - \hat{x}_i[0]/t$, guarantees that if $x_i[t] - \hat{x}_i[0]/t \in Z_i$ (i.e. MPC-i problem (9) is feasible) then there is a $\lambda_i > 0$ such that $x_i[t+1] - \hat{x}_i[1]/t \in \lambda_i Z_i$. To compute the control law $\tilde{r}_{i}(z_i)$ one can use the methods proposed in [15] or in [11].

### Definition 2: The feasibility region for the MPC-i problem is

$$X_i^N = \{ s_i \in X_i : (9) \text{ is feasible for } x_i[t] = s_i \}$$

### Algorithm 2

**Input:** set $X_i$ defined as in (10), points $z_i^{(s,f)}$, $\forall s \in 0 : k_i - 1, \forall f \in 1 : q_i$ and scalar $\alpha_i$.

**Output:** set $\tilde{X}_i$.

(I) $\tilde{C}_i = \{ c_{x_i,1}^{T}, \ldots, c_{x_i,n_i}^{T} \} \in \mathbb{R}^{n_i \times n_i}$ and $\tilde{D}_i = (d_{x_i,1}, \ldots, d_{x_i,n_i}) \in \mathbb{R}^{n_i}$.

(II) For each $s \in 0 : k_i - 1$

(i) For each $f \in 1 : q_i$

$\tilde{C}_{i} = (C_{i}, \tilde{C}_{i})$ and $\tilde{D}_{i} = (D_{i}, \tilde{D}_{i} = (1 - \alpha_i)^{-1} \tilde{C}_{i} \hat{z}_{i}^{(s,f)})$

(ii) Remove redundant constraints from $\tilde{C}_{i} \tilde{X}_{i} \leq \tilde{D}_{i}$ so obtaining the inequalities $\tilde{C}_{i} \tilde{X}_{i} \leq \tilde{D}_{i}$

(III) Set $\tilde{X}_i = \{ \tilde{x}_i[t] \in \tilde{C}_i \tilde{x}_i[t] \leq \tilde{D}_i \}$ where $\tilde{C}_i \in \mathbb{R}^{n_i \times n_i}$ and $\tilde{D}_i \in \mathbb{R}^{n_i}$. 

...
and the collective feasibility region is $X^N = \prod_{i \in M} X^N_i$.

The next theorem summarizes the key properties of the closed-loop system.

**Theorem 1**: Let Assumptions 1 and 2 hold, assume controllers $C_i$ in (6) are computed using Algorithm 1 and let the function $\tilde{\kappa}_i(\cdot)$ be given by (20). Then, the origin of (21) is asymptotically stable, $X^N$ is a region of attraction and $x(0) \in X^N$ guarantees constraints (4) are fulfilled at all time instants.

The proof of Theorem 1, that exploits properties of RCI sets and Minkowski operations, is provided in Appendix A of [12]. Here, we just highlight that asymptotic stability does not trivially follow from results on centralized tube MPC for systems affected by bounded and persistent disturbances (see [10], [11] and references therein). Indeed, the main challenge is to show that when all subsystems simultaneously use controllers $C_i$ (that are “robust to coupling”), the collective state (and then the coupling terms) tends to zero.

**Remark 2**: The main difference with respect to the PnP scheme proposed in [1] is the computation of sets $\mathcal{Z}_i$ and functions $\tilde{\kappa}_i(\cdot)$, $\forall i \in M$. For instance, in [1], using the robust control scheme proposed in [16], we set $\tilde{\kappa}_i(\cdot)$ as a linear function, i.e. $\tilde{\kappa}_i(x_i - \hat{x}_i) = K_i (x_i - \hat{x}_i)$. We note that in [1], the computation of $K_i$ and $\mathcal{Z}_i$ requires to solve a nonlinear optimization problem. In this paper, we have shown that for the PnP scheme proposed in Section II, using results from [11], we can compute set $\mathcal{Z}_i$ and function $\tilde{\kappa}_i(\cdot)$ solving LP problems only.

**D. PnP operations**

As in [1], the design method described in this paper allows for off-line PnP operations. As a starting point, we consider a plant composed by subsystems $\Sigma_i$, $i \in M$ equipped with local controllers $C_i$, $i \in M$ produced by Algorithm 1. When a new subsystem $\Sigma_{M+1}$ wants to plug in, Algorithm 1 must be executed for designing $C_{M+1}$ and local controllers for the successors of $\Sigma_{M+1}$ (see [1] for details). If all controllers can be designed (i.e. if Algorithm 1 never stops in step 2), then $\Sigma_{M+1}$ can be added to the network without spoiling stability and constraint satisfaction.

Concerning unplugging operations, similarly to [1], the removal of a system $\Sigma_k$ does not require the redesign of any controller in order to preserve stability and constraints satisfaction. However, the redesign of controllers for the successors of $\Sigma_k$ through Algorithm 1 could further improve performances.

**IV. Example: Power Network System**

We apply the proposed PnP-DeMPC scheme to the design of the AGC layer in power networks composed by several power generation areas connected through tie-lines. The goal is to keep frequencies as close as possible to a prescribed nominal value. The dynamics of an area linearized around equilibrium value for all variables is described by the following LTI model [17]

$$\Sigma_{\Sigma}^C : \dot{x}_i = A_{ii}x_i + B_iu_i + L_i\Delta P_{Li} + \sum_{j \in N_i} A_{ij}x_j$$

where $x_i = (\Delta \theta_i, \Delta \omega_i, \Delta P_{mi}, \Delta P_{vi})$ is the local state, $u_i = \Delta P_{ref,i}$ is the local control input, $\Delta P_{Li}$ is the local load change and $N_i$ is the sets of predecessor areas connected to $\Sigma_i^C$ through tie-lines. The meaning of state variables and matrices in (22) are described in [18]. Here, we just highlight that the main goal of the AGC is to steer frequency deviations $\Delta \omega_i$ to zero. Each subsystem $\Sigma_i^C$ is endowed with the constraints on $\Delta \theta_i$ and on $\Delta P_{ref,i}$ specified in [18]. Moreover, systems $\Sigma_i^C$ are obtained by discretizing models $\Sigma_i^C$ with 1 s sampling time treating $u_i$, $\Delta P_{Li}$, $x_j$, $j \in N_i$ as exogenous signals. Next, we design controllers $C_i$ for a power network composed by four areas (Scenario 1). Then, we connect a new area (Scenario 2) and update the control scheme. A further example where an area gets disconnected is provided in [12]. All computations have been performed using the PnPMPC toolbox for Matlab [19].

1) **Scenario 1**: We consider the power network in Figure 1. For each system $\Sigma_i$, the controller $C_i$, $i \in M$ is designed by executing Algorithm 1.

In Figure 2 we compare the performance of the proposed DeMPC scheme with the performance of the centralized MPC controller described in [20]. In the control experiment, step power loads $\Delta P_{Li}$ are specified in Table 3 of [20] and they account for the step-like changes of the control variables that are represented in Figure 2. We highlight that decentralized and centralized MPC are totally comparable in terms of frequency deviation (Figure 2(a)) and control variables (Figure 2(b)). In [20], we also compare the closed-loop behavior due to various controllers using suitably defined performance indices. This analysis shows that, compared to centralized MPC, PnP-DeMPC makes each area counteract local loads by producing locally more power (as also witnessed by higher values of $\Delta P_{ref,i}$) instead of receiving power from predecessor areas. Compared with the PnP controllers proposed in [1], PnP-DeMPC has better performances both in terms of tracking and reduction of average power transferred between areas.

2) **Scenario 2**: We consider the power network of Scenario 1 and add a fifth area connected as in Figure 3. Therefore, the set of successors to system 5 is $S_5 = \{2, 4\}$. The controllers $C_{\Sigma}$, $j \in \{5\} \cup S_5$ are tuned using Algorithm...
We highlight that no retuning of controllers $C_{[1]}$ and $C_{[3]}$ is needed since $\Sigma_{[1]}$ and $\Sigma_{[3]}$ are not predecessors of system $\Sigma_{[5]}$. As described in [12], performance levels close to those provided by centralized MPC are achieved. Moreover, similarly to Scenario 1, the distributed version of the controller proposed in this paper outperforms the PnP controllers in [1] by achieving better tracking and reducing power transfers.

### V. CONCLUSIONS

In this paper we proposed a PnP-DeMPC scheme based on the notion of tube MPC capable to guarantee asymptotic stability of the closed-loop system and constraints satisfaction at each time instant. Differently from [1], local controllers can be computed solving local LP problems. Future research will consider generalizations of PnP-DeMPC to output feedback schemes and tracking problems.

### REFERENCES


