MASTER

Non-centralized model predictive control techniques for power networks

Damoiseaux, A.C.R.M.

Award date:
2008

Eindhoven University of Technology
Department of Electrical Engineering
Control Systems

technische
universiteit
eindhoven

Non-centralized model predictive control techniques for power networks

by

A.C.R.M. Damoiseaux

Master of Science thesis

Project period: February 2008
Report Number: 08A/01

Supervisors:
Prof.dr.ir. P.P.J. van den Bosch
Dr. M. Lazar

Panel members:
Dr. A. Jokic
Dr. J. Myrcik

NON-CENTRALIZED MODEL PREDICTIVE CONTROL TECHNIQUES FOR POWER NETWORKS

A.C.R.M. Damoiseaux

Abstract—**Model predictive control (MPC) is one of the few advanced control methodologies that have proven to be very successful in real-life control applications. MPC has the capability to guarantee optimality with respect to a desired performance cost function, while explicitly taking constraints into account. Recently, there has been an increasing interest in the usage of MPC schemes to control power networks. The major obstacle for implementation is the large scale of power networks, which is prohibitive for a centralized approach.**

In this paper we critically assess and compare the suitability of three model predictive control schemes for controlling power networks. These techniques are analyzed with respect to the following relevant characteristics: the performance of the closed-loop system, which is evaluated and compared to the performance achieved with the classical automatic generation control (AGC) structure; the decentralized implementation, which is investigated in terms of size of the models used for prediction, required measurements and data communication, type of cost function and the computational time required by each algorithm to obtain the control action. Based on the investigated properties mentioned above, the study presented in this paper provides valuable insights to construct a non-centralized MPC technique for real-life electrical power networks.

Index Terms—**Model predictive control, Decentralized control, Distributed control, Power systems.**

I. INTRODUCTION

Current power networks consist of large scale power generating units where automatic generation control (AGC) is used for real-time control of the system frequency and tie line power interchange among control areas in the system [1]. However, there is a strong tendency to implement an increasing amount of decentralized power generating units and liberalize the power markets. Distributed generation introduces uncertainties in generation and therefore complicates control [2]. Large unpredictable power fluctuations from renewable energy sources, e.g. wind power, require efficient and fast acting controllers.

Recently, it was observed [3], [4], [5], that the model predictive control (MPC) technique has a potential for solving the above mentioned problems that will appear in future electrical power networks. The reason for this lies in the capability of MPC to guarantee optimality with respect to a desired performance objective, while explicitly taking constraints into account. Furthermore, MPC allows the usage of disturbance models, which can be employed to counteract the uncertainties introduced by renewable energy sources. For a detailed survey of MPC and constrained optimal control, the interested reader is referred to [6], [7].

Nevertheless, the fact that model predictive control is a global centralized control technique is a considerable drawback when power system control is considered. Centralized control implies that a single controller is able to perform the following sequence of operations within a time sample: measure all outputs of the system, compute an optimal control action and apply this control action to all actuators in the power system. As power networks are large scale systems, computationally as well as geographically, it is practically impossible to implement a centralized MPC controller.

This is one of the reasons for which the non-centralized formulation and implementation of MPC receives more and more interest, see for example [3], [8], [9], [10], [11], [12]. Roughly speaking, non-centralized MPC schemes can be divided into two categories: *decentralized techniques*, where there is no communication in between different controllers, and *distributed techniques*, where communication between different controllers is allowed. Furthermore, distributed MPC techniques can be categorized as techniques that require communication with all the controllers in the network and techniques that require communication solely with directly neighboring controllers.

A distinction between non-centralized MPC techniques can also be made depending on the level of coupling, i.e. some schemes handle dynamically coupled systems, while others handle dynamically decoupled systems with coupled objectives. The challenge for power systems is to obtain a computationally viable non-centralized MPC algorithm, without losing properties such as optimality and state constraint satisfaction. The latter property is crucial, due to the dynamic coupling present in power networks.

Among the research that focusses on non-centralized MPC, implementations for power system control have already been reported [3], [8], [9]. In this paper we select three non-centralized MPC techniques that can handle coupled dynamics. The selected techniques belong to one of the above mentioned categories[1]. The first non-centralized MPC technique [11] considered for power system control does not require any communication (for a specific choice of subsystem decomposition) and, therefore, belongs to the decentralized category. However, this scheme also allows for overlapping subsystems, case in which it requires communication. In what follows we will refer to the algorithm from [11] as decentralized model predictive control (DMPC). The second non-centralized MPC scheme [8] that we investigate

A.C.R.M. Damoiseaux is with the Eindhoven University of Technology, Department of Electrical Engineering, The Netherlands; E-mail: a.c.r.m.damoiseaux@student.tue.nl.

[1]Regarding other non-centralized MPC techniques, not considered in this paper, the interested reader is referred to [3], [8], [9], [10], [11], [12] and the references therein.

requires communication solely between directly neighboring controller areas and is in the following referred to as stability constrained distributed model predictive control (SC-DMPC). The third non-centralized MPC technique [4] requires communication between all subsystems and uses an iterative procedure to compute the control action. This scheme is referred to as feasible cooperation based model predictive control (FC-MPC). These three non-centralized MPC schemes will be compared to a centralized MPC algorithm and with the classical AGC control structure currently employed in control of real-life power systems.

The remainder of this article is organized as follows. Section II introduces the (centralized) MPC methodology along with conditions for stability. The three non-centralized MPC techniques are presented in detail in Section III. Section IV presents a simulation study of the considered MPC algorithms based on a suitably constructed example of a power network. In Subsection IV-C the investigated techniques are compared and based on the insights obtained from Section III and Section IV an alternative non-centralized MPC technique is proposed. This technique is described in more detail in Subsection V-A and simulation results are given in Subsection V-B. Conclusions are summarized in Section VI.

II. CENTRALIZED MODEL PREDICTIVE CONTROL

Model predictive control (also referred to as receding horizon control) is a control strategy that belongs to the finite horizon optimal control category. The unique, distinguishing feature of MPC is its ability to guarantee optimality with respect to a desired performance objective while explicitly taking constraints into account. This is achieved by solving online a finite horizon open-loop optimal control problem at each time instant. Within this problem, a model of the plant initialized with the current state is used to obtain a prediction of the future behavior of the plant. In this way, constraints on states and inputs can be explicitly taken into account in the computation of the control law. After a sequence of optimal control moves is computed, only the first one is applied to the plant and the whole process is repeated at the next time instant. This is the main difference from conventional control which commonly uses a pre-computed control law.

A graphical illustration of the basic principles behind MPC is depicted in Figure 1.

The typical system model considered in this paper is a discrete time state-space representation, which is given in the linear case by

$$x(t+1) = Ax(t) + Bu(t), \tag{1}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $x \in \mathbb{R}^n$ is the state and $u \in \mathbb{R}^m$ is the control input. $\mathbb{R}$ is the set of real numbers. For an arbitrary sequence $\mathbf{u} = (u(0), u(1), \ldots)$ we use the notation $\mathbf{u}_{[k]}$ to denote the truncation of $\mathbf{u}$ at $k \in \mathbb{Z}$, i.e. $\mathbf{u}_{[k]} := (u(0), u(1), \ldots, u(k))$ with $k \geq 1$. $\mathbb{Z}$ is the set of integers.

The open-loop finite horizon optimal control problem to be solved online is formally defined as follows.

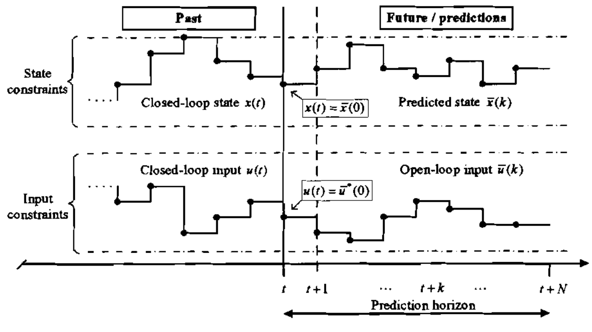

Fig. 1. A graphical illustration of Model Predictive Control.

Problem II.1 At discrete time $t \in \mathbb{Z}_+$ let $x(t)$ and $N \geq 1$ be given, set $\bar{x}(0) := x(t)$ and solve

$$\mathscr{P}_N(x) : V_N^*(x) = \min_{\bar{\mathbf{u}}_{[N-1]}} \{V_N(x, \bar{\mathbf{u}}_{[N-1]}) \,|\, \bar{\mathbf{u}}_{[N-1]} \in \mathscr{U}_N(x)\} \tag{2a}$$

$$V_N(x, \bar{\mathbf{u}}_{[N-1]}) = \sum_{k=0}^{N-1} \ell(\bar{x}(k), \bar{u}(k)) + F(\bar{x}(N)) \tag{2b}$$

$$= \sum_{k=0}^{N-1} \bar{x}^\top(k) Q \bar{x}(k) + \bar{u}^\top(k) R \bar{u}(k) + \bar{x}^\top(N) P \bar{x}(N). \tag{2c}$$

In Problem II.1, $\bar{\mathbf{u}}_{[N-1]} = (\bar{u}(0), \ldots, \bar{u}(N-1))$ is the sequence of control moves and N is the prediction horizon. $Q = Q^\top \succeq 0$, i.e. Q is a positive semidefinite symmetric matrix, while $R = R^\top \succ 0$, i.e. R is a positive definite symmetric matrix. The matrix $P = P^\top \succ 0$ weights the terminal state and is usually computed off-line such that stability is guaranteed, as will be shown below. $\bar{x}(k)$ and $\bar{u}(k)$ denote the predicted state and control input at time instant $t+k$. Given an open-loop control trajectory $\bar{\mathbf{u}}_{[N-1]}$, a prediction model of the form

$$\bar{x}(k+1) = A\bar{x}(k) + B\bar{u}(k), \qquad k = 0, \ldots, N-1, \tag{3}$$

is used to predict the future behavior of the system. At each time instant t, this model is initialized with the state measurement of the real system, i.e.

$$\bar{x}(0) := x(t). \tag{4}$$

Note that the real system (1) and the prediction model (3) do not have to be identical, as will be seen in the next section. The prediction model is usually only an approximation of the real system.

The control problem defined in (2a) is to minimize the cost function $V_N(x, \bar{\mathbf{u}}_{[N-1]})$ subject to all input sequences $\bar{\mathbf{u}}_{[N-1]}$ in the set $\mathscr{U}_N(x)$. This set contains all input sequences that satisfy certain desired state and input constraints, i.e.

$$\mathscr{U}_N(x) := \{\bar{\mathbf{u}}_{[N-1]} \in \mathbb{U}^N \,|\, \bar{x}(k) \in \mathbb{X}, \; k = 1, \ldots, N-1, \bar{x}(N) \in \mathbb{X}_f\}, \tag{5}$$

where $\mathbb{U}^N$ is the N-times Cartesian product, i.e. $\mathbb{U}^N := \mathbb{U} \times \ldots \times \mathbb{U}$. We assume that $\mathbb{U}$ is a convex compact subset of $\mathbb{R}^m$ and $\mathbb{X}$ is a convex, closed subset of $\mathbb{R}^n$, each set containing the origin in its interior. These sets implement physical input and state constraints. To guarantee stability, the terminal state $\bar{x}(N)$ is constrained in a terminal set $\mathbb{X}_f$, which must satisfy certain properties, outlined later in this section.

After the open loop optimization problem (2) is solved, the first element of the calculated optimal control sequence $\bar{\mathbf{u}}^*_{[N-1]} = (\bar{u}^*(0), \bar{u}^*(1), \ldots, \bar{u}^*(N-1))$ is applied to the system (1), i.e.

$$u(t) := \bar{u}^*(0), \tag{6}$$

and the rest of the control sequence is discarded. At the next time instant, i.e. $t = t+1$, the state of the system is measured and the procedure described above is repeated. This strategy is referred to as the moving or the receding horizon strategy. In this way feedback is introduced in a closed loop way and robustness is increased.

Stability of the resulting closed-loop system can be guaranteed *a priori* by choosing a terminal set and a terminal weight P that satisfy the following conditions [6]: $\mathbb{X}_f$ must be a positively invariant set [13] and satisfy the property

$$\begin{aligned} \mathbb{X}_f \subseteq \mathcal{O}_\infty :=\{x \in \mathbb{R}^n \mid K(A-BK)^k x \in \mathbb{U} \text{ and} \\ (A-BK)^k x \in \mathbb{X},\ k = 0, \ldots, \infty\}. \end{aligned} \tag{7}$$

The pair $\{P, K\}$ can be obtained as the solution of the unconstrained infinite horizon LQR problem [6], i.e.

$$P = (A+BK)^\top P(A+BK) + K^\top RK + Q, \tag{8a}$$

$$K = -(R + B^\top PB)^{-1} B^\top PA. \tag{8b}$$

In [6] it is proven that system (1) in closed-loop with a predictive control law obtained by solving Problem II.1 in a receding horizon manner, with $\mathbb{X}_f$ calculated for instance as in [13], is asymptotically stable. Clearly, a non-centralized implementation of MPC affects both feasibility of Problem II.1 and closed-loop stability and necessitates new stabilization conditions. The following Section presents possible solutions in this framework.

III. DESCRIPTION OF THE COMPARED NON-CENTRALIZED MPC SCHEMES

As already explained in the Introduction, the centralized implementation of the MPC methodology described in Section II is not possible in the case of power networks due to the very large scale of the system. Therefore, in this section we will present three non-centralized MPC techniques that are more suitable for power system control.

A. Decentralized MPC (DMPC)

The DMPC technique [11] uses the fact that the system can be divided into M subsystems and proposes the design of local MPC controllers, one for each subsystem. The subsystems are derived from the total system via an explicit transformation. The total system to be controlled is described by a discrete time state space model of the form:

$$x(t+1) = Ax(t) + Bu(t), \tag{9}$$

with $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$. The division into M subsystems is then performed via suitably defined matrices W_i and Z_i. These matrices collect the states and inputs belonging to subsystem i and are further employed to define the weighting matrices for each subsystem's states and inputs as follows:

$$x_i = W_i^\top x, \qquad u_i = Z_i^\top u, \tag{10a}$$

$$Q_i = W_i^\top Q W_i, \qquad R_i = Z_i^\top R Z_i, \tag{10b}$$

with $x_i \in \mathbb{R}^{n_i}$, $u_i \in \mathbb{R}^{m_i}$, $Q_i = Q_i^\top \succeq 0$ and $R_i = R_i^\top \succ 0$. Note that by (10a) each entry in x is assigned to one or more x_i and each entry in u is assigned to one or more u_i. This allows for overlapping subsystems. Compared to centralized MPC, the DMPC control scheme assigns a controller to each subsystem and each controller then solves online its own local open loop finite horizon optimization problem. More precisely, with each subsystem i the following finite horizon problem is defined:

Problem III.1 DMPC

$$\begin{aligned} \mathscr{P}_{N,i}(x_i) : V^*_{N,i}(x_i) = \min_{\bar{\mathbf{u}}_{[N-1],i}} \{ V_{N,i}(x_i, \bar{\mathbf{u}}_{[N-1],i}) \\ | \bar{\mathbf{u}}_{[N-1],i} \in \mathscr{U}_{N,i}(x_i)\}. \end{aligned} \tag{11a}$$

$$V_{N,i}(x_i, \bar{\mathbf{u}}_{[N-1],i}) = \sum_{k=0}^{N-1} \ell_i(\bar{x}_i(k), \bar{u}_i(k)) + F_i(\bar{x}_i(N)) \tag{11b}$$

$$\begin{aligned} &= \sum_{k=0}^{N-1} \bar{x}_i^\top(k) Q_i \bar{x}_i(k) + \bar{u}_i^\top(k) R_i \bar{u}_i(k) \\ &+ \bar{x}_i^\top(N) P_i \bar{x}_i(N). \end{aligned} \tag{11c}$$

Note that for each subsystem i the cost function now depends solely on the local states and inputs, i.e. on $x_i(t)$ and $\bar{\mathbf{u}}_{[N-1],i}$, and therefore a solution to the DMPC problem is no longer optimal with respect to the centralized MPC objective (2a), unless $x \equiv x_i$ and $u \equiv u_i$ for all i.

Given a certain open-loop control input sequence a prediction model of the form

$$\bar{x}_i(k+1) = A_i \bar{x}_i(k) + B_i \bar{u}_i(k) \qquad k = 0, \ldots, N-1, \tag{12a}$$

$$A_i = W_i^\top A W_i \quad B_i = W_i^\top B Z_i, \tag{12b}$$

is used to predict the state trajectories, where $A_i \in \mathbb{R}^{n_i\times n_i}$, $B_i \in \mathbb{R}^{n_i\times m_i}$. The prediction model uses an approximation of the real system, by *neglecting* dynamic coupling existing among neighboring subsystems. The reduction of the complexity of the prediction model compared to the total system depends on the number of subsystems. The prediction model is initialized with the partial state measurement of the real system at the current discrete-time instant, i.e.

$$\bar{x}_i(0) := W_i^\top x(t). \tag{13}$$

The optimization problem (11a) employs the following set of feasible control moves:

$$\mathscr{U}_{N,i}(x_i) := \{\bar{\mathbf{u}}_{[N-1],i} \in \mathbb{U}^N | \bar{u}_i(k) = K_i \bar{x}_i(k), \\ k = N_u, \ldots, N-1\}, \tag{14}$$

where $1 \leq N_u \leq N-1$ is the so-called control horizon [6]. Note that the DMPC approach does not incorporate state constraints. To guarantee *a priori* stability of each subsystem, a terminal penalty matrix P_i for each subsystem i can be defined following the centralized case as follows:

$$P_i = (A_i + B_i K_i)^\top P_i (A_i + B_i K_i) + K_i^\top R_i K_i + Q_i, \tag{15a}$$

$$K_i = -(R_i + B_i^\top P_i B_i)^{-1} B_i^\top P_i A_i. \tag{15b}$$

If $K_i \equiv 0$, as done in [11], then (15a) reduces to the Lyapunov equation for each subsystem:

$$P_i = A_i^\top P_i A_i + Q_i. \tag{16}$$

Note that with $Q_i \succ 0$, this implies that each matrix A_i has to be open-loop stable, i.e. all eigenvalues of A_i must be within the unit circle. The local optimal control input sequence $\bar{\mathbf{u}}^*_{[N-1],i}$ of all M controllers is collectively applied as input to the global system (9), i.e.

$$u(t) = [\bar{u}_1^*(0), \ldots, \bar{u}_i^*(0), \ldots, \bar{u}_M^*(0)]. \tag{17}$$

After the input is applied the whole procedure is repeated at the next time instant.

It is important to notice that in general $x(t+k) \neq \bar{x}(k)$ for $k = 1, \ldots, N$, i.e. the predicted state is not identical with the real system state trajectory. This is a result of the fact that the prediction model (12) only approximates the real system and is initialized with partial state measurements. The dynamic coupling between the subsystems is ignored. If the dynamic coupling is strong, the prediction mismatch can be large. Due to the fact that the optimization problem is based on possibly wrong predictions, this can result in loss of performance.

The advantages of this scheme are the relatively simple optimization problems that have to be solved by each subsystem, so the computational requirements are low. Furthermore it is important to notice that this scheme allows for non-overlapping subsystems, i.e each entry in x is assigned to one and only one x_i and each entry in u is assigned to one and only one u_i. Although overlapping subsystems are expected to achieve better performance, non-overlapping subsystems have a big advantage as no communication network is required between subsystems.

Feasibility of the optimization problem is guaranteed, because no state constraints are taken into account. Furthermore the article [11] provides *a posteriori* verifiable stability conditions. More precisely, the proposed stability test ensures *overall* stability of the entire system (9) in closed loop with the M decentralized MPC controllers, once the matrix P_i of the controller i is chosen as indicated in (16). This *a posteriori* stability condition checks whether the sum of all cost functions is a Lyapunov function or not and is based on the explicit form of each MPC controller thereby creating a PWA system. For certain conditions this reduces to a positive semidefiniteness check of a square $n \times n$ matrix. The drawback of this proof is that it has to carried out at a centralized level, which partly cancels out the advantages of the decentralized structure. For more detailed information about the stability test, the reader is referred to [11].

B. Stability constrained distributed MPC (SC-DMPC)

The SC-DMPC scheme from [8] requires that the plant dynamics, i.e.

$$x(t+1) = Ax(t) + Bu(t), \tag{18}$$

are given by the following matrices:

$$A = \begin{pmatrix} A_{11} & \ldots & A_{1i} & \ldots & A_{1M} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ A_{i1} & \ldots & A_{ii} & \ldots & A_{iM} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ A_{M1} & \ldots & A_{Mi} & \ldots & A_{MM} \end{pmatrix}, \tag{19a}$$

$$B = \begin{pmatrix} B_{11} & \ldots & 0 & \ldots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \ldots & B_{ii} & \ldots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \ldots & 0 & \ldots & B_{MM} \end{pmatrix}, \tag{19b}$$

with $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $A_{ii} \in \mathbb{R}^{n_i \times n_i}$, $A_{ij} \in \mathbb{R}^{n_i \times n_j}$, $B_{ii} \in \mathbb{R}^{n_i \times m_i}$, $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$. Note that the B matrix is block diagonal, i.e. a certain input only affects a single subsystem directly.

At each discrete-time instant t the state $x_i(t)$ of each subsystem is measured and each MPC controller solves the following local open loop quadratic optimization problem.

Problem III.2 SC-DMPC

$$\mathscr{P}_{N,i}(x_i) : V^*_{N,i}(x_i) = \min_{\bar{\mathbf{u}}_{[N-1],i}} \{V_{N,i}(x_i, \bar{\mathbf{u}}_{[N-1],i}) \\ |\bar{\mathbf{u}}_{[N-1],i} \in \mathscr{U}_{N,i}(x_i)\} \tag{20a}$$

$$V_{N,i}(x_i, \bar{\mathbf{u}}_{[N-1],i}) = \sum_{k=0}^{N-1} \ell_i(\bar{x}_i(k), \bar{u}_i(k)) + F_i(\bar{x}_i(N)) \tag{20b}$$

$$= \sum_{k=0}^{N-1} \bar{x}_i^\top(k) Q_i \bar{x}_i(k) + \bar{u}_i^\top(k) R_i \bar{u}_i(k) \\ + \bar{x}_i^\top(N) P_i \bar{x}_i(N). \tag{20c}$$

The SC-DMPC uses the following prediction model:

$$\bar{x}_i(k+1) = A_{ii}\bar{x}_i(k) + B_{ii}\bar{u}_i(k) + \sum_{j \neq i}^{M} A_{ij}\bar{x}_j(k). \tag{21}$$

Notice that this model takes the dynamic coupling of neighboring subsystems into account. However, as the real state trajectory of the neighbors is unknown, the predicted state trajectory of the previous time instant $t-1$, received from all direct neighbors, is used instead, i.e.

$$\bar{x}_j(k) := \bar{x}_j^*(k|t-1), \tag{22}$$

where $\bar{x}_j^*(k|t-1)$ is the optimal predicted state at time k with initial condition $\bar{x}_j(0) := x_j(t-1)$.

The prediction model is initialized with the current partial state measurement of the system, i.e.

$$\bar{x}_i(0) := x_i(t). \tag{23}$$

Note that $A_{ij} = 0$ if subsystem j is not directly coupled to subsystem i. Certain systems, as for example the power systems, are loosely coupled, so the number of elements A_{ij} $(i \neq j)$ that are zero is large. This significantly reduces the complexity of the prediction model.

The set of feasible input sequences for Problem III.2 is defined as follows:

$$\mathscr{U}_{N,i}(x_i) := \{\bar{\mathbf{u}}_{[N-1],i} \in \mathbb{U}^N | \ \|\bar{x}_i(1)\|_2^2 \leq \hat{l}_i\}, \tag{24}$$

where

$$\hat{l}_i := \max\{\|\bar{x}_i(1|t-1)\|_2^2, \|\bar{x}_i(0)\|_2^2\} - \beta_i\|x_i^1(0)\|_2^2, \tag{25}$$

with $0 < \beta_i < 1$ a tuning parameter. Furthermore,

$$\bar{x}_i(1|t-1) := A_{ii}\bar{x}_i(0) + B_{ii}\bar{u}_i(0) + \sum_{j\neq i}^{M} A_{ij}\bar{x}_j(0), \tag{26a}$$

$$\bar{x}_i(0) := x_i(t-1), \quad \bar{u}_i(0) := \bar{u}_i^*(t-1), \tag{26b}$$

and $x_i^1(0)$ is obtained from $x(t)$ via a similarity transformation that is based on the controllable companion form [8].

The terminal penalty matrix P that weights the terminal state $\bar{x}(N)$ can be computed as the solution of the unconstrained infinite horizon LQR problem, i.e.

$$P_i = (A_i + B_iK_i)^\top P_i(A_i + B_iK_i) + K_i^\top R_iK_i + Q_i, \tag{27a}$$

$$K_i = -(R_i + B_i^\top P_iB_i)^{-1}B_i^\top P_iA_i, \tag{27b}$$

as done also in the centralized case. After all M controllers have calculated the local optimal control input sequence $\bar{\mathbf{u}}^*_{[N-1],i}$, the collection of all local inputs is applied as input to the global system (18), i.e.

$$u(t) = [\bar{u}_1^*(0), \ldots, \bar{u}_i^*(0), \ldots, \bar{u}_M^*(0)]. \tag{28}$$

After the input is applied, the whole procedure is repeated at the next time instant. Notice that the stability of the SC-DMPC closed-loop system is ensured by the contraction constraint on the one-step ahead predicted state for each subsystem, which is explicitly required in each $\mathscr{U}_{N,i}(x_i)$.

Although SC-DMPC takes dynamic coupling into account in the prediction model, there is a prediction mismatch, i.e. $x(t+k) \neq \bar{x}(k)$ for $k = 1, \ldots, N$, because the assumed dynamic coupling is an estimation received from the neighbors, $x_j(k) \neq \bar{x}_j^*(k|t-1)$. However, the prediction mismatch, i.e. the mismatch between the predicted state trajectories $[\bar{x}_1(k), \ldots, \bar{x}_i(k), \ldots, \bar{x}_M(k)]$ and the state trajectories in case the collection of open loop inputs $[\bar{u}_1(k), \ldots, \bar{u}_i(k), \ldots, \bar{u}_M(k)]$ is applied to the full system (18), is in general smaller compared to DMPC, where dynamic coupling is fully neglected, i.e. Therefore, it is expected that the performance will be improved. The advantage of this approach lies in the fact that the predictions are improved at the cost of a slight increase in complexity of the optimization problem. Furthermore, only local state measurements are required to initialize the prediction model. However, this technique requires a communication network.

Feasibility of the SC-DMPC scheme is guaranteed, as the stabilization constraint, i.e. the contraction constraint defined by (24) and (25), is based on a controllable companion form. In [8] it is proven that this ensures the existence of a feasible control input in case of no state constraints. Moreover, it is proven that the collection of calculated control inputs comprises a feasible solution for the overall system. Furthermore, it is proven that the stability constraints of each subsystem, stabilize the overall system. For more details on feasibility and stability, the reader is referred to [8].

C. Feasible Cooperation based MPC (FC-MPC)

DMPC and SC-DMPC solve locally different optimization problems. Such strategies converge to suboptimal Nash equilibria, at best[2]. Feasible Cooperation-based MPC [9], on the other hand, solves the global optimization problem within every subsystem, thus ensuring that the resulting solution is Pareto optimal. This is an attractive feature of the FC-MPC over the DMPC and SC-DMPC. However, the FC-MPC requires communication with all subsystems, not just with directly neighboring ones.

The open-loop optimization problem solved online by each FC-MPC controller minimizes the same cost function over the local control input sequence[3] $\bar{\mathbf{u}}_{[N-1],i}$. As a controller is only able to optimize over its own optimization variables, an iterative procedure is used to achieve the global optimal solution. In the following, p denotes the iteration number. The FC-MPC problem is formulated next.

Problem III.3 FC-MPC

$$\mathscr{P}^p_{N,i}(x) : V^{p*}_{N,i}(x) = \min_{\bar{\mathbf{u}}^p_{[N-1],i}} \{V^p_{N,i}(x, \bar{\mathbf{u}}^p_{[N-1],i}) \ |\bar{\mathbf{u}}^p_{[N-1],i} \in \mathscr{U}^p_{N,i}(x)\}, \tag{29a}$$

$$V^p_{N,i}(x, \bar{\mathbf{u}}^p_{[N-1],i}) = \sum_{k=0}^{N-1} \ell(\bar{x}^p(k), \bar{u}^p_i(k)) + F(\bar{x}^p(N)) \tag{29b}$$

$$= \sum_{k=0}^{N-1} \bar{x}^{p\top}(k)Q\bar{x}^p(k) + \bar{u}_i^{p\top}(k)R_i\bar{u}^p_i(k) + \bar{x}^{p\top}(N)P\bar{x}^p(N). \tag{29c}$$

[2]Examples have been found where the Nash equilibria are unstable [4]. In such cases, the optimization process is divergent.

[3]Note that the notation used in [9] is adapted so that it is consistent with the notation of this paper.

In order to predict the future state trajectory $\bar{x}$, the following prediction model is used for each subsystem i:

$$\bar{x}^p(k+1) = A\bar{x}^p(k) + B[\bar{u}_1^{i,p}(k), \dots, \bar{u}_i^p(k), \dots, \bar{u}_M^{i,p}(k)], \quad (30)$$

where $\bar{u}_i^p(k)$ is the control input and optimization variable of subsystem i. The inputs of the other subsystems used by controller i (denoted by $\bar{u}_j^{i,p}(k)$) are set equal to the optimal solution obtained during the previous iteration, i.e.

$$\bar{u}_j^{i,p}(k) := \bar{u}_j^{p-1}(k),\ k = 0, \dots, N-1,\ j = 1, \dots, M,\ j \neq i, \quad (31)$$

where M denotes the total number of subsystems. The prediction model (30) is initialized with the current state of the system, i.e.

$$\bar{x}^p(0) := x(t), \qquad \forall p. \quad (32)$$

The set of feasible control moves is defined as:

$$\mathscr{U}_{N,i}^p(x) := \{\bar{\mathbf{u}}_{[N-1],i}^p \in \mathbb{U}^N\}, \qquad \forall p. \quad (33)$$

The terminal penalty matrix P used in (29c) is the solution of the unconstrained infinite horizon LQR problem, i.e.

$$P = (A + BK)^\top P(A + BK) + K^\top RK + Q, \quad (34a)$$

$$K = -(R + B^\top PB)^{-1}B^\top PA, \quad (34b)$$

with

$$R = \begin{pmatrix} R_1 & \dots & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \dots & R_i & \dots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & \dots & R_M \end{pmatrix} \quad (35)$$

In [9], K is chosen equal to zero, which yields:

$$P = A^\top PA + Q. \quad (36)$$

At each discrete time instant t the optimal control action is calculated via an iterative procedure, defined as follows:

- $p = 0$; the iteration variable is set to zero.
- $\bar{u}_i^0(k) := u_i^{p*}(k+1|t-1)$; the initial guess for the inputs at the first iteration, $p = 0$, is equal to the optimal input sequence of the previous time instant, $t-1$.

while($\rho_i > \varepsilon$)

- $\bar{\mathbf{u}}_{[N-1],i}^{p*} = \arg V_{N,i}^{p*}(x)\ \ \forall i$; all M controllers solve the optimization problem resulting in a local optimal control sequence.
- $\bar{\mathbf{u}}_{[N-1],i}^{p} = w_i\bar{\mathbf{u}}_{[N-1],i}^{p*} + (1-w_i)\bar{\mathbf{u}}_{[N-1],i}^{p-1}$; the actual control sequence is updated using the tuning parameter w.
- $\rho_i = \|\bar{\mathbf{u}}_{[N-1],i}^{p} - \bar{\mathbf{u}}_{[N-1],i}^{p-1}\|$; a check is performed to verify the stopping criterion.
- The local solution $\bar{\mathbf{u}}_{[N-1],i}^{p}$ is transmitted to all other controllers $j = 1, \dots, M,\ j \neq i$.
- $p = p + 1$; the iteration variable is increased by one.

end

When the stop criterion is satisfied for some $p \geq 1$, all M controllers communicate the calculated control actions, which are collected to form the control input of the overall system, i.e.

$$u(t) = [\bar{u}_1^p(0), \dots, \bar{u}_i^p(0), \dots, \bar{u}_M^p(0)]. \quad (37)$$

The feasibility of FC-MPC is guaranteed because there are no state constraints. Furthermore, convergence of the iterative procedure and stability of the closed loop system is proven[4]. As the controllers solve a global optimization problem and the sequence of cost functions is non-increasing with the iteration number p, the cost function can be used as a Lyapunov function to prove stability. For more detailed information about these results, the reader is referred to [9].

IV. Description of the Benchmark Test Example and Simulation Results

Automatic generation control (AGC) provides a suitable example for assessment and comparison of non-centralized MPC schemes for control of electrical power systems. All the simulations performed in this paper are based on the benchmark power system from [9], which is presented in the following subsection.

A. Test network and simulation scenario

A schematic representation of the test power system is depicted in Figure 2. The system consists of 4 control areas,

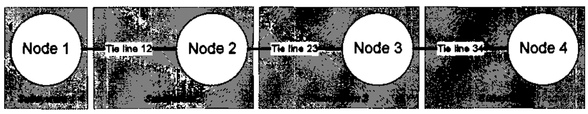

Fig. 2. Schematic representation of the power network.

with the dynamics of each area given by the following standard model [1]:

$$\frac{d\Delta\omega_i}{dt} = \frac{1}{J_i}(\Delta P_{M_i} - D_i\Delta\omega_i - \Delta P_{tie}^{ij} - \Delta P_{L_i}), \quad (38a)$$

$$\frac{d\Delta P_{M_i}}{dt} = \frac{1}{\tau_{T_i}}(\Delta P_{V_i} - \Delta P_{M_i}), \quad (38b)$$

$$\frac{d\Delta P_{V_i}}{dt} = \frac{1}{\tau_{G_i}}(\Delta P_{ref_i} - \Delta P_{V_i} - \frac{1}{r_i}\Delta\omega_i), \quad (38c)$$

$$\frac{d\Delta P_{tie}^{ij}}{dt} = b_{ij}(\Delta\omega_i - \Delta\omega_j), \quad (38d)$$

$$\Delta P_{tie}^{ji} = -\Delta P_{tie}^{ij}. \quad (38e)$$

These equations describe the dynamics of a generator and a tie line connecting the generators. These elements are graphically depicted in Figure 3. By combining these two "building blocks" a dynamical model of a power network with an arbitrary structure can be constructed.

The parameter values chosen for the example are given in the Appendix. In particular, note that the control input to the control area i is the signal ΔP_{ref_i}, which denotes the change in the reference value for power production in that area. The

[4]In fact, in [4] it has been shown that only a single iteration of the algorithm is required to guarantee stability.

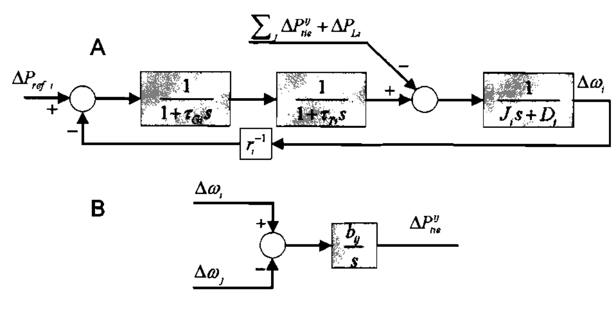

Fig. 3. Graphical representation of the dynamics of a generator (A) and tie line (B).

exogenous input signal ΔP_{L_i} represents the accumulated change of the power demand in the corresponding area.

The classical AGC structure used in current power networks consists of local PI controllers that bring the frequency deviation and the tie line power flow deviation to zero. The controller for control area i is defined as follows

$$\frac{d\Delta P_{ref_i}}{dt} = K(-B_i \Delta\omega_i - \Delta P_{tie}^{ij}), \quad (39)$$

with K and B as tuning parameters. For a more detailed description of classical AGC the reader is referred to [1].

The simulation scenario used in the assessment of the closed-loop performance is the following. At time instant zero, the system is in steady state with frequency and tie-line flow deviations equal to zero. At $t = 10$ a step disturbance of $+0.25$ affects control area 2, while at time instant $t = 60$ a step disturbance of -0.25 affects control area 3.

The simulation parameters such as weighting matrices and prediction horizon are chosen the same for all techniques. The numerical data related to the example can be found in the appendix, Table III. Furthermore, the optimization problems for all the assessed MPC schemes are implemented as quadratic programs and solved using the Matlab *quadprog* solver [5]. The SC-DMPC scheme is slightly adapted to make an implementation with a QP solver possible. More precisely, the 2-norm used for the contraction constraint is replaced by the 1-norm. Also, the number of iterations of FC-MPC is fixed to 1. Although DMPC allows for overlapping subsystems, decoupled subsystems are used for the simulation to investigate the performance of a completely decentralized control scheme.

B. Simulation results

The simulation results for all 3 non-centralized schemes are presented in Figure 4 together with results obtained with a centralized MPC algorithm and the classical AGC structure. From the state trajectories, only the trajectories of the network frequency deviation $\Delta\omega_2$ and of the tie-line power flow deviation ΔP_{tie}^{23} are plotted. Furthermore, the control inputs applied to the subsystems in area 2 and 3, i.e. ΔP_{ref_2} and ΔP_{ref_3}, are depicted in Figure 4. Table I shows the settling time [6] of the weighted states, i.e. the states for which elements in Q_i are nonzero, and the performance in the 2-norm, i.e. $\sum_{t=0}^{200} x(t)^\top Q x(t)$.

TABLE I

PERFORMANCE FIGURES: SETTLING TIME AND 2-NORM

	Settling time (s)							Norm
	ω_1	ω_2	ω_3	ω_4	P_{12}	P_{23}	P_{34}	
AGC	-	-	-	-	-	-	-	0.914
MPC	112	109	105	94	124	116	125	0.215
FC-MPC	111	109	104	94	124	116	125	0.213
DMPC	130	132	148	139	177	199	151	0.331
SC-DMPC	117	116	124	125	152	186	127	0.416

The simulation results indicate that the centralized MPC scheme achieves the best performance in terms of the settling time and the overshoots. In contrast, the classical AGC structure is characterized by the worst performance with respect to settling time and overshoots, i.e. all the non-centralized MPC schemes outperform the classical AGC control method.

It is expected that the performance of the non-centralized control techniques is directly correlated with the level of communication between subsystems. The results of the performed simulations are in conformance with this expectation, although the observed difference in the settling time between DMPC and the SC-DMPC controllers is very small. Finally, note that FC-MPC performs almost identically with the centralized MPC, in spite the fact that only one iteration in the FC-MPC scheme was allowed. It is remarkable that, given the fact that each time instant the cost based on the centralized objective for each FC-MPC controller is larger then the cost of the centralized MPC controller, the FC-MPC technique performs slightly better.

The computational complexity can be determined by inspection of the size of the optimization problem. As explained before, the optimization problems are quadratic programs of the form:

$$\min_x \; x^\top H x, \quad (40a)$$

$$\text{subject to } Ax \leq B, \quad (40b)$$

$$A_{eq}x = B_{eq}. \quad (40c)$$

The computational complexity depends on the sizes of A, B, A_{eq}, B_{eq} and H. The size of the matrices A and A_{eq} is listed in Table II for each algorithm (the size of B, B_{eq} and H can be derived from these matrices). The computational

TABLE II

SIZES OF OPTIMIZATION PROBLEM FOR QP SOLVER

Technique	A	A_{eq}
Cent. MPC	152×376	300×376
FC-MPC	38×319	300×319
SC-DMPC	39×99	80×99
DMPC	38×99	80×99

[5] From the available QP solvers (NAG, CLP, SeDuMi), quadprog (version 3.1.2 (R2007b)) was the fastest solver for this problem.

[6] The formal definition of settling time is slightly adapted, as we consider settling time as the time it takes for the states to settle within a certain bound after two step disturbances.

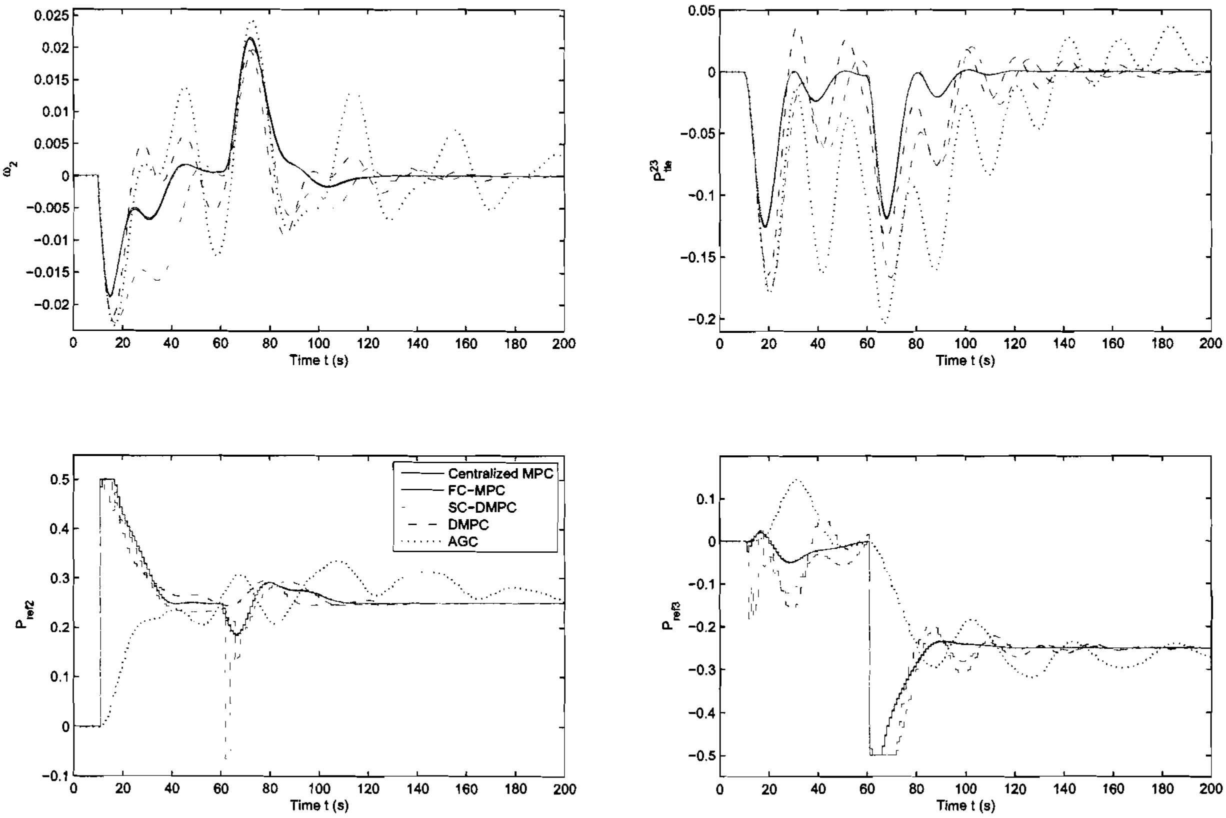

Fig. 4. Simulation results of the considered control techniques.

complexity can also be judged by the computation time that is required for solving the optimization problem of controller i at time instant t. The results are depicted in Figure 5. The results show that from a computational point of view DMPC and SC-DMPC are preferred because these techniques require significantly less computational power compared to centralized MPC. The computational time of DMPC and SC-DMPC are comparable as the size of the optimization problems is the same, except from an additional state constraint in SC-DMPC. Whenever this constraint is active, as can be observed in Figure 5 around $t = 10s$, the computational time required to solve the SC-DMPC problem can increase considerably. This issue could be resolved by relaxing the stabilization constraint. FC-MPC shows a slightly reduced computational time compared to centralized MPC and is therefore only advantageous for a small number of iterations.

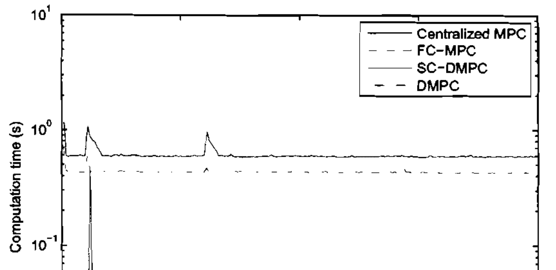

Fig. 5. The computation time for all algorithms.

Furthermore, it is important to notice that the required computational time for a controller of DMPC and SC-DMPC is independent on the number of subsystems. On the other hand, the computational time required by a centralized MPC controller and a FC-MPC controller, is dependent on the number of subsystems. To investigate this relation, a series of simulations has been done and the computational time as a function of the number of control areas is plotted. The results are shown in Figure 6. The data points can be very well approximated by a third order polynomial. Assuming that this polynomial fit is valid, the required computational time by a centralized MPC controller in case of a real life power network consisting of 1000 control areas is approximately 17 days. Given the fact that the available computation time is 1 second (sample time), this experiment underlines the importance of a non-centralized implementation of power network control.

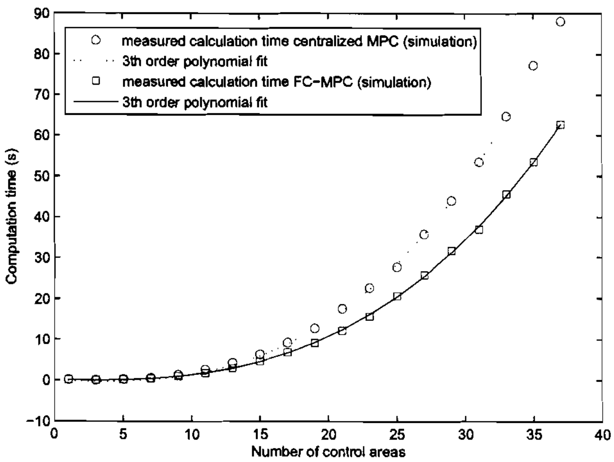

Fig. 6. The computation time for a centralized MPC controller and a FC-MPC controller as function of the number of control areas.

C. Assessment

A schematic representation of the investigated techniques is depicted in Figure 7. The results of Section III and Sub-

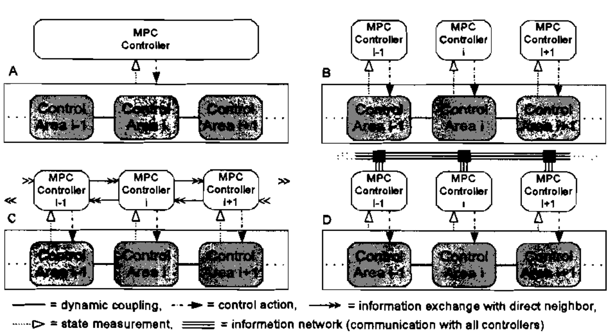

Fig. 7. A schematic representation of centralized (A) and non-centralized MPC techniques; DMPC (B), SC-DMPC (C), FC-MPC (D)

section IV-B give valuable insights in performance and complexity. Firstly, the large scale of real-life power networks prohibits a non-centralized implementation of MPC that requires communication with a large number of subsystems in the network. This means that Pareto optimality of the non-centralized MPC control action is not a feasible goal for the current power networks. Future advances in communications and increasing processing power might bring this goal closer to realization. Currently, however communication with a large number of subsystems, is unrealistic.

Secondly, a completely decentralized implementation of MPC seems to be less efficient with respect to performance but still, much better than the classical AGC control structure. Therefore there is room for a tradeoff: one can either use decentralized MPC if the provided performance is acceptable or, distributed MPC with limited communication can provide a feasible alternative for increasing performance in real-life power system control. In this latter case, the accuracy of the estimates of the real-system state trajectories plays a crucial role in improving performance.

It is observed in Section III and Subsection IV-B there are two important aspects that determine performance of a non-centralized MPC technique.

- *Prediction accuracy.* As a result of dynamic coupling between subsystems, a prediction model neglecting this dynamic coupling creates a prediction mismatch, i.e. the mismatch between the predicted state trajectories and the state trajectories in case the collection of open loop inputs is applied to the full system, i.e.

$$\varepsilon(k) = A^k \bar{x}(0) + \sum_{\lambda}^{k-1} A^{\lambda} B \begin{bmatrix} \bar{u}_1(\lambda) \\ \vdots \\ \bar{u}_i(\lambda) \\ \vdots \\ \bar{u}_M(\lambda) \end{bmatrix} - \begin{bmatrix} \bar{x}_1(k) \\ \vdots \\ \bar{x}_i(k) \\ \vdots \\ \bar{x}_M(k) \end{bmatrix}. \tag{41}$$

  Communication between MPC controllers to estimate the dynamic coupling, results in an improvement of the prediction accuracy. However, the dynamic coupling can only be estimated and is not known exactly as the controllers solve their optimization problem simultaneously. Accurate predictions are important, as solving a optimization problem based on wrong predictions results in a non-optimal solution. Furthermore, accurate predictions are required for state constraint handling, as a constrained local optimal solution can be non-feasible on the global system if predictions are not accurate.
- *Optimality with respect to the centralized solution.* The solution depends not only on the prediction accuracy. Even if the prediction mismatch (41) is zero, i.e. $\varepsilon(k) = \mathbf{0}$ for all k, the solution of an optimization problem using a local objective will differ from the centralized solution. As the prediction model of a non centralized controller doesn't incorporate all states, the objective function of a non-centralized controller can only depend on local states predicted by the prediction model and local inputs, and this results in a Nash equilibrium. A Nash equilibrium is in general not equal to the centralized Pareto optimal solution. Achieving the centralized solution leads inevitable to the requirement of a full prediction model.

Preferably, a non-centralized MPC technique achieves accurate predictions by communication with a limited number of controllers, similar to SC-DMPC, such that the extent of decentralization is high. As controllers cannot achieve accurate predictions without communication, the next best thing is communication with solely direct neighboring controllers. Pareto optimality can only be achieved with objectives that take all states of the overall system into account, such as FC-MPC. Although a structure such as FC-MPC is not preferred, a non-centralized MPC technique should allow for a flexible way of constructing local control problems by decomposition of the overall system, like DMPC, such that the global Pareto optimal solution can be approximated. Finally, an iterative method to improve the accuracy of the predictions, similar to FC-MPC, should be incorporated. The non-centralized MPC technique proposed in Section V

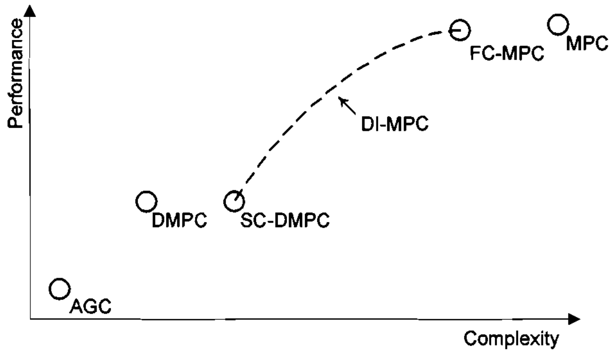

Fig. 8. Comparison of the complexity-performance relation. Complexity and performance of DI-MPC depends on the size of the prediction model and number of iterations.

uses an iterative technique to achieve accurate predictions and guarantees zero one step ahead prediction mismatch, i.e. $\varepsilon(1) = \mathbf{0}$. As a result of this, if each controller i finds a feasible solution to their local optimization problem subject to state constraints, the collective control input will be feasible on the overall system, i.e. none of the constraints is violated. Furthermore, the technique allows for overlapping prediction models of arbitrary size such that the dynamic behavior of the centralized controlled closed loop system can be approximated by the choice of the size of the prediction models thereby creating a tuning parameter that allows for a trade off between complexity and the approximation of the centralized solution. The technique is in the following referred to as distributed iterative MPC (DI-MPC).

Figure 8 shows a non-quantified comparison of the complexity-performance relation of all investigated techniques. Note that DI-MPC with no iterations and non overlapping prediction models will be almost equal to SC-DMPC when complexity and performance is considered. In case of full system prediction models and iterations, DI-MPC equals FC-MPC. The tuning options of DI-MPC, i.e. size of the prediction model and the number of iterations, determine the complexity and performance of this technique and will be on the dotted line.

V. Distributed Iterative MPC technique

The distributed iterative MPC technique proposed in Subsection IV-C will be described in Section V-A in more detail. Subsection V-B will give simulation results of this technique.

A. Distributed Iterative MPC technique

To use the DI-MPC technique the system is decomposed into M subsystems and a local MPC controller is assigned to each subsystem. The subsystems are derived from the total system using an explicit transformation. The total system to be controlled is described by a discrete time state space model of the form:

$$x(t+1) = Ax(t) + Bu(t), \tag{42}$$

with $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$. The division into M subsystems is then performed via suitably defined matrices W_i and Z_i.[7] These matrices collect the states and inputs belonging to subsystem i and are further employed to define the weighting matrices for each subsystem's states and inputs as follows:

$$x_i = W_i^\top x, \qquad u_i = Z_i^\top u, \tag{43a}$$

$$Q_i = W_i^\top Q W_i, \qquad R_i = Z_i^\top R Z_i, \tag{43b}$$

with $x_i \in \mathbb{R}^{n_i}$, $u_i \in \mathbb{R}^{m_i}$, $Q_i = Q_i^\top \succeq 0$ and $R_i = R_i^\top \succ 0$. With subsystem i the following finite horizon optimization problem is defined and is solved by its assigned controller.

Problem V.1 DI-MPC

$$\mathscr{P}^p_{N,i}(x_i) : V^{p*}_{N,i}(x_i) = \min_{\bar{\mathbf{u}}^p_{[N-1],i}} \{V^p_{N,i}(x_i, \bar{\mathbf{u}}^p_{[N-1],i}) \,|\, \bar{\mathbf{u}}^p_{[N-1],i} \in \mathscr{U}^p_{N,i}(x_i)\}, \tag{44a}$$

$$V^p_{N,i}(x_i, \bar{\mathbf{u}}^p_{[N-1],i}) = \sum_{k=0}^{N-1} \ell_i(\bar{x}^p_i(k), \bar{u}^p_i(k)) + F_i(\bar{x}^p_i(N)) \tag{44b}$$

$$= \sum_{k=0}^{N-1} \bar{x}^{p\top}_i(k) Q_i \bar{x}^p_i(k) + \bar{u}^{p\top}_i(k) R_i \bar{u}^p_i(k) + \bar{x}^{p\top}_i(N) P_i \bar{x}^p_i(N). \tag{44c}$$

where p is the iteration number.

Given a certain open-loop control input sequence a prediction model for each subsystem i of the form

$$\bar{x}^p_i(k+1) = A_i \bar{x}^p_i(k) + B_i \bar{u}^p_i(k) + A^i_j \bar{x}^{i,p}_j(k) + B^i_j \bar{u}^{i,p}_j(k), \quad k = 0, \ldots, N-1, \quad \forall p, \tag{45a}$$

$$A_i = W_i^\top A W_i, \quad B_i = W_i^\top B Z_i, \tag{45b}$$

$$A^i_j = W_i^\top A(I - W_i W_i^\top), \quad B^i_j = W_i^\top B(I - Z_i Z_i^\top), \tag{45c}$$

is used to predict the future state trajectories $\bar{x}_i$, where $A_i \in \mathbb{R}^{n_i\times n_i}$, $B_i \in \mathbb{R}^{n_i\times m_i}$, $A^i_j \in \mathbb{R}^{n_i\times n}$, $B^i_j \in \mathbb{R}^{n_i\times m}$. The control input $\bar{u}^p_i(k)$ of subsystem i is the optimization variable. The inputs of other subsystems used by controller i (denoted by $\bar{u}^{i,p}_j(k)$) are set equal to the optimal solution obtained by other controllers during the previous iteration, i.e.

$$\bar{u}^{i,p}_j(k) := [\bar{u}^{p-1*}_1(k), \ldots, \bar{u}^{p-1*}_i(k), \ldots, \bar{u}^{p-1*}_M(k)], \quad k = 0, \ldots, N-1. \tag{46}$$

Note that the matrix B^i_j is zero for inputs that do not directly affect the states x_i. Due to the structure of power systems, the amount of communication is not so large. The state

[7] The notation of DMPC is used to allow for definition of overlapping prediction models.

$\bar{x}_j^{i,p}(k)$ is received by communication with other controllers, as follows:

$$\bar{x}_j^{i,p}(k) := [\bar{x}_1^{p-1*}(k), \dots, \bar{x}_i^{p-1*}(k), \dots, \bar{x}_M^{p-1*}(k)], \quad k = 0, \dots, N. \tag{47}$$

Note that the matrix A_j^i is zero for non directly coupled states. This implies that for loosely coupled systems, such as power systems, the required amount of communication is not so large.

The prediction model (45) is initialized with the current state of the system, i.e.

$$\bar{x}_i^p(0) := W_i^\top x(t), \qquad \forall p. \tag{48}$$

The optimization problem (44a) employs the following set of feasible control moves:

$$\mathscr{U}_{N,i}^p(x) := \{\bar{\mathrm{u}}_{[N-1],i}^p \in \mathbb{U}^N | \bar{x}_i^p(k) \in \mathbb{X}_i, \quad k = 1, \dots, N-1, \ \bar{x}_i^p(N) \in \mathbb{X}_{f,i}\}, \quad \forall p, \tag{49}$$

The terminal set $\mathbb{X}_{f,i}$ must be a positively invariant set satisfying the property

$$\mathbb{X}_{f,i} \subseteq \mathcal{O}_{\infty,i} := \{x_i \in \mathbb{R}_i^n \mid K_i(A_i - B_iK_i)^k x_i \in \mathbb{U} \ \text{and} \ (A_i - B_iK_i)^k x_i \in \mathbb{X}_i, \ k = 0, \dots, \infty\}. \tag{50}$$

The pair $\{P_i, K_i\}$ used in (44c) and (50) can be obtained as the solution of the unconstrained infinite horizon LQR problem, i.e.

$$P_i = (A_i + B_iK_i)^\top P_i(A_i + B_iK_i) + K_i^\top R_iK_i + Q_i, \tag{51a}$$

$$K_i = -(R_i + B_i^\top P_iB_i)^{-1}B_i^\top P_iA_i. \tag{51b}$$

If the matrix A_i is stable then $K_i \equiv 0$ is possible. So (51a) reduces to the Lyapunov equation for each subsystem:

$$P_i = A_i^\top P_iA_i + Q_i. \tag{52}$$

At each discrete time instant t the optimal control action is calculated via an iterative procedure, defined as follows:

- $p = 0$; the iteration variable is set to zero.
- $\bar{u}_j^{i,0}(k) := u_j^{p*}(k+1|t-1)$ and $\bar{x}_j^{i,0}(k) := x_j^{p*}(k+1|t-1)$; the initial guess for the inputs of neighboring subsystems and the dynamic coupling at the first iteration, $p = 0$, is equal to the sequences calculated at the previous time instant, $t-1$.

while$(\rho_i > \varepsilon$ or $p \leq p_{max})$

- $\bar{\mathrm{u}}_{[N-1],i}^{p*} = \arg V_{N,i}^{p*}(x_i), \quad \forall i$; all M controllers solve the optimization problem (44), using (46) and (47) for $p \geq 1$ resulting in a local optimal control sequence.
- $\rho_i = \|\bar{\mathrm{u}}_{[N-1],i}^{p*} - \bar{\mathrm{u}}_{[N-1],i}^{p-1*}\|$; a check is performed to verify the stopping criterion.
- The local optimal sequences $\bar{\mathrm{u}}_{[N-1],i}^{p*}$ and $\bar{\mathrm{x}}_{[N-1],i}^{p*}$ are transmitted to all neighboring controllers that require it to calculate their optimization problem.
- $p = p + 1$; the iteration variable is increased by one.

end

When the stopping criterion is satisfied for some $p \geq 1$ or the maximum number of iterations is reached, all M controllers apply their calculated control actions to the control input of the subsystems, such that the collection of all local inputs is the input to the global system (42), i.e.

$$u(t) = [\bar{u}_1^{p*}(0), \dots, \bar{u}_i^{p*}(0), \dots, \bar{u}_M^{p*}(0)]. \tag{53}$$

After the input is applied the whole procedure is repeated at the next time instant, $t+1$.

The one step ahead prediction of the controlled area is:

$$x_i^{c,p}(1) = A_i^{c,p}\bar{x}_i^{c,p}(0) + A_i^{cn}\bar{x}_i^{cn,p}(0) + B_i\bar{u}_i^p(0) \quad \forall p, \tag{54}$$

where $x_i^{c,p}$ are the states of the control area directly affected by the optimization variable, $\bar{u}_i$, with $x_i^{cn,p}$ the states of the directly neighboring control areas. The states $\bar{x}_i^{c,p}(0)$ and $\bar{x}_i^{cn,p}(0)$ are known exactly, as these are measured. Furthermore $\bar{u}_i^p(0)$ is the optimization parameter and is therefore exactly known and actually applied to the system. This results in the fact that the one step ahead prediction of the state of the controlled area is exactly known even for a finite number of iterations. As a result, the collective control input, i.e. $[\bar{u}_1^{p*}(0), \dots, \bar{u}_i^{p*}(0), \dots, \bar{u}_M^{p*}(0)]$ will be a feasible control action on the global system (42) subject to state constraints, assumed that each controller i found a feasible control sequence to their local optimization problem. Although only the one step ahead prediction is correct and the predictions for $k \geq 2$ generally have a prediction mismatch, correct one step ahead prediction is sufficient to guarantee the effect mentioned above, due to the fact that MPC is implemented in closed loop, i.e. states are measured each time instant.

B. Simulation results of the DI-MPC technique

For DI-MPC it is observed that the predictions of the controllers converge to the real state trajectories by iterative communication between neighboring controllers. This iterative convergence is shown in Figure 9 where the prediction of the state trajectory P_{12} is plotted for 5 subsequent iterations. The prediction gets worse for larger prediction time instants k but the overall prediction error decreases every iteration.

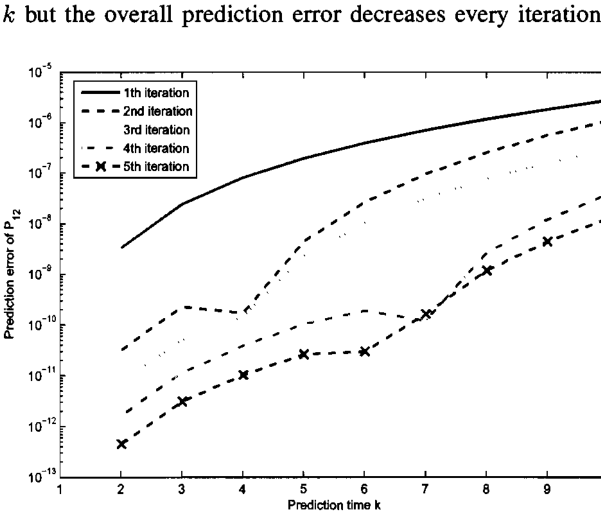

Fig. 9. Error of state P_{12} for 5 iterations.

As explained in Subsection V-A, the collective control signal to the overall system is feasible, assumed that each controller found a feasible control move to their optimization problem. To investigate the performance of the controlled four generator power system (described in Section IV) subject to state constraints a simulation has been done. For clarity a single step disturbance event is chosen (disturbance control area 2: $+0.25$, disturbance control area 2: -0.25 at time $t = 0$). The simulation results of constrained DI-MPC control of the power network are shown in Figure 10. The states P_{23}, ω_2 and ω_3 are constrained to ± 0.01. The simulation results show that the constraints are not violated and the control moves are feasible at each time instant. If the same system is controlled with e.g. DMPC and the same constraints are incorporated in the optimization problems, the collective control move is in general infeasible on the global system, even if each controller i finds a feasible solution to their local optimization problem.

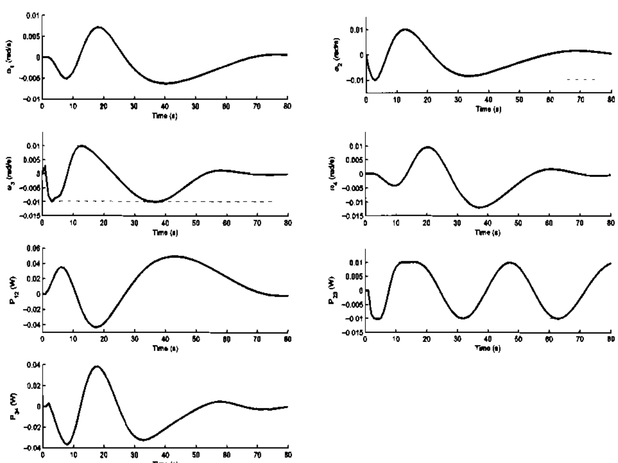

Fig. 10. Simulation results of a DI-MPC controlled power network with state constraints.

The DI-MPC technique allows for overlapping prediction models, thereby creating a possibility to find a trade off between complexity and optimality with respect to the centralized solution, as described in Subsection V-A. To investigate the optimality of the solution with respect to the centralized solution, a simulation has been done on the benchmark power system described in Section IV. A controller is assigned to each control area, and the size of the prediction model is varied, thereby creating overlapping prediction models. Note that this is not equal to increasing the size of the subsystems as in this case the number of optimization parameters is increased. The results for the benchmark system are shown in Figure 11. The results show that the behavior of the DI-MPC controlled power network approximates the centralized Pareto optimal solution as the size of the prediction model increases. Note that if the prediction model covers all four control areas, the solution is equal to the FC-MPC solution.

These results are important because they show that controllers assigned to a control area can approximate the Pareto solution by increasing the size of the prediction model. This gives a tuning possibility that allows for a trade off between the extent of decentralization (which is related to the computational complexity) and the optimality with respect to the centralized Pareto optimal solution.

An experimental method to determine the size of the prediction models is by investigating the sensitivity of the states of the overall system to a control input. An impulse input is applied to control input i and the response of the states exceeding a certain level γ are part of the prediction model of control area i. States that don't exceed the tuning parameter γ are not strongly influenced by control input i and therefore their contribution to the cost function of controller i will not be significant. By removing the states that are not significantly influenced, the complexity can be strongly reduced, whereas the solution of the optimization problem is not significantly different from the optimization problem with a centralized objective.

VI. Conclusions

In this section we summarize the results of this paper and give constructive recommendations for future research on the topic of non-centralized MPC in power system control.

Firstly, the large scale of real-life power networks prohibits a non-centralized implementation of MPC that requires communication with a large number of subsystems in the network. This means that Pareto optimality of the non-centralized MPC control action is not a feasible goal for the current power networks.

Secondly, a completely decentralized implementation of MPC seems to be less efficient with respect to performance but still, much better than the classical AGC control structure. Therefore there is room for a tradeoff: one can either use decentralized MPC if the provided performance is acceptable or, distributed MPC with limited communication can provide a feasible alternative for increasing performance by increasing prediction accuracy. In this latter case, the prediction accuracy of the real-system state trajectories by using estimates of the dynamic coupling plays a crucial role in improving performance.

The proposed DI-MPC technique combines the advantageous properties of existing non-centralized MPC techniques. The iterative technique achieves improvement of the prediction accuracy. Furthermore DI-MPC guarantees zero one step ahead prediction mismatch, which is important in case of state constraints. More precisely, the collective control input is feasible on the overall system, assumed that each controller found a feasible solution to their local optimization problem. Furthermore, this technique allows for arbitrarily large (overlapping) prediction models, such that a trade off can be made between the level of decentralization and optimality with respect to the centralized Pareto optimal solution.

Although the simulations show convergence, stability and state constraint satisfaction for a DI-MPC controlled power network, a significant problem, currently not solved, is

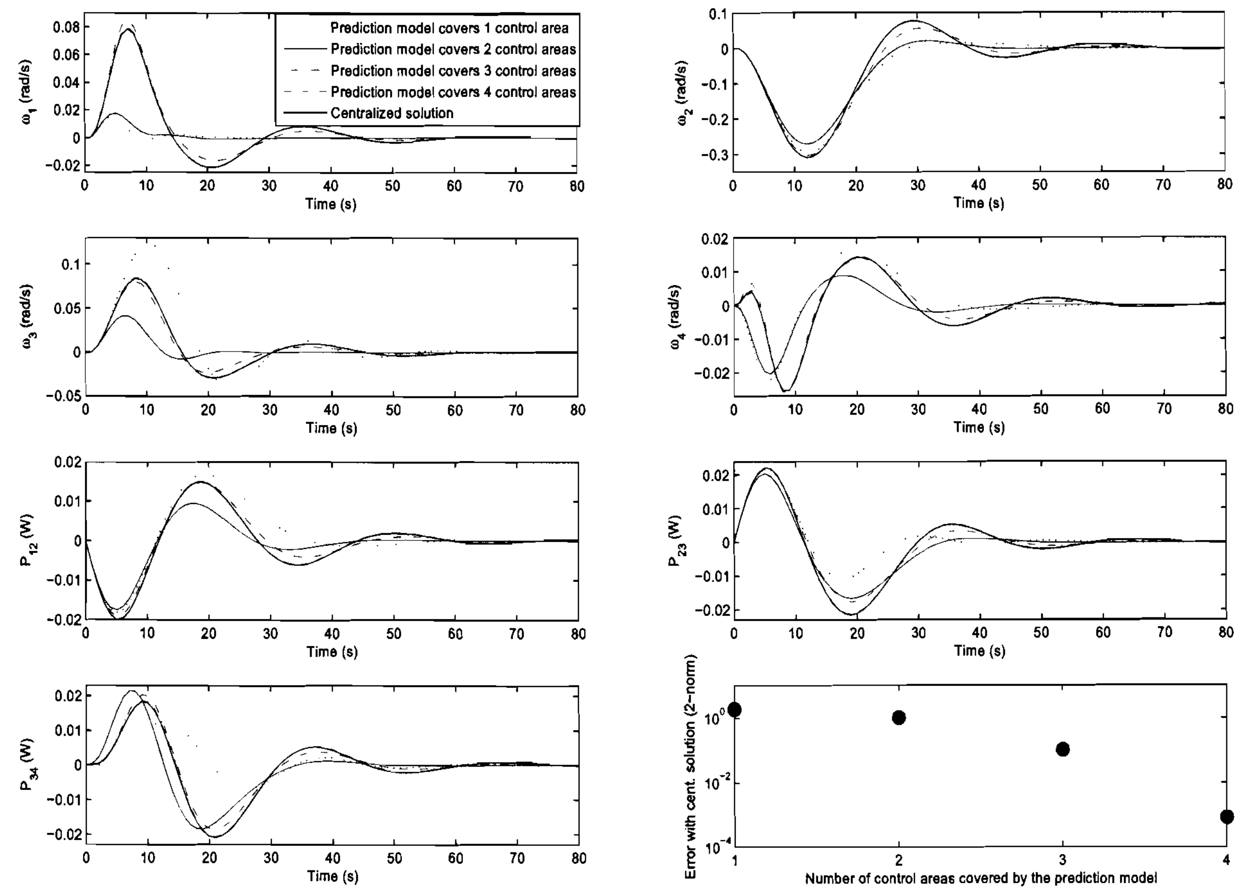

Fig. 11. Simulation results of a DI-MPC controlled power network with different sizes of the prediction model. The lower right figure gives the error compared to the centralized Pareto optimal solution.

the fact that there is no mathematical proof to guarantee convergence, stability and feasibility of this technique. Future research should focus on this problem, and search for either methods to guarantee convergence and stability or a posteriori verifiable conditions to check convergence and stability.

To summarize, a non-centralized MPC technique that is viable for real-life control of power systems should have the following characteristics: communication only with a small amount of controllers, improved state trajectory predictions (even at the price of iterations in between samples), ability to deal with coupling state constraints and guarantee of closed-loop stability.

ACKNOWLEDGMENTS

I would like to thank my supervisors dr. ir. Andrej Jokic, dr. ir. Mircea Lazar and Prof. dr. ir. Paul van den Bosch for their support and guidance during my master thesis project.

REFERENCES

[1] P. Kundur, *Power system stability and control*. McGraw-Hill, 1994.

[2] J. Frunt, "Distributed generation in the Dutch electricity market," 2006, MSc thesis, Eindhoven University of Technology, The Netherlands.

[3] E. Camponogara, "Controlling networks with collaborative nets," Ph.D. dissertation, Carnegie Mellon University, Pittsburgh, Pennsylvania, 2000.

[4] A. N. Venkat, "Distributed model predictive control: Theory and applications," Ph.D. dissertation, University of Wisconsin-Madison, Wisconsin-Madison, USA, 2006.

[5] A. Jokic, M. Lazar, and P. P. J. van den Bosch, "Price-based optimal control of power flow in energy distribution networks," in *Hybrid System Computation and Control Conf.*, Italy, 2007, pp. 315–328.

[6] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[7] G. C. Goodwin, M. M. Seron, and J. A. De Dona, *Constrained control and estimation An optimization approach*, ser. Communications and control engineering. Springer, 2005.

[8] E. Camponogara, D. Jia, B. H. Krogh, and S. Talukdar, "Distributed model predictive control," *IEEE Control Systems Magazine*, vol. 22, no. 1, pp. 44–52, 2002.

[9] A. N. Venkat, H. A. Hiskens, J. B. Rawlings, and S. J. Wright, "Distributed feedback MPC for power system control," in *45th IEEE Conf. on Decision and Control*, San Diego, 2006, pp. 4038–4045.

[10] T. Keviczky, F. Borrelli, and G. J. Balas, "Decentralized receding horizon control for large scale dynamically decoupled systems," *Automatica*, vol. 42, no. 12, pp. 2105–2115, 2006.

[11] A. Alessio and A. Bemporad, "Decentralized model predictive control of constrained linear systems," in *Proceedings European Control Conference*, Kos, Greece, 2007, pp. 2813–2818.

[12] W. B. Dunbar, "Distributed receding horizon control of dynamically coupled nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 52, no. 7, pp. 1249–1263, 2007.

[13] F. Blanchini, "Ultimate boundedness control for uncertain discrete-time systems via set-induced Lyapunov functions," *IEEE Transactions on Automatic Control*, vol. 39, no. 2, pp. 428–433, 1994.

APPENDIX

TABLE III

SIMULATION PARAMETERS

Sample time	$1s$
Simulation time	$200s$
Prediction horizon N	20
Iterations (FC-MPC)	1
States of subsystem 1 States of subsystem 2 States of subsystem 3 States of subsystem 4	$\Delta P_{V1}, \Delta P_{M1}, \Delta\omega_1$ $\Delta\delta_{12}, \Delta P_{V2}, \Delta P_{M2}, \Delta\omega_2$ $\Delta\delta_{23}, \Delta P_{V3}, \Delta P_{M3}, \Delta\omega_3$ $\Delta\delta_{34}, \Delta P_{V4}, \Delta P_{M4}, \Delta\omega_4$
Disturbance ΔP_{L_1} Disturbance ΔP_{L_2} Disturbance ΔP_{L_3} Disturbance ΔP_{L_4}	$0, \ \forall t$ $0, \ t < 10, \ \ +0.25, \ t \geq 10$ $0, \ t < 60, \ \ -0.25, \ t \geq 60$ $0, \ \forall t$
Constraint on ΔP_{ref}	$-0.5 \leq \Delta P_{ref} \leq 0.5$
Generator damping: D_1, D_2, D_3, D_4	$3, 0.275, 2, 2.75$
Generator inertia: J_1, J_2, J_3, J_4	$4, 40, 35, 10$
Speed regulation: r_1, r_2, r_3, r_4	$0.06, 0.14, 0.08, 0.06$
Governor time constant: $\tau_{G1}, \tau_{G2}, \tau_{G3}, \tau_{G4}$	$4, 25, 15, 5$
Turbine time constant: $\tau_{T1}, \tau_{T2}, \tau_{T3}, \tau_{T4}$	$5, 10, 20, 10$
Q_1, Q_2 Q_3, Q_4	$\text{diag}\,(0,0,5), \text{diag}\,(5,0,0,5)$ $\text{diag}\,(5,0,0,5), \text{diag}\,(5,0,0,5)$
R_1, R_2, R_3, R_4	$1, 1, 1, 1$