ASSESSMENT OF DECENTRALIZED MODEL PREDICTIVE CONTROL TECHNIQUES FOR POWER NETWORKS

Armand Damoiseaux
University of Technology
Eindhoven, The Netherlands
a.c.r.m.damoiseaux@student.tue.nl

Andrej Jokic
University of Technology
Eindhoven, The Netherlands
a.jokic@tue.nl

Mircea Lazar
University of Technology
Eindhoven, The Netherlands
m.lazar@tue.nl

Alessandro Alessio
University of Siena
Siena, Italy
alessio@dii.unisi.it

Paul van den Bosch
University of Technology
Eindhoven, The Netherlands
p.p.j.v.d.bosch@tue.nl

Ian Hiskens
University of Wisconsin-Madison
Wisconsin-Madison, USA
hiskens@engr.wisc.edu

Alberto Bemporad
University of Siena
Siena, Italy
bemporad@unisi.it

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 lies in 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 that can contribute to the successful decentralized implementation of MPC in real-life electrical power networks.

Keywords - Model predictive control, Decentralized control, Distributed control, Power systems.

1 INTRODUCTION

CURRENT power networks consist of large scale power generating units and automatic generation control (AGC) is used for real-time control of the system frequency and tie line 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 to 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 to power systems, due to the dynamic coupling present in a power network.

Among the research that focusses on non-centralized MPC, implementations for power system control have already been illustrated [3, 8, 9], differing in the requirements for computational power, data communication and model size. 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 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.

2 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 lies in 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 system 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 proces 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.

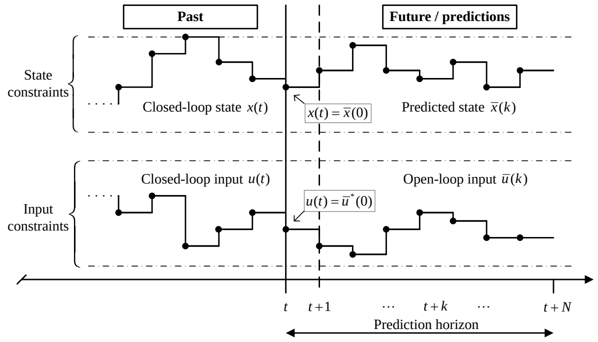

Figure 1: A graphical illustration of Model Predictive Control.

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 integer numbers.

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

Problem 2.1 At discrete time $t \in \mathbb{Z}_+$ let $x(t)$ and $N \geq 1$ be given, set $\bar{x}(0) := x(t)$ and solve

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

$$\begin{aligned} V_N(x, \bar{\mathbf{u}}_{[N-1]}) &= \sum_{k=0}^{N-1} \ell(\bar{x}(k), \bar{u}(k)) + F(\bar{x}(N)) \\ &= \sum_{k=0}^{N-1} \bar{x}^\top(k) Q \bar{x}(k) + \bar{u}^\top(k) R \bar{u}(k) \\ &\quad + \bar{x}^\top(N) P \bar{x}(N). \end{aligned} \tag{2b}$$

In Problem 2.1, $\bar{\mathbf{u}}_{[N-1]} = (\bar{u}(0), \ldots, \bar{u}(N-1))$ 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 $P = P^\top \succ 0$ weighs 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

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

[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.

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

$$\begin{aligned}\mathscr{U}_N(x) :=&\{\bar{\mathbf{u}}_{[N-1]} \in \mathbb{U}^N | \bar{x}(k) \in \mathbb{X}, \\ &k = 1, \ldots, N-1,\ \bar{x}(N) \in \mathbb{X}_f\},\end{aligned} \tag{4}$$

where $\mathbb{U}^N$ is the N-times Cartesian product, i.e. $\mathbb{U}^N := \mathbb{U} \times \ldots \times \mathbb{U}$. $\mathbb{U}$ is a compact subset of $\mathbb{R}^m$ and $\mathbb{X}$ is a closed subset of $\mathbb{R}^n$. These sets implement physical input and state constraints. Furthermore, 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. $u(t) := \bar{u}^*(0)$, 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] satisfying the following property:

$$\begin{aligned}\mathbb{X}_f \subseteq \mathcal{O}_\infty :=&\{x \in \mathbb{R}^n \mid K(A-BK)^k x \in \mathbb{U} \text{ and} \\ &(A-BK)^k x \in \mathbb{X},\ k = 0, \ldots, \infty\}.\end{aligned} \tag{5}$$

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

$$P = (A+BK)^\top P(A+BK) + K^\top RK + Q, \tag{6a}$$
$$K = -(R + B^\top PB)^{-1} B^\top PA. \tag{6b}$$

In [6] it is proven that system (1) in closed-loop with a predictive control law obtained by solving Problem 2.1 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 2.1 and closed-loop stability and necessitates new stabilization conditions. The following section presents possible solutions in this framework.

3 Description of the non-centralized MPC schemes

As already explained in the Introduction, the centralized implementation of the MPC methodology described in Section 2 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.

3.1 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 total system to be controlled is described by a discrete time state space model of the form:

$$x(t+1) = Ax(t) + Bu(t), \tag{7}$$

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 by an explicit transformation 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:

$$x_i = W_i^\top x, \qquad u_i = Z_i^\top u, \tag{8a}$$
$$Q_i = W_i^\top Q W_i, \qquad R_i = Z_i^\top R Z_i, \tag{8b}$$

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 (8a) each entry in x is in general assigned to one or more x_i and each entry in u is assigned to one or more u_i. This allows for definition of overlapping subsystems. For more information about the construction of these matrices, the reader is referred to [11]. 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 subsystem i the following finite horizon problem is defined:

Problem 3.1 DMPC

$$\begin{aligned}\mathscr{P}_{N,i}(x_i) : V^*_{N,i}(x_i) = \min_{\bar{\mathbf{u}}_{[N-1],i}} &\{V_{N,i}(x_i, \bar{\mathbf{u}}_{[N-1],i})... \\ &...|\bar{\mathbf{u}}_{[N-1],i} \in \mathscr{U}_{N,i}(x_i)\},\end{aligned} \tag{9a}$$

$$\begin{aligned}V_{N,i}(x_i, \bar{\mathbf{u}}_{[N-1],i}) &= \sum_{k=0}^{N-1} \ell_i(\bar{x}_i(k), \bar{u}_i(k)) + F_i(\bar{x}_i(N)) \\ &= \sum_{k=0}^{N-1} \bar{x}_i^\top(k) Q_i \bar{x}_i(k) + \bar{u}_i^\top(k) R_i \bar{u}_i(k) \\ &\quad + \bar{x}_i^\top(N) P_i \bar{x}_i(N).\end{aligned} \tag{9b}$$

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 $u \equiv u_i$, $\forall i$.

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

$$\bar{x}_i(k+1) = A_i\bar{x}_i(k) + B_i\bar{u}_i(k),\ k = 0,\dots,N-1, \tag{10a}$$

$$A_i = W_i^\top AW_i, \quad B_i = W_i^\top AZ_i, \tag{10b}$$

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 partially *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.

$$\bar{x}_i(0) := W_i^\top x(t). \tag{11}$$

The optimization problem (9a) employs the following feasibility set:

$$\begin{aligned}\mathscr{U}_{N,i}(x_i) :=\{&\bar{\mathbf{u}}_{[N-1],i} \in \mathbb{U}^N | \bar{u}_i(k) = K_i\bar{x}_i(k),\\ &k = N_u,\dots,N-1\},\end{aligned} \tag{12}$$

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 local* stability of each subsystem, a terminal penalty matrix P_i for each subsystem i can be defined following the centralized case as follows:

$$P_i = (A_i + B_iK_i)^\top P_i(A_i + B_iK_i) + K_i^\top R_iK_i + Q_i, \tag{13a}$$

$$K_i = -(R_i + B_i^\top P_iB_i)^{-1}B_i^\top P_iA_i. \tag{13b}$$

If $K_i \equiv 0$, as done in [11], then (13a) reduces to the Lyapunov equation, i.e. to $P_i = A_i^\top P_iA_i+Q_i$. Note that with $Q_i \succ 0$, this implies that each subsystem has to be open-loop stable, i.e. all eigenvalues of A_i must be within the unit circle. After all M controllers have calculated their local optimal control input sequence $\bar{\mathbf{u}}^*_{[N-1],i}$, the collection of all local inputs is applied to the global system (7), i.e.

$$u(t) = [\bar{u}^*_1(0),\dots,\bar{u}^*_i(0),\dots,\bar{u}^*_M(0)], \tag{14}$$

and 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 $k = 1,\dots,N$, i.e. in general the predicted state does not coincide with the real system state trajectory. This is a result of the fact that the prediction model (10) only approximates the real system and is initialized with partial state measurements, while the dynamic coupling between the subsystems is partially 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 can be assigned to one and only one x_i and each entry in u 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 checks *overall* stability of the entire system (7) in closed loop with the M decentralized MPC controllers, if the matrix P_i of the controller i is chosen as indicated in (13a). This *a posteriori* stability condition checks whether the sum of all cost functions is a Lyapunov function for the overall system, and is based on the explicit form of each MPC controller thereby creating a PWA system. Under certain conditions this reduces to a positive semidefiniteness check of a square $n\times n$ matrix. The drawback of this approach is that it has to be 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].

3.2 *Stability constrained distributed MPC (SC-DMPC)*

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

$$x(t+1) = Ax(t) + Bu(t), \tag{15}$$

are given by the following matrices:

$$A = \begin{pmatrix} A_{11} & \dots & A_{1i} & \dots & A_{1M} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ A_{i1} & \dots & A_{ii} & \dots & A_{iM} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ A_{M1} & \dots & A_{Mi} & \dots & A_{MM} \end{pmatrix}, \tag{16a}$$

$$B = \begin{pmatrix} B_{11} & \dots & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \dots & B_{ii} & \dots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & \dots & B_{MM} \end{pmatrix}, \tag{16b}$$

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 optimization problem.

Problem 3.2 SC-DMPC

$$\mathscr{P}_{N,i}(x_i) : V^*_{N,i}(x_i) = \min_{\bar{\mathbf{u}}_{[N-1],i}} \{V_{N,i}(x_i, \bar{\mathbf{u}}_{[N-1],i})...$$
$$...|\bar{\mathbf{u}}_{[N-1],i} \in \mathscr{U}_{N,i}(x_i)\}, \tag{17a}$$

$$\begin{aligned} V_{N,i}(x_i, \bar{\mathbf{u}}_{[N-1],i}) &= \sum_{k=0}^{N-1} \ell_i(\bar{x}_i(k), \bar{u}_i(k)) + F_i(\bar{x}_i(N)) \\ &= \sum_{k=0}^{N-1} \bar{x}_i^\top(k) Q_i \bar{x}_i(k) + \bar{u}_i^\top(k) R_i \bar{u}_i(k) \\ &\quad + \bar{x}_i^\top(N) P_i \bar{x}_i(N). \end{aligned} \tag{17b}$$

The SC-DMPC uses the following prediction model:

$$\bar{x}_i(k+1) = A_{ii}\bar{x}_i(k) + B_{ii}\bar{u}_i(k) + \sum_{j \neq i}^{M} A_{ij}\bar{x}_j(k). \tag{18}$$

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.

$$\bar{x}_j(k) := \bar{x}_j^*(k|t-1), \tag{19}$$

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.

$$\bar{x}_i(0) := x_i(t). \tag{20}$$

Note that $A_{ij} = 0$ if the the 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 sequence for Problem 3.2 is defined as follows:

$$\mathscr{U}_{N,i}(x_i) := \{\bar{\mathbf{u}}_{[N-1],i} \in \mathbb{U}^N |\ \|\bar{x}_i(1)\|_2^2 \leq \hat{l}_i\}, \tag{21}$$

where

$$\hat{l}_i := \max\{\|\bar{x}_i(1|t-1)\|_2^2, \|\bar{x}_i(0)\|_2^2\} - \beta_i \|x_i^1(0)\|_2^2, \tag{22}$$

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

$$\bar{x}_i(1|t-1) := A_{ii}\bar{x}_i(0) + B_{ii}\bar{u}_i(0) + \sum_{j \neq i}^{M} A_{ij}\bar{x}_j(0), \tag{23a}$$

$$\bar{x}_i(0) := x_i(t-1), \quad \bar{u}_i(0) := \bar{u}_i^*(t-1), \tag{23b}$$

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

$$P_i = (A_i + B_i K_i)^\top P_i (A_i + B_i K_i) + K_i^\top R_i K_i + Q_i, \tag{24a}$$

$$K_i = -(R_i + B_i^\top P_i B_i)^{-1} B_i^\top P_i A_i, \tag{24b}$$

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 (15), i.e.

$$u(t) = [\bar{u}_1^*(0), \ldots, \bar{u}_i^*(0), \ldots, \bar{u}_M^*(0)]. \tag{25}$$

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 still 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 (15), is in general smaller compared to DMPC, where dynamic coupling is (partially) neglected. 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 a local state measurement is required for initialization of 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 (21) and (22), is based on a controllable companion form. In [8] it is proven that this ensures the existence of a feasible control input. 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 collection of calculated control inputs comprises a feasible solution for the overall system. For more details on feasibility and stability, the reader is referred to [8].

3.3 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

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

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] $\bar{\mathbf{u}}_{[N-1],i}$. Due to the fact that 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 3.3 FC-MPC

$$\mathscr{P}^p_{N,i}(x) : V^{p*}_{N,i}(x) = \min_{\bar{\mathbf{u}}^p_{[N-1],i}} \{V^p_{N,i}(x, \bar{\mathbf{u}}^p_{[N-1],i})... \\ ...|\bar{\mathbf{u}}^p_{[N-1],i} \in \mathscr{U}^p_{N,i}(x)\}, \tag{26a}$$

$$V^p_{N,i}(x, \bar{\mathbf{u}}^p_{[N-1],i}) = \sum_{k=0}^{N-1} \ell(\bar{x}^p(k), \bar{u}^p_i(k)) + F(\bar{x}^p(N)) \\ = \sum_{k=0}^{N-1} \bar{x}^{p\top}(k) Q \bar{x}^p(k) + \bar{u}_i^{p\top}(k) R_i \bar{u}^p_i(k) \\ + \bar{x}^{p\top}(N) P \bar{x}^p(N). \tag{26b}$$

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

$$\bar{x}^p(k+1) = A\bar{x}^p(k) + B[\bar{u}^{i,p}_1(k), \ldots, \bar{u}^p_i(k), \ldots, \bar{u}^{i,p}_M(k)], \tag{27}$$

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

$$\bar{u}^{i,p}_j(k) := \bar{u}^{p-1}_j(k),\, k = 0, \ldots, N-1, \\ j = 1, \ldots, M,\ j \neq i, \tag{28}$$

where M denotes the total number of subsystems. The prediction model (27) is initialized with the current state of the system, i.e.

$$\bar{x}^p(0) := x(t), \qquad \forall p. \tag{29}$$

The feasible set is defined as:

$$\mathscr{U}^p_{N,i}(x) := \{\bar{\mathbf{u}}^p_{[N-1],i} \in \mathbb{U}^N\}, \qquad \forall p. \tag{30}$$

The terminal penalty matrix P used in (26b) is the solution of the unconstrained infinite horizon LQR problem, i.e.

$$P = (A + BK)^\top P(A + BK) + K^\top RK + Q \tag{31a}$$

$$K = -(R + B^\top PB)^{-1} B^\top PA. \tag{31b}$$

In [9], K is chosen equal to zero, which yields $P = A^\top PA + Q$. 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.
- $\bar{u}^0_i(k) := u^{p*}_i(k+1|t-1)$; 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$(\rho_i > \varepsilon)$

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

$$u(t) = [\bar{u}^p_1(0), \ldots, \bar{u}^p_i(0), \ldots, \bar{u}^p_M(0)]. \tag{32}$$

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 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].

4 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.

4.1 Test network and simulation scenario

A schematic representation of the test power system is depicted in Figure 2.

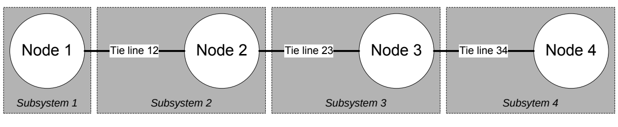

Figure 2: Schematic representation of the power network.

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

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

The system consists of 4 control areas, with the dynamics of each area given by the following standard model:

$$\frac{d\Delta\omega_i}{dt} = \frac{1}{J_i}(\Delta P_{M_i} - D_i\Delta\omega_i - \Delta P_{tie}^{ij} - \Delta P_{L_i}), \tag{33a}$$

$$\frac{d\Delta P_{M_i}}{dt} = \frac{1}{\tau_{T_i}}(\Delta P_{V_i} - \Delta P_{M_i}), \tag{33b}$$

$$\frac{d\Delta P_{V_i}}{dt} = \frac{1}{\tau_{G_i}}(\Delta P_{ref_i} - \Delta P_{V_i} - \frac{1}{r_i}\Delta\omega_i), \tag{33c}$$

$$\frac{d\Delta P_{tie}^{ij}}{dt} = b_{ij}(\Delta\omega_i - \Delta\omega_j), \tag{33d}$$

$$\Delta P_{tie}^{ji} = -\Delta P_{tie}^{ij}. \tag{33e}$$

These equations describe the dynamics of a generator and a tie line connecting the generators, and 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.

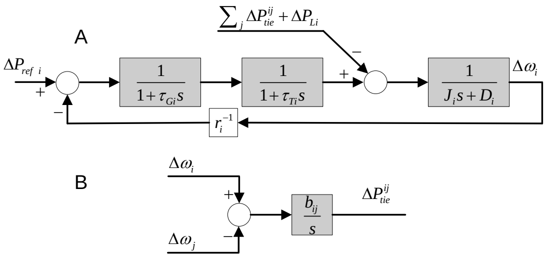

Figure 3: Graphical representation of the dynamics of a generator (A) and tie line (B).

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 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 by $\frac{d\Delta P_{ref_i}}{dt} = K_i(-B_i\Delta\omega_i - \Delta P_{tie}^{ij})$, with K_i and B_i 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 the time instant zero, the system is in a steady state with the frequency and the 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. 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.

4.2 Simulation results

The simulations 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 1 shows the settling time [6] of the weighted states and the performance in the 1-norm, i.e. $\sum_{t=0}^{200} |Qx(t)|$.

	Settling time (s)							Norm
	ω_1	ω_2	ω_3	ω_4	P_{12}	P_{23}	P_{34}	
AGC	-	-	-	-	-	-	-	19.25
MPC	112	109	105	94	124	116	125	7.03
FC-MPC	111	109	104	94	124	116	125	7.05
DMPC	130	132	148	139	177	199	151	9.55
SC-DMPC	117	116	124	125	152	186	127	10.99

Table 1: Performance figures, settling time and 1-norm

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. 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.

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:

$$\min_x \; x^\top Hx, \tag{34a}$$

$$\text{subject to } Ax \leq B, \tag{34b}$$

$$A_{eq}x = B_{eq}. \tag{34c}$$

The computational complexity depends on the size of A, B, A_{eq}, B_{eq} and H. The size of the matrices A and A_{eq}

[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.

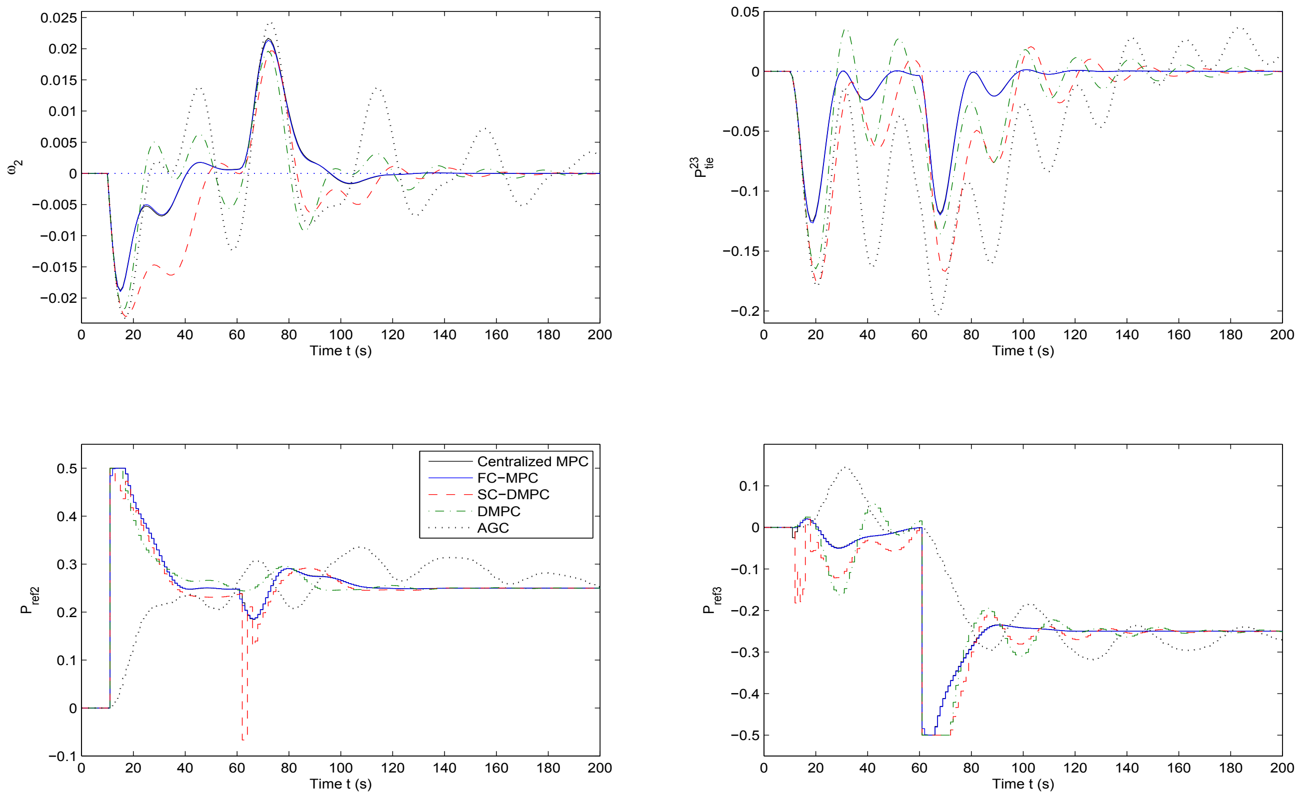

Figure 4: Simulation results of the considered control techniques

for this specific problem is listed in Table 2 for each algorithm (the size of B, B_{eq} and H can be derived from these matrices).

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

Table 2: Sizes of optimization problem for QP solver

The computational 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.

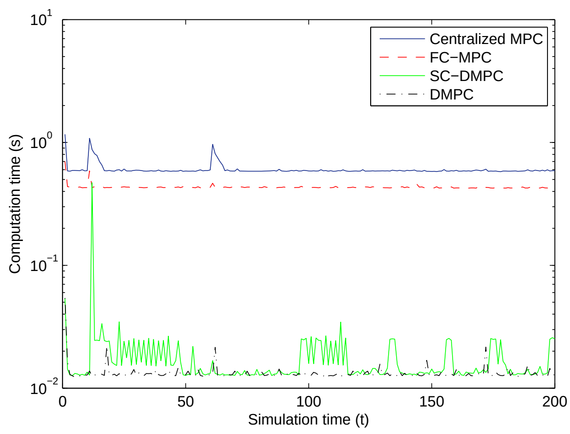

Figure 5: The computation time for all algorithms.

The results show that from a computational point of view DMPC and SC-DMPC are preferred as 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 problem 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.

5 Conclusions

In this section we summarize some constructive recommendations for the future implementation 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 a non-centralized MPC control action is not a feasible goal for current power networks. Future advances in communications and increasing processing power might bring this goal closer to realization. Currently, however communication beyond directly neighboring 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. A significant problem that is currently not solved within existing non-centralized MPC schemes is posed by coupling state constraints. This issue is of paramount importance to power systems, where coupling state constraints are inherent. Clearly, achieving better estimates of the real-system state trajectories would also be useful for dealing with coupling state constraints.

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 direct neighboring subsystems, 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.

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 electrical energy distribution networks," in *Hybrid System Computation and Control Conference*, Pisa, 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 output feedback MPC for power system control," in *45th IEEE Conference on Decision and Control*, San Diego, CA, 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.

6 Appendix

Sample time	$1s$
Simulation time	$200s$
Prediction horizon N	20
Iterations (FC-MPC)	1
States of subsystem 1	$\Delta P_{V1}, \Delta P_{M1}, \Delta\omega_1$
States of subsystem 2	$\Delta\delta_{12}, \Delta P_{V2}, \Delta P_{M2}, \Delta\omega_2$
States of subsystem 3	$\Delta\delta_{23}, \Delta P_{V3}, \Delta P_{M3}, \Delta\omega_3$
States of subsystem 4	$\Delta\delta_{34}, \Delta P_{V4}, \Delta P_{M4}, \Delta\omega_4$
Disturbance ΔP_{L_1}	$0,\ \forall t$
Disturbance ΔP_{L_2}	$0,\ t < 10,\ \ +0.25,\ t \geq 10$
Disturbance ΔP_{L_3}	$0,\ t < 60,\ \ -0.25,\ t \geq 60$
Disturbance ΔP_{L_4}	$0,\ \forall t$
Constraint on ΔP_{ref}	$-0.5 \leq \Delta P_{ref} \leq 0.5$
Generator friction: D_1, D_2, D_3, D_4	$3, 0.275, 2, 2.75$
Generator inertia: M_1, M_2, M_3, M_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	$\text{diag}\,(0,0,5), \text{diag}\,(5,0,0,5)$
Q_3, Q_4	$\text{diag}\,(5,0,0,5), \text{diag}\,(5,0,0,5)$
R_1, R_2, R_3, R_4	$1, 1, 1, 1$