Multi-agent Model Predictive Control for Transportation Networks: Serial versus Parallel Schemes

R.R. Negenborn, B. De Schutter, J. Hellendoorn

Delft University of Technology, Delft, The Netherlands

Multi-agent model predictive control for transportation networks: Serial versus parallel schemes

R.R. Negenborn [a,*] B. De Schutter [a,1] J. Hellendoorn [a]

[a]*Delft University of Technology, Delft Center for Systems and Control,*
Mekelweg 2, 2628 CD Delft, The Netherlands

Abstract

We consider the control of large-scale transportation networks, like road traffic networks, power distribution networks, water distribution networks, etc. Control of these networks is often not possible from a single point by a single intelligent control agent; instead control has to be performed using multiple intelligent agents. We consider multi-agent control schemes in which each agent employs a model-based predictive control approach. Coordination between the agents is used to improve decision making. This coordination can be in the form of parallel or serial schemes. We propose a novel serial coordination scheme based on Lagrange theory and compare this with an existing parallel scheme. Experiments by means of simulations on a particular type of transportation network, viz., an electric power network, illustrate the performance of both schemes. It is shown that the serial scheme has preferable properties compared to the parallel scheme in terms of the convergence speed and the quality of the solution.

Key words: Multi-agent control, model predictive control, transportation networks, power systems.

1. Introduction

1.1. *Transportation networks and their control*

Transportation networks, like road traffic networks, power distribution networks, water distribution networks, gas networks, etc. are usually large in size, consist of multiple subnetworks, have many actuators and sensors, and exhibit complex dynamics. These transportation networks can be considered at a generic level, at which commodity is brought into the network at sources, flows over links to sinks, and is influenced in its way of flowing by elements inside the network. The similarities between several types of transportation networks are the motivation for studying these networks in a generic way.

Typical control goals for transportation networks involve avoiding congestion of links, maximizing throughput, minimizing costs of control inputs, etc. In the daily operation of transportation networks, network operators have to adjust the actuators in the network to meet these control objectives. Control from a single point by a single, centralized, control agent is often not possible due to technical or commercial issues. Technical issues arise from, e.g., communication delays and too high computational requirements. Some commercial issues are, e.g., unavailability of information from one network operator to another, restricted control access, and costs of sensors. Moreover, robustness and reliability of the network may become a problem in single-agent control, e.g., when the single control agent breaks down.

For these reasons, transportation networks typically have to be operated using a multi-agent, or distributed, control approach (Weiss, 2000; Sycara, 1998; Siljak, 1991). In such an approach the overall network consists of multiple smaller subnetworks. Each of the subnetworks is controlled by an agent with only limited information gathering and processing skills and moreover limited action capabilities. It is noted that in particular due to the commercial issues multi-agent control is not only restricted to networks that span large geographical areas, but may also be used for control of relatively small networks. E.g., in power networks typically the topology and system parameters of the network in one country are not made available to surrounding countries, making multi-agent control necessary.

[*] Corresponding author. Tel.: +31 1527 86524; fax: +31 1527 86679.
Email address: `r.r.negenborn@tudelft.nl` (R.R. Negenborn).

[1] Bart De Schutter is also with the Marine and Transport Technology department of Delft University of Technology.

1.2. *Multi-agent model predictive control*

1.2.1. *Model predictive control*

To determine which actions to take, an intelligent control agent typically has some sort of model of the system it controls, a set of constraints under which it has to perform the control, and an objective function describing the goals of the control. Using the model and the constraints the agent can to some extent predict the consequences of its actions over a certain time span in the future. Using in addition to this the objective function, the agent can determine those actions that are optimal with respect to its predictions. When such an approach to control is used at each control step, i.e., in a receding horizon fashion, it is called *model predictive control* (MPC) (Maciejowski, 2002; Mayne et al., 2000).

The major advantage of MPC is its straightforward design procedure. Given a model of the system, hard constraints can be incorporated directly as inequalities and one only needs to set up an objective function reflecting the control goal. Soft constraints can also be accounted for in the objective by using penalties for violations. Additional advantages of MPC are its explicit way of integrating constraints and its straightforward way of integrating forecasts. E.g., for transportation networks MPC provides a convenient way to include capacity limits on links, maximums on queue lengths, measurements from upstream sensors, profiles of demands, etc.

1.2.2. *Single-agent MPC*

In a single-agent setting, MPC has shown successful application in the process industry over the last decades (Camacho and Bordons, 1995; Morari and Lee, 1999), and is now gaining increasing attention in many other fields, like food processing, automotive, and aerospace (Qin and Badgewell, 1997), and power networks (Geyer et al., 2003), road traffic networks (Kotsialos et al., 2006; Hegyi et al., 2005), sewer networks (Marinaki and Papageorgiou, 2001), water networks (Wahlin, 2004), and railway networks (De Schutter et al., 2002). MPC thus has shown to be a promising control strategy, when a single-agent, centralized, control scheme can be implemented. However, when this is not the case, due to technical or commercial reasons, a multi-agent MPC scheme has to be employed.

1.2.3. *Multi-agent MPC*

The theoretical research in multi-agent MPC started in the 90s (Aicardi et al., 1992; Acar, 1992; Katebi and Johnson, 1997; Jia and Krogh, 2001, 2002; Camponogara et al., 2002), with applications to water distribution systems (Georges, 1999), delivery canals (Sawadogo et al., 1998), irrigation systems (El Fawal et al., 1998), multi-reach canals (Gomez et al., 1998), dynamic routing (Baglietto et al., 1999), cascading failures in power networks (Hines et al., 2005), distributed vehicle coordination (Dunbar and Murray, 2006), and distributed emergency voltage control (Beccuti and Morari, 2006).

In multi-agent MPC it is usually assumed that the system to be controlled has been divided into subsystems, and that each subsystem has been assigned an agent. Each of the agents uses MPC to determine its actions. In particular, at each control cycle, each agent performs the following:

(i) It obtains a measurement of the current state of its subsystem, and receives information from other agents.
(ii) It solves an optimization problem that finds over a certain horizon the actions that result in the best subsystem behavior according to a specified objective. This typically involves communication.
(iii) It implements the solution of the optimization problem of step ii.
(iv) It moves on to the next control cycle.

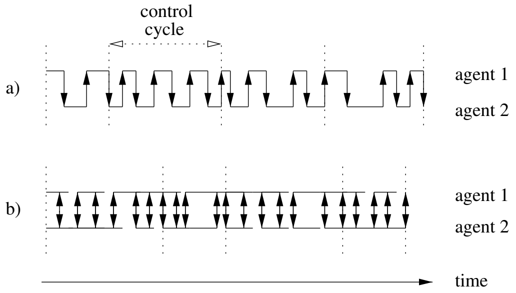

Fig. 1. Two types of local computation and communication schemes between two agents: a) serial, b) parallel. Solid arrows indicate information exchange. Dotted arrows indicate time spans. Vertical dotted lines indicate the end of a control cycle. Horizontal solid lines indicate local computations being performed. A control cycle consists of a number of iterations, in each of which each agent performs a single step.

We focus on the challenge in implementing step ii of such a scheme. The actions that an agent takes influence both the evolution of its own subsystem, and the evolution of the subsystems connected to its subsystem. Since the agents in a multi-agent setting usually have no global overview and can only access a relatively small number of sensors and actuators, predicting the evolution of a subsystem over a horizon involves even more uncertainty than when a single agent is employed. Therefore, usually communication is used to reduce this uncertainty, since this allows agents to inform one another about their plans. Typically, at each control cycle, the agents perform a number of *iterations*, within which each agent performs a *local computation* and *communication step*. The agents can in this way take into account the plans of other agents and anticipate any undesirable situation. Through communication agents may obtain agreement on taking actions that yield a good overall performance.

1.2.4. *Parallel versus serial schemes*

There are many ways in which a multi-agent MPC scheme can be implemented (Negenborn et al., 2006). For a given multi-agent MPC scheme, the quality of the solution that the agents determine and the convergence and rate of convergence to this solution depends on various aspects, e.g., the particular implementation of the scheme, the way in which the agents perform communication and local computations, the way in which information received from other agents is used, etc. In this paper we focus on the second point, for which we distinguish between schemes that work in *parallel* and schemes that work in *serial*, see Fig. 1. In the literature on multi-agent MPC mainly parallel schemes have been proposed, e.g., (Hines et al., 2005; Camponogara et al., 2002; El Fawal et al., 1998; Georges, 1999), in which all agents simultaneously perform a local step, then exchange information, then solve their next local step, and so on. In this paper we propose a novel serial scheme, in which only one agent at a time performs a local step, sends information to a next agent, after which this next agent performs a local computation step, sends information to a next agent, etc. Only after all agents have made a local step, the next round of local steps is started. We compare the serial scheme with a parallel scheme and assess the performance of both schemes experimentally. In experiments on a particular type of transportation network, viz., a power network, we show that the proposed serial approach has preferable properties in terms of the convergence speed and the quality of the solution.

1.3. *Outline*

This paper is organized as follows. In Section 2 we formalize the control setting as consisting of interconnected model predictive control problems. In Section 3 we develop a general multi-agent MPC scheme for dealing with the interconnections between the control problems. In Section 4 we discuss an existing parallel implementation of this scheme and propose a novel serial implementation. In Section 5 we experimentally compare and assess the performance of both schemes on a power network.

2. Control setting

Assume that a transportation network is given with a partitioning into n subnetworks, each controlled by a control agent that has a dynamical model of its subnetwork.

2.1. *Model of subnetwork dynamics*

Let the dynamics of subnetwork i be given by a deterministic linear discrete-time time-invariant model (possibly obtained after symbolic or numerical linearization of a nonlinear model), with noise-free outputs:

$$\begin{aligned} x_{i,k+1} &= A_i x_{i,k} + B_{1,i} u_{i,k} + B_{2,i} d_{i,k} + B_{3,i} v_{i,k} \\ y_{i,k} &= C_i x_{i,k} + D_{1,i} u_{i,k} + D_{2,i} d_{i,k} + D_{3,i} v_{i,k} \end{aligned} \qquad (1)$$

where at time step k, for subnetwork i, $x_{i,k} \in \mathbb{R}^{n_{i,x}}$ are local states, $u_{i,k} \in \mathbb{R}^{n_{i,u}}$ are local inputs, $d_{i,k} \in \mathbb{R}^{n_{i,d}}$ are local known disturbances, $y_{i,k} \in \mathbb{R}^{n_{i,y}}$ are local outputs, $v_{i,k} \in \mathbb{R}^{n_{i,v}}$ are remaining variables influencing the local dynamical states and outputs, and $A_i \in \mathbb{R}^{n_{i,x} \times n_{i,x}}, B_{1,i} \in \mathbb{R}^{n_{i,x} \times n_{i,u}}, B_{2,i} \in \mathbb{R}^{n_{i,x} \times n_{i,d}}, B_{3,i} \in \mathbb{R}^{n_{i,x} \times n_{i,v}}$, $C_i \in \mathbb{R}^{n_{i,y} \times n_{i,x}}, D_{1,i} \in \mathbb{R}^{n_{i,y} \times n_{i,u}}, D_{2,i} \in \mathbb{R}^{n_{i,y} \times n_{i,d}}, D_{3,i} \in \mathbb{R}^{n_{i,y} \times n_{i,v}}$ determine how the different variables influence the local state and output of subnetwork i. Note that for completeness inputs $u_{i,k}$ are also allowed to influence outputs $y_{i,k}$ at time k. Such a situation with direct feed-through terms typically appears when algebraic relations are linearized, e.g., when linearizing equations describing instantaneous (power) flow distributions.

The $v_{i,k}$ variables appear due to the fact that a subnetwork is connected to other subnetworks. If $v_{i,k}$ is known by agent i, this agent can compute the dynamics of subnetwork i independently of the other subnetworks.

2.2. *Model predictive control of a single subnetwork*

Assume for now that the control agent of subnetwork i operates individually and that it therefore does not communicate with other agents. The agent employs MPC to determine which actions to take. In MPC, an agent determines its local inputs by computing over a prediction horizon of N steps optimal inputs according to an objective function, subject to a model of the subnetwork and additional constraints. For notational convenience, in the following, a tilde over a variable is used to denote variables over the horizon for the overall network, e.g., i.e., $\tilde{a}_k = [\ a_k^{\mathrm{T}}, \ldots, a_{k+N-1}^{\mathrm{T}}\ ]^{\mathrm{T}}$, or for a particular subnetwork i, e.g., $\tilde{a}_{i,k} = [\ a_{i,k}^{\mathrm{T}}, \ldots, a_{i,k+N-1}^{\mathrm{T}}\ ]^{\mathrm{T}}$.

Given the measured initial local state [2] at time k as $x_{i,0}$, local known disturbances over the horizon as $\tilde{d}_{i,0}$, and locally predicted influences of the rest of the network over the prediction horizon as $\tilde{v}_{i,0}$, the following optimization problem is solved by agent i:

$$\min_{\tilde{u}_{i,k}} J_{\mathrm{local},i}(\tilde{x}_{i,k+1}, \tilde{u}_{i,k}, \tilde{y}_{i,k}) = \sum_{l=0}^{N-1} J_{\mathrm{stage},i}(x_{i,k+1+l}, u_{i,k+l}, y_{i,k+l}) \qquad (2)$$

subject to

$$\begin{aligned} x_{i,k+1+l} &= A_i x_{i,k+l} + B_{1,i} u_{i,k+l} + B_{2,i} d_{i,k+l} + B_{3,i} v_{i,k+l} && (3) \\ y_{i,k+l} &= C_i x_{i,k+l} + D_{1,i} u_{i,k+l} + D_{2,i} d_{i,k+l} + D_{3,i} v_{i,k+l} && (4) \\ & \qquad \text{for } l = 0, \ldots, N-1 \\ x_{i,k} &= x_{i,0} && (5) \\ \tilde{d}_{i,k} &= \tilde{d}_{i,0} && (6) \\ \tilde{v}_{i,k} &= \tilde{v}_{i,0}, && (7) \end{aligned}$$

where $J_{\mathrm{stage},i}(\cdot)$ is a twice differentiable (e.g., quadratic) function that gives the cost per prediction step given a certain local state, local input, and local output. A typical choice for the stage cost is

$$J_{\mathrm{stage},i}(\tilde{x}_{i,k+1}, \tilde{u}_{i,k}, \tilde{y}_{i,k}) = \begin{bmatrix} \tilde{x}_{i,k+1} \\ \tilde{u}_{i,k} \\ \tilde{y}_{i,k} \end{bmatrix}^{\mathrm{T}} Q \begin{bmatrix} \tilde{x}_{i,k+1} \\ \tilde{u}_{i,k} \\ \tilde{y}_{i,k} \end{bmatrix} + f^{\mathrm{T}} \begin{bmatrix} \tilde{x}_{i,k+1} \\ \tilde{u}_{i,k} \\ \tilde{y}_{i,k} \end{bmatrix},$$

where Q and f are a weighting matrix and vector respectively. After agent i has solved the optimization problem and found the actions over the horizon, it implements the actions determined until the next control cycle, waits for the physical subnetwork to transition to a new state, and starts the next control cycle.

[2] The measured initial local state is in this case the exact initial local state, since no measurement noise is considered.

We assumed that the agent does not use communication and that it can locally predict the influence of the rest of the network over the prediction horizon $\tilde{v}_{i,k}$, included in the control problem as (7). However, agent i cannot know this influence *a priori*, since actions taken by agent i influence the dynamics of its own subnetwork and therefore also the dynamics of a neighboring subnetwork, which thus changes $\tilde{v}_{i,k}$. Thus constraint (7) cannot be added explicitly, but has to be dealt with through the interconnecting constraints between control problems and communication between agents that enforces these interconnecting constraints.

2.3. *Interconnected control problems*

The interconnections between control problems are modeled using so-called *interconnecting variables*. A particular variable of the control problem of agent i is an interconnecting variable with respect to the control problem of agent j if the variable of agent i refers to the same quantity as a variable in the control problem of agent j. E.g., a flow going from subnetwork i into subnetwork j is represented with an interconnecting variable in the control problems of both agents.

Given the interconnecting variables of two agents referring to the same quantity, it is convenient to define one of these variables as an interconnecting *input* variable and the other as an interconnecting *output* variable. On the one hand, an interconnecting input variable $w_{\text{in},ji}$ of the control problem of agent i with respect to agent j can be seen as an input caused by agent j on the control problem of agent i. On the other hand, an interconnecting output variable $w_{\text{out},ij}$ of the control problem of agent j with respect to the control problem of agent i can be seen as the influence that agent j has on the control problem of agent i. In general the interconnecting variables can come from any domain, in the following, however, we consider interconnecting variables $w_{\text{in},ji} \in \mathbb{R}^{n_{ji,w_{\text{in}}}}, w_{\text{out},ji} \in \mathbb{R}^{n_{ji,w_{\text{out}}}}$.

Define the interconnecting inputs and outputs for agent i as

$$w_{\text{in},i} = \tilde{v}_{i,k} \tag{8}$$

$$w_{\text{out},i} = E_i \left[\tilde{x}^{\mathrm{T}}_{i,k+1} \; \tilde{u}^{\mathrm{T}}_{i,k} \; \tilde{y}^{\mathrm{T}}_{i,k} \right]^{\mathrm{T}}, \tag{9}$$

where E_i is an interconnecting-output selection matrix that contains zeros everywhere, except for a single 1 per row corresponding to a local variable that relates to an interconnecting-input variable of another agent.

Remark 2.1 *For the sake of simplicity of notation the subscript k for the time step and the tilde for variables of the prediction horizon are not used for the interconnecting variables.*

The variables $w_{\text{in},i}, w_{\text{out},i}$ are partitioned such that

$$w_{\text{in},i} = \left[w^{\mathrm{T}}_{\text{in},j_{i,1}i}, \ldots, w^{\mathrm{T}}_{\text{in},j_{i,m_i}i} \right]^{\mathrm{T}} \tag{10}$$

$$w_{\text{out},i} = \left[w^{\mathrm{T}}_{\text{out},j_{i,1}i}, \ldots, w^{\mathrm{T}}_{\text{out},j_{i,m_i}i} \right]^{\mathrm{T}}, \tag{11}$$

where $\mathcal{N}_i = \{j_{i,1}, \ldots, j_{i,m_i}\}$ is the set of indexes of the m_i subnetworks connected to subnetwork i, i.e., the set of neighbors of subnetwork i. The interconnecting inputs to the control problem of agent i with respect to agent j must be equal to the interconnecting outputs from the control problem of agent j with respect to agent i, since the variables of both control problems model the same quantity. For agent i this thus gives rise to the following *interconnecting constraints*:

$$w_{\text{in},ji} = w_{\text{out},ij} \tag{12}$$

$$w_{\text{out},ji} = w_{\text{in},ij}, \tag{13}$$

for $j \in \mathcal{N}_i$.

An interconnecting constraint cannot be added explicitly to the control problems of any of the individual agents, since each interconnecting constraint depends on variables of two different control problems. Instead the agents use communication to determine in an iterative way which values to give to the interconnecting inputs and outputs.

3. General multi-agent MPC scheme

One way for agent i to deal with its interconnecting constraints is to just ignore each neighboring agent $j \in \mathcal{N}_i$ and simply assume some values for the interconnecting outputs of that agent j, which essentially means solving problem (2). However, since the actions that an agent computes are optimal only with respect to the predicted values of the interconnecting input variables $w_{\text{in},ji}$ for all $j \in \mathcal{N}_i$, just assuming some values for the interconnecting output variables $w_{\text{out},ij}$ of agent j introduces high uncertainty, potentially deteriorating the performance of the control. To reduce this uncertainty agent i has to come to an agreement with agent $j \in \mathcal{N}_i$ on the values of its interconnecting output variables $w_{\text{out},ij}$. Each agent i obtains agreement through iterations that inform the neighboring agents $j \in \mathcal{N}_i$ about what agent i prefers the values of interconnecting inputs to be.

To obtain this agreement, an agent i does not only compute optimal local variables for its own subnetwork, but also optimal interconnecting input variables $w_{\text{in},ji}$. Moreover, the other agents $j \in \mathcal{N}_i$ compute both their optimal local variables and optimal interconnecting output variables $w_{\text{out},ij}$. Through exchange of these desired interconnecting variables, the values of the interconnecting output and input variables should converge to each other, and a set of local inputs that is overall optimal should be found.

A general scheme that implements these ideas is obtained in three steps: 1) formulating the combined overall control problem, i.e., aggregating the subproblems including the interconnecting constraints; 2) constructing an augmented Lagrange formulation by replacing each interconnecting constraint with an additional linear cost term, based on Lagrange multipliers, and a quadratic penalty term (Boyd and Vandenberghe, 2004; Bertsekas, 1982); 3) decomposing this formulation back into subproblems for each agent.

3.1. *Combined overall control problem*

We define the combined overall control problem as the problem formed by the aggregation of the local control problems without assuming the influence from the rest of the network formulated through equation (7) know, but including the definition of the interconnecting inputs and outputs (8)–(9) and the interconnecting constraints (12)–(13), i.e.,

$$\min_{\tilde{x}_{1,k+1},\tilde{u}_{1,k},\tilde{y}_{1,k},\ldots,\tilde{x}_{n,k+1},\tilde{u}_{n,k},\tilde{y}_{n,k}} \sum_{i=1}^{n} J_{\text{local},i}(\tilde{x}_{i,k+1},\tilde{u}_{i,k},\tilde{y}_{i,k},) \tag{14}$$

subject to, for $i = 1,\ldots,n$,

$$w_{\text{in},j_{i,1}i} = w_{\text{out},ij_{i,1}} \tag{15}$$

$$\vdots$$

$$w_{\text{in},j_{i,m_i}i} = w_{\text{out},ij_{i,m_i}} \tag{16}$$

and the dynamics (3)–(4) of subnetwork i over the horizon, and the initial constraints (5)–(6) of subnetwork i. Note that it is sufficient to include in the combined overall control problem formulation only the interconnecting input constraints (8) for each agent i, since the interconnecting output constraints (9) of agent i will also appear as interconnecting input constraints of its neighboring agents.

3.2. *Augmented Lagrange formulation*

The overall control problem (14) is not separable into subproblems using only local variables $\tilde{x}_{i,k+1},\tilde{u}_{i,k},\tilde{y}_{i,k}$ of one agent i alone due to the interconnecting constraints (15)–(16). In order to deal with the interconnecting constraints, an augmented Lagrangian formulation of this problem can be formulated (Boyd and Vandenberghe, 2004; Bertsekas, 1982). Using such an approach, the interconnecting constraints are removed from the constraint set and added to the objective function in the form of additional linear cost terms, based on Lagrange multipliers, and additional quadratic terms. The augmented Lagrange function is defined as

$$\begin{aligned} &L(\tilde{x}_{1,k+1},\tilde{u}_{1,k},\tilde{y}_{1,k},\ldots,\tilde{x}_{n,k+1},\tilde{u}_{n,k},\tilde{y}_{n,k},\lambda_{\text{in},j_{1,1}1},\ldots,\lambda_{\text{in},j_{n,m_n}n}) \\ &= \sum_{i=1}^{n} \Bigg(J_{\text{local},i}\left(\tilde{x}_{i,k+1},\tilde{u}_{i,k},\tilde{y}_{i,k},\right) \\ &\qquad + \sum_{j\in\mathcal{N}_i} \left(\lambda_{\text{in},ji}(w_{\text{in},ji} - w_{\text{out},ij}) + \frac{c}{2}\left\| w_{\text{in},ji} - w_{\text{out},ij}\right\|_2^2 \right) \Bigg), \end{aligned} \tag{17}$$

where c is a positive constant and $\lambda_{\text{in},ji}$ is the Lagrange multiplier associated with the interconnecting constraint $w_{\text{in},ji} = w_{\text{out},ji}$.

By duality theory (Boyd and Vandenberghe, 2004; Bertsekas, 1982), the resulting optimization problem follows as maximization over the Lagrange multipliers while minimizing over the other variables,

$$\max_{\substack{\lambda_{\text{in},j_{1,1}1},\\ \vdots \\ \lambda_{\text{in},j_{n,m_n}n}}} \min_{\substack{\tilde{x}_{1,k+1},\tilde{u}_{1,k},\tilde{y}_{1,k},\\ \vdots \\ \tilde{x}_{n,k+1},\tilde{u}_{n,k},\tilde{y}_{n,k}}} L(\tilde{x}_{1,k+1},\tilde{u}_{1,k},\tilde{y}_{1,k},\ldots,\tilde{x}_{n,k+1},\tilde{u}_{n,k},\tilde{y}_{n,k}, \\ \lambda_{\text{in},j_{1,1}1},\ldots,\lambda_{\text{in},j_{n,m_n}n}), \tag{18}$$

subject to, for $i = 1,\ldots,n$, the dynamics (3)–(4) of subnetwork i over the horizon, and the initial constraints (5)–(6) of subnetwork i.

Under convexity assumptions on the objective functions and affinity of the subnetwork model constraints it can be proved that a minimum of the original problem (14) can be found iteratively through repeatedly solving of the minimization part of (18) for fixed Lagrange multipliers, followed by updating of the Lagrange multipliers using the solution of the minimization, until the Lagrange multipliers do not change anymore from one iteration to the next (Bertsekas, 1982).

3.3. *Distributing the solution approach*

The iterations to compute the solution of the combined overall control problem based on the augmented Lagrange formulation (17) include quadratic terms and can therefore not directly be distributed over the agents. To deal with this the nonseparable problem (17) can be approximated by solving n separated problems of the form:

$$\min_{\substack{\tilde{x}_{i,k+1},\tilde{u}_{i,k},\tilde{y}_{i,k},\\ w_{\text{in},j_{i,1}i},\ldots,w_{\text{in},j_{i,m_i}i},\\ w_{\text{out},j_{i,1}i},\ldots,w_{\text{out},j_{i,m_i}i}}} J_{\text{local},i}\left(\tilde{x}_{i,k+1},\tilde{u}_{i,k},\tilde{y}_{i,k}\right) + \sum_{j\in\mathcal{N}_i} J_{\text{inter},i}\left(w_{\text{in},ji},w_{\text{out},ji},\lambda^{(s)}_{\text{in},ji},\lambda^{(s)}_{\text{out},ij}\right), \tag{19}$$

subject to the dynamics (3)–(4) of subnetwork i over the horizon, and the initial constraints (5)–(6) of subnetwork i, where the additional cost term $J_{\text{inter},i}(\cdot)$ deals with the interconnecting variables. At iteration (s), the variables $\lambda^{(s)}_{\text{in},ji}$ are the Lagrange multipliers computed by agent i for its interconnecting constraints $w_{\text{in},ji} = w_{\text{out},ij}$, and the variables $\lambda^{(s)}_{\text{out},ij}$ are the Lagrange multipliers for its interconnecting constraints $w_{\text{out},ji} = w_{\text{in},ij}$. The $\lambda^{(s)}_{\text{out},ij}$ variables are received by agent i through communication with agent j, that computed these variables for its interconnecting constraints with respect to agent i. The general multi-agent MPC scheme that results from this comprises at control cycle k the following:

(i) For $i = 1,\ldots,n$, agent i makes a measurement of the current state of the subnetwork $x_{i,0}$ and estimates expected disturbances $\tilde{d}_{i,0}$.

(ii) The agents cooperatively solve their control problems in the following iterative way:

(a) Set the iteration counter s to 1 and initialize the Lagrange multipliers $\lambda^{(s)}_{\text{in},ji},\lambda^{(s)}_{\text{out},ij}$ arbitrarily.

(b) Either serially or in parallel, for $i = 1,\ldots,n$, agent i determines $\tilde{x}^{(s+1)}_{i,k+1},\tilde{u}^{(s+1)}_{i,k},w^{(s+1)}_{\text{in},ji},w^{(s+1)}_{\text{out},ij}$, for $j\in\mathcal{N}_i$, by solving:

$$\min_{\substack{\tilde{x}_{i,k+1},\tilde{u}_{i,k},\tilde{y}_{i,k},\\ w_{\text{in},j_{i,1}i},\ldots,w_{\text{in},j_{i,m_i}i},\\ w_{\text{out},j_{i,1}i},\ldots,w_{\text{out},j_{i,m_i}i}}} J_{\text{local},i}\left(\tilde{x}_{i,k+1},\tilde{u}_{i,k},\tilde{y}_{i,k}\right) + \sum_{j\in\mathcal{N}_i} J_{\text{inter},i}\left(w_{\text{in},ji},w_{\text{out},ji},\lambda^{(s)}_{\text{in},ji},\lambda^{(s)}_{\text{out},ij}\right), \tag{20}$$

subject to the local dynamics (3)–(4) of subnetwork i over the horizon and the initial constraints (5)–(6) of subnetwork i.

(c) Update the Lagrange multipliers,

$$\lambda^{(s+1)}_{\text{in},ji} = \lambda^{(s)}_{\text{in},ji} + c(w^{(s+1)}_{\text{in},ji} - w^{(s+1)}_{\text{out},ij}). \tag{21}$$

(d) Move on to the next iteration $s+1$ and repeat steps ii.(a)–ii.(c). The iterations stop when the following stopping condition is satisfied:

$$\left\| \begin{bmatrix} \lambda^{(s+1)}_{\text{in},j_{1,1}1} - \lambda^{(s)}_{\text{in},j_{1,1}1} \\ \vdots \\ \lambda^{(s+1)}_{\text{in},j_{n,m_n}n} - \lambda^{(s)}_{\text{in},j_{n,m_n}n} \end{bmatrix} \right\|_\infty \leq \varepsilon,$$

where ε is a small positive scalar and $\|\cdot\|_\infty$ denotes the infinity norm. Note that satisfaction of this stopping condition can be determined in a distributed way, because each individual component of the infinity norm depends only on variables of one particular agent Negenborn et al. (2007).

(iii) The agents implement the actions until the beginning of the next control cycle.

(iv) The next control cycle is started.

Remark 3.1 *The Lagrange multipliers can be initialized arbitrarily, however, initializing them with values close to the optimal Lagrange multipliers will increase the convergence of the decision making process. Therefore, also initializing the Lagrange multipliers with values obtained from the previous decision-making step is beneficial, since typically these Lagrange multipliers will be good initial guesses for the new solution. We refer to this as a* warm start.

The schemes proposed in the literature implement step ii.(b) in a parallel fashion, e.g., (Camponogara et al., 2002; El Fawal et al., 1998; Georges, 1999). In the following we first discuss a scheme that implements step ii.(b) in a parallel fashion and then propose a novel scheme that implements it in a serial fashion. We then assess the performance of both schemes experimentally.

4. Serial versus parallel schemes

4.1. *Parallel implementation*

The parallel implementation is the result of using the *auxiliary problem principle* (Batut and Renaud, 1992; Kim and Baldick, 1997; Royo, 2001) of approximating the non-separable quadratic term in the augmented Lagrangian formulation of the combined overall control problem. The parallel scheme involves a number of parallel iterations in which all agents perform their local computing step at the same time.

Given the previous information $w_{\text{prev},ij} = w^{(s)}_{ij}$, and $w_{\text{prev},ji} = w^{(s)}_{ji}$ of the agents $j \in \mathcal{N}_i$ *of the last iteration* $s-1$, agent i solves problem (20) using the following additional objective function term for the interconnecting constraints:

$$J_{\text{inter},i}\left(w_{\text{in},ji}, w_{\text{out},ji}, \lambda^{(s)}_{\text{in},ji}, \lambda^{(s)}_{\text{out},ij}\right)$$
$$= \begin{bmatrix} \lambda^{(s)}_{\text{in},ji} \\ -\lambda^{(s)}_{\text{out},ij} \end{bmatrix}^T \begin{bmatrix} w_{\text{in},ji} \\ w_{\text{out},ji} \end{bmatrix} + \frac{c}{2} \left\| \begin{bmatrix} w_{\text{in,prev},ij} - w_{\text{out},ji} \\ w_{\text{out,prev},ij} - w_{\text{in},ji} \end{bmatrix} \right\|_2^2$$
$$+ \frac{b-c}{2} \left\| \begin{bmatrix} w_{\text{in},ji} - w_{\text{in,prev},ji} \\ w_{\text{out},ji} - w_{\text{out,prev},ji} \end{bmatrix} \right\|_2^2 .$$

This scheme uses only information computed during the last iteration $s-1$. The parallel implementation of step ii.(b) of the general multi-agent MPC scheme therefore consists of the following steps at decision step k, iteration s:

(ii) (b) For all agents $i \in \{1,\ldots,n\}$, *at the same time*, agent i solves the problem (20) to determine $\tilde{x}^{(s+1)}_{i,k+1}, \tilde{u}^{(s+1)}_{i,k}$, $w^{(s+1)}_{\text{in},ji}$, $w^{(s+1)}_{\text{out},ji}$, and sends to agent $j \in \mathcal{N}_i$ the computed values $w^{(s+1)}_{\text{out},ji}$.

The positive scalar c penalizes the deviation from the interconnecting variable iterates that were computed during the last iteration. This makes that the interconnecting variables that agent i computes at the current iteration will stay close to the interconnecting variables that neighboring agent $j \in \mathcal{N}_i$ computed earlier when c is chosen larger. With increasing c, it becomes more expensive for an agent to deviate from the interconnecting-variable values computed by the other agents. This results in a faster convergence of the interconnecting variables to values that satisfy the interconnecting constraints. However, it may still take some iterations to obtain optimal values for the local variables. On the one hand a higher c results in a higher number of iterations before reaching optimality, although the interconnecting constraints will be satisfied quickly. On the other hand, when c is smaller a large number of iterations will be necessary before reaching optimality, and the interconnecting constraints will not be satisfied quickly.

As additional parameter this scheme uses a positive scalar b. If $b > c$, then the term penalizes the deviation between the interconnecting variables of the current iteration and the interconnecting variables of the last iteration of agent i; it thus gives the agent less incentive to change its interconnecting variables from one iteration to the next. When $b \geq 2c$, and moreover the overall combined problem is convex, it can be proved that the iterations converge toward the overall minimum for sufficiently small ε (Bertsekas and Tsitsiklis, 1997; Kim and Baldick, 1997).

4.2. *Serial implementation*

The novel serial implementation that we propose is the result of using a *block coordinate descent* (Bertsekas and Tsitsiklis, 1997; Royo, 2001) for dealing with the non-separable quadratic term in the augmented Lagrange formulation of the combined overall control problem (17). The approach minimizes the quadratic term directly, in a serial way. Contrarily to the parallel implementation, in the serial implementation one agent after another minimizes its local and interconnecting variables while the other variables stay fixed.

Given the information $w_{\text{in, prev},ij} = w^{(s+1)}_{\text{in},ij}, w_{\text{out, prev},ij} = w^{(s+1)}_{\text{out},ij}$ computed at the current iteration s for each agent $j \in \mathcal{N}_i$ that has solved its problem *before* agent i in the *current* iteration s, and given the previous information $w_{\text{prev},ij} = w^{(s)}_{ij}$ of the *last* iteration $s-1$ for the other agents, agent i solves problem (19) using the following additional objective function:

$$J_{\text{inter},i}\left(w_{\text{in},ji}, w_{\text{out},ji}, \lambda_{\text{in},ji}^{(s)}, \lambda_{\text{out},ij}^{(s)}\right) = \begin{bmatrix} \lambda_{\text{in},ji}^{(s)} \\ -\lambda_{\text{out},ij}^{(s)} \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} w_{\text{in},ji} \\ w_{\text{out},ji} \end{bmatrix} + \frac{c}{2} \left\| \begin{bmatrix} w_{\text{in, prev},ij} - w_{\text{out},ji} \\ w_{\text{out, prev},ij} - w_{\text{in},ji} \end{bmatrix} \right\|_2^2 .$$

Thus, contrarily to the parallel implementation, the serial implementation uses both information from the current iteration and from the last iteration. The serial implementation implements step ii.(b) of the general scheme as follows at decision step k, iteration s:

(ii) (b) For $i = 1, \ldots, n$, *one agent after another*, agent i determines $\tilde{x}_{i,k+1}^{(s+1)}, \tilde{u}_{i,k}^{(s+1)}$, $w_{\text{in},ji}^{(s+1)}$, $w_{\text{out},ji}^{(s+1)}$ by solving (20), and sends to each agent $j \in \mathcal{N}_i$ the computed values $w_{\text{out},ji}^{(s+1)}$.

The role of the scalar c is similar as for the parallel implementation, except for that c now penalizes the deviation from the interconnecting variable iterates that were computed by the agents before agent i in the current iteration and by the other agents during the last iteration. Note that when for the parallel scheme $b = c$ the additional objective functions are the same, except for the previous information used: the parallel implementation uses only information from the last iteration, the serial also from the current.

5. Experiments

In this section we perform simulation experiments on a particular type of transportation network, viz., a power network, to compare and assess the performance of the schemes of Section 4. A power network consists of all generating units, substations, and interconnecting power lines whose purpose is to provide the necessary energy to consumers. The frequency is one of the main variables characterizing the system. The purpose of load-frequency control is to keep power generation equal to power consumption under consumption disturbances, such that the frequency is maintained close to a nominal frequency of typically 50 or 60Hz (Kundur, 1994). In a distributed setting, agents have to obtain agreement on power flowing over lines between subnetworks in order to be able to perform adequate local frequency control.

A large number of control strategies has been developed for load frequency control (Ibraheem et al., 2005). In the 70s, load-frequency control started being developed with control strategies based on centralized, non-MPC, control (see (Quazza, 1966; Elgerd and Fosha, 1970; Fosha and Elgerd, 1970)). From the 80s on also, distributed, non-MPC, schemes appeared (Kawabata and Kido, 1982; Park and Lee, 1984; Aldeen and Marsh, 1990; Yang et al., 1998, 2002). Recently, also MPC-based schemes have been proposed. A centralized MPC scheme for load-frequency control was proposed in (Rerkpreedapong et al., 2003). A decentralized MPC scheme for load-frequency control was proposed in (Atic et al., 2003). The latter approach is a decentralized approach, that does not take the interconnections between subnetworks explicitly into account. In (Camponogara et al., 2002) a distributed MPC scheme is proposed for load-frequency control assuming that only once per control step information between agents can be exchanged. Also in (Venkat et al., 2006) a distributed MPC scheme is applied to a load-frequency control example. The scheme uses distributed state estimation to provide nominal stability and performance properties. We consider distributed MPC using the parallel and serial scheme.

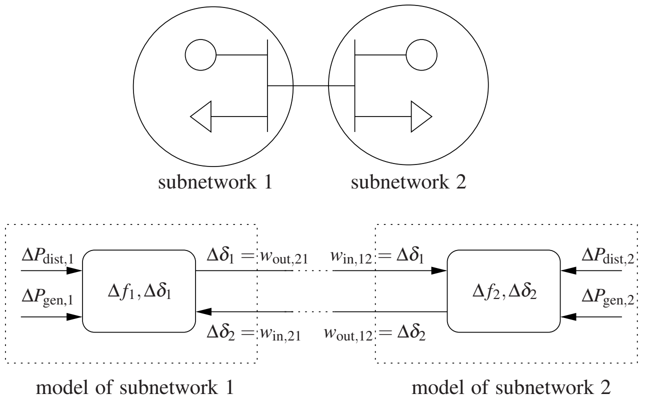
subnetwork 1
subnetwork 2
$\Delta P_{\text{dist},1}$
$\Delta P_{\text{gen},1}$
$\Delta f_1, \Delta\delta_1$
$\Delta\delta_1 = w_{\text{out},21}$
$\Delta\delta_2 = w_{\text{in},21}$
$w_{\text{in},12} = \Delta\delta_1$
$w_{\text{out},12} = \Delta\delta_2$
$\Delta f_2, \Delta\delta_2$
$\Delta P_{\text{dist},2}$
$\Delta P_{\text{gen},2}$
model of subnetwork 1
model of subnetwork 2

Fig. 2. Illustration of the physical network and the variables of the subnetwork models. In the top illustration a circle represents power generation and a triangle power consumption.

In a power network, each subnetwork has power generation capabilities and power consumption, see Fig. 2. Each control agent has to keep the frequency deviation within its subnetwork close to zero under minimal control input, accessing only local variables. For political and/or security reasons the agents only know the topology of their own subnetwork. Furthermore, each control agent can only sens the power consumption and change the power generation in its own subnetwork. Therefore this is a typical situation in which multi-agent control has to be employed.

5.1. *Control setup*

5.1.1. *Dynamical subnetwork models*

The continuous-time dynamics of subnetwork i are described by the following second-order system (Camponogara et al., 2002):

$$\frac{d}{dt}\Delta\delta_i(t) = 2\pi\Delta f_i(t)$$

$$\frac{d}{dt}\Delta f_i(t) = -\frac{1}{T_{P_i}}\Delta f_i(t) + \frac{K_{P_i}}{T_{P_i}}\Big(\Delta P_{\text{gen},i}(t) - \Delta P_{\text{dist},i}(t) + \sum_{j\in\mathcal{N}_i} \frac{K_{S_{ij}}}{2\pi}\left(\Delta\delta_j(t) - \Delta\delta_i(t)\right)\Big)$$

$$y_i(t) = \begin{bmatrix} \Delta\delta_i(t) \\ \Delta f_i(t) \end{bmatrix},$$

where at time t, for subnetwork $i \in \{1, \ldots, n\}$, $\Delta\delta_i$ is the angle deviation, Δf_i is the frequency deviation, $\Delta P_{\text{gen},i}$ is the change in power generation, $\Delta P_{\text{dist},i}$ is a disturbance in the load, y_i is the measurement of the state, and $K_{P_i}, T_{P_i}, K_{S_{ij}}$ are constants.

The values for the parameters are $K_{P_i} = 120$, $K_{S_{ij}} = K_{S_{ji}} = 0.5$, $T_{P_i} = 20$, for $i = 1, \dots, n$, $j \in \mathcal{N}_i$. Because we assume that the outputs y_i measure the full state noise-free, we will without loss of generality leave out the outputs y_i and only focus on the states in the following.

Defining the local control input $u_{i,k} = [\Delta P_{\mathrm{g},i,k}]$, local disturbances $d_{i,k} = [\Delta P_{\mathrm{d},i,k}]$, local state $x_{i,k} = [\Delta\delta_{i,k},\ \Delta f_{i,k}]^{\mathrm{T}}$, remaining variables $v_{i,k} = [\Delta\delta_{j_{i,1},k}, \dots, \Delta\delta_{j_{i,m_i},k}]^{\mathrm{T}}$, and discretizing the continuous-time model using an Euler approximation (with a step size of $\tau = 0.25\mathrm{s}$), the model can be written as:

$$x_{i,k+1} = A_i x_{i,k} + B_{1,i} u_{i,k} + B_{2,i} d_{i,k} + B_{3,i} v_{i,k}$$

where

$$A_i = \begin{bmatrix} 1 & \tau 2\pi \\ \tau \dfrac{-K_{P_i} K_{S_{ij}}}{2\pi T_{P_i}} & 1 - \tau \dfrac{1}{T_{P_i}} \end{bmatrix} \qquad B_{1,i} = \begin{bmatrix} 0 \\ \tau \dfrac{K_{P_i}}{T_{P_i}} \end{bmatrix}$$

$$B_{2,i} = \begin{bmatrix} 0 \\ -\tau \dfrac{K_{P_i}}{T_{P_i}} \end{bmatrix} \qquad B_{3,i} = \begin{bmatrix} 0 & \dots & 0 \\ \tau \dfrac{K_{P_i} K_{S_{ij_{i,1}}}}{2\pi T_{P_i}} & \dots & \tau \dfrac{K_{P_i} K_{S_{ij_{i,m_i}}}}{2\pi T_{P_i}} \end{bmatrix}.$$

5.1.2. *Interconnecting variables*

The interconnecting inputs for agent i are defined as in (8), and the interconnecting outputs for agent i are defined as in (9), with

$$E_i = \begin{bmatrix} 1 \; 0 & & 0 & \\ \vdots \; \vdots & & \vdots & \\ 1 \; 0 & & 0 & \\ & \ddots & & \ddots \\ & & 1 \; 0 & & 0 \\ & & \vdots \; \vdots & & \vdots \\ & & 1 \; 0 & & 0 \end{bmatrix}.$$

5.1.3. *Local control goals*

Since agent i has to minimize the frequency deviation and control input changes in its subnetwork, it uses the following quadratic local objective function:

$$J_{\mathrm{local},i}\left(\tilde{x}_{i,k+1}, \tilde{u}_{i,k}\right) = \sum_{l=0}^{N-1} \begin{bmatrix} x_{i,k+1+l} \\ u_{i,k+l} \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} Q_{i,x} & 0 \\ 0 & Q_{i,u} \end{bmatrix} \begin{bmatrix} x_{i,k+1+l} \\ u_{i,k+l} \end{bmatrix}$$

where

$$Q_{i,x} = \begin{bmatrix} 0 & 0 \\ 0 & 100 \end{bmatrix}, \quad Q_{i,u} = \begin{bmatrix} 50 \end{bmatrix}$$

A quadratic function has the advantage that larger deviations are penalized more, and moreover that the objective function is convex.

Remark 5.1 *Note that the defined subnetwork models, interconnecting variables, and local control goals lead to an overall combined control problem* (14) *that is convex.*

5.2. *Simulations*

5.2.1. *Scenario*

We consider a network divided into two subnetworks, each controlled by a control agent, see Fig. 2. We simulate the network in Matlab 7.1 and solve the optimization problems of the controllers using the CPLEX v10 Barrier QP solver, through the Tomlab interface to Matlab. The network is simulated in discrete time steps of 0.25s, for $k_{\mathrm{f}} = 20$ steps, thus yielding a total simulation time of to 5s. The subnetworks are initially in steady state, until a consumption disturbance of $\Delta P_{\mathrm{dist},2} = 1$ per unit (p.u.) occurs in subnetwork 2 after 0.5 seconds. At that time the dynamics of the subnetworks become highly dependent on each other, and the agents cannot make adequate predictions on the evolution of their own subnetworks unless they obtain agreement on the values of their interconnecting variables. In the following we first consider the uncontrolled situation, and then compare three controlled situations: 1) a hypothetical centralized agent uses the overall combined control problem to determine its actions for all subnetworks; 2) the agents of the subnetworks use the serial multi-agent MPC scheme; 3) the agents of the subnetworks use the parallel multi-agent MPC scheme. We first consider the performance of the resulting control over all control cycles in the full simulation span of 5s for a particular setting of the parameters, and then focus on a particular control cycle to consider the iterations within that control cycle and gain more insight into how the parameters influence the performance of the multi-agent controllers.

5.2.2. *Full simulation evaluation criterion*

To compare and assess the performance of the overall combined, the serial, and the parallel scheme over the full simulation period, costs are computed over the full simulation time span, i.e.,

$$J_{\mathrm{simulation}}(\cdot) = \sum_{i=1}^{n} \sum_{l=0}^{k_{\mathrm{f}}-1} J_{\mathrm{stage},i}(\bar{x}_{i,1+l}, \bar{u}_{i,l}, \bar{y}_{i,l}),$$

where the bar indicates that the value of the variable is the actual and not predicted value, e.g., $\bar{x}_{i,k}$ refers to the actual state of subnetwork i at time k, and not the state predicted by an agent.

5.2.3. *Uncontrolled simulation*

Fig. 3 shows the evolution of the frequency deviation in both subnetworks when no actions are taken, and Fig. 4 shows the resulting power exported from subnetwork 1 to subnetwork 2. Due to the increase in power consumption in subnetwork 2, the frequency in subnetwork 2 decreases, since the generation capacity of subnetwork 2 cannot directly provide the required new amount of power. Subnetwork 1 responds by automatically exporting some power to subnetwork 2, making that in subnetwork 1 a shortage of power appears, causing a drop in the frequency of subnetwork 1. This again triggers subnetwork 2 to export some power to subnetwork 1, but as can be seen in the figure, the natural power flows over the interconnecting line destabilize the frequency in both subnetworks. The performance over the full simulation period is $J_{\mathrm{simulation}}(\cdot) = 9042$.

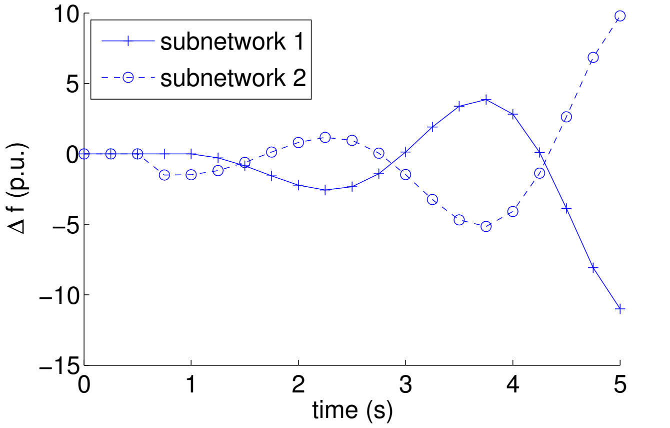

Fig. 3. Uncontrolled simulation of frequency deviation after a disturbance in subnetwork 2.

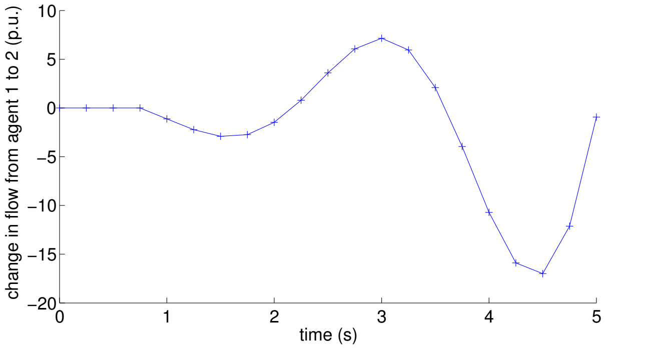

Fig. 4. Resulting power flowing from subnetwork 1 to subnetwork 2 for the uncontrolled simulation.

5.2.4. *Controlled simulation*

Now consider the situation that appears when every 0.25s new actions are determined by an overall MPC scheme based on 1) a hypothetical centralized agent that uses the combined overall control problem defined in (14), or 2) agents that use the serial scheme with warm start, or 3) agents that use the parallel scheme with warm start. For now we choose as parameters a prediction horizon of $N = 5$ (corresponding to a horizon of 1.25s), $c = 1$, $\varepsilon = 1e^{-4}$, $b = 2c$ (which for overall convex problems guarantees convergence). In Section 5.3 we discuss the influence of different values for the parameters on the performance.

Fig. 5 shows the controlled evolution of the frequency deviations, Fig. 6 shows the resulting power exported from subnetwork 1, and Fig. 7 shows the inputs that have been implemented, obtained using each of the three control approaches. We mentioned before, that the overall combined control problem is convex, and therefore good performance of the multi-agent schemes is expected. Indeed, for the chosen parameters, the difference between the performance of overall combined control problem and the two distributed schemes is negligible; the performance over the full simulation is $J_{\text{simulation}}(\cdot) = 198$ for each of the schemes, which is clearly an improvement over the uncontrolled situation. Furthermore, each of the controllers takes actions that in the end bring back the frequency deviations and changes in power generation to zero and in this way the agents stabilize the system. The agents have in a distributed way obtained the performance of a centralized controller.

The number of iterations performed by the serial and parallel scheme is shown in Fig. 8. Initially, when the disturbance has not appeared yet, the agents require few iterations in their

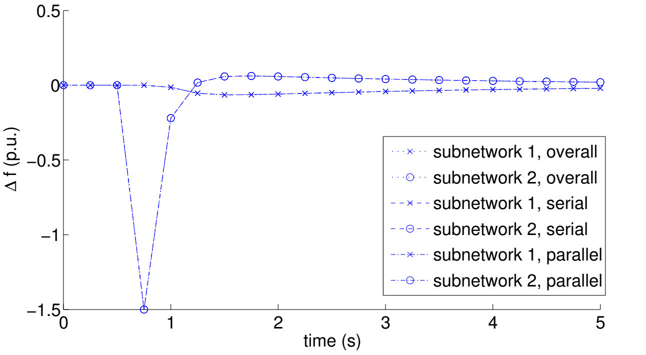

Fig. 5. Controlled simulation of frequency deviation using the overall combined scheme, the serial scheme, and the parallel scheme. Note the significantly smaller range of Δf, compared with the range in the uncontrolled evolution in Fig. 3.

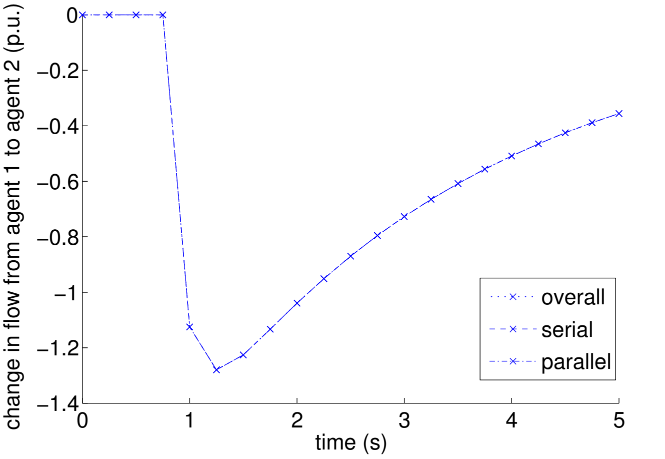

Fig. 6. Resulting power flowing from subnetwork 1 to subnetwork 2 for the controlled simulations. Note the significantly smaller range of the change in the power flow, compared with the range in the uncontrolled evolution in Fig. 4.

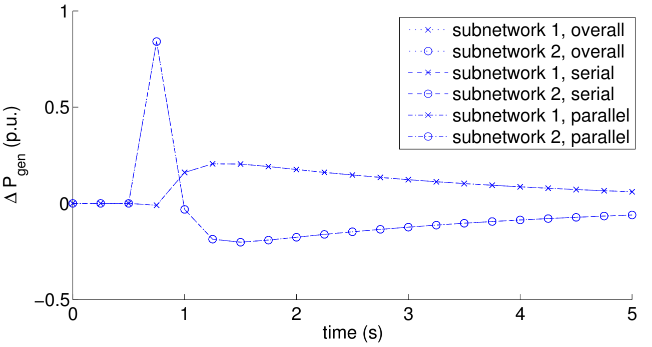

Fig. 7. Controlled evolution of inputs computed by overall combined scheme, the serial scheme, and the parallel scheme.

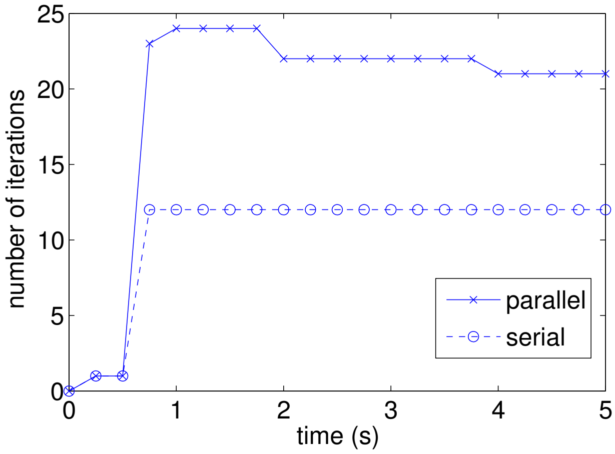

Fig. 8. Number of iterations required per control cycle of the serial and parallel scheme.

control cycles making. After the disturbance at 0.5s has appeared the agents require significantly more iterations, reflected by the increasing number of iterations at time 0.75s. We notice that the serial scheme requires fewer iterations than the parallel scheme, explained by the fact that the serial schemes uses information from both the previous and the current iteration.

5.3. *A single control cycle*

To gain more insight in the role of the parameters and in the iterations that the serial and the parallel scheme perform at a single control cycle we now focus on the iterations of a single, representative, control cycle among the agents. We consider the iterations of the serial and parallel scheme right after a disturbance has taken place. Consider the situation in which the state of subnetwork 1 is $x_{1,0} = [0,0]^{\mathrm{T}}$ and the state of subnetwork 2 is $x_{2,0} = [0,0.5]^{\mathrm{T}}$.

5.3.1. *Control cycle evaluation criterion*

To evaluate the solution over the prediction horizon determined by the different schemes at a single control cycle, the inputs coming from the different schemes are implemented to determine the resulting state trajectory, after which the cycle performance is as

$$J_{\text{cycle}}(\cdot) = \sum_{i=1}^{n} \sum_{l=0}^{N-1} J_{\text{stage},i}(\bar{x}_{i,1+l}, u_{i,l}, \bar{y}_{i,l}).$$

5.3.2. *Varying c and prediction horizon N*

We vary the parameters N and c, while keeping the stopping tolerance $\varepsilon = 1e^{-4}$, and $b = 2c$. For values of $c \in \{1, 10, 100\}$, the number of iterations required by the parallel scheme and the number of iterations required by the serial scheme is shown in Fig. 9. For a given value of c, the serial scheme requires fewer iterations than the parallel scheme for all except a small interval of prediction horizon lengths. For values of c close to zero, the influence of the additional objective function $J_{\text{inter},i}$ of both the parallel and the serial scheme vanishes, making that the difference between the two schemes vanishes as well.

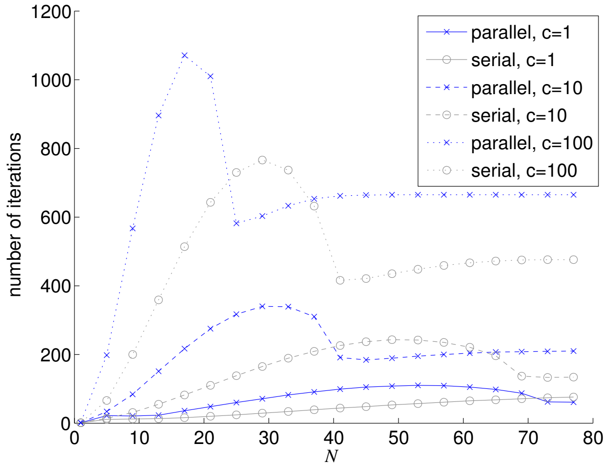

Fig. 9. For varying N and varying c, the number of iterations that the parallel and the serial scheme require before stopping.

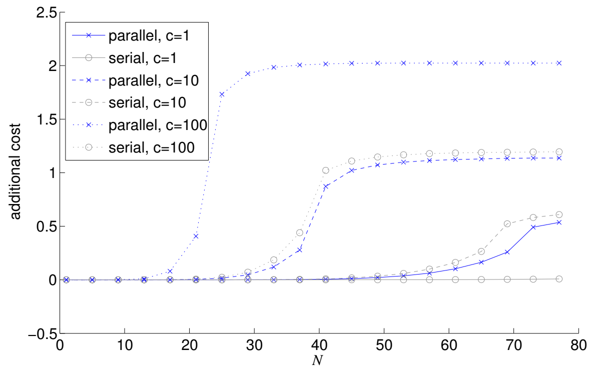

Fig. 10. For varying N and varying c, the additional cost of the parallel and the serial scheme compared to the overall optimal costs.

When increasing the prediction horizon N, it is expected that the number of iterations required increases as well, since with a longer horizon the number of interconnecting variables increases. We see in Fig. 9 that the number of iterations does increase with an increasing prediction horizon length, although only up to a certain prediction horizon length. Interestingly, from a certain prediction horizon length the number of iterations decreases again, when compared to a smaller prediction horizon. This behavior is due to the inputs of the subnetworks over the first prediction steps being relatively more important for obtaining low costs, than the inputs at later prediction steps. Therefore, obtaining satisfying interconnecting constraints for the earlier prediction horizon steps involves more iterations. From a certain prediction horizon length, the information that the agents obtain from the communicated interconnecting inputs and outputs for later prediction horizon steps restricts the values for the interconnecting variables of earlier prediction horizon steps, thus resulting in faster convergence.

Fig. 10 shows the additional cost imposed by using the serial or the parallel scheme instead of the overall control scheme. For

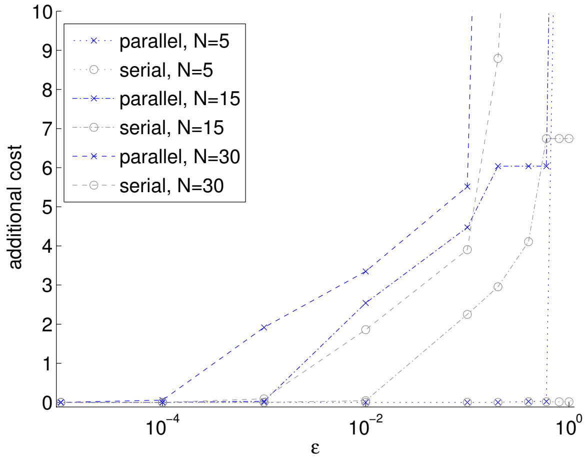

Fig. 11. For varying ε and N, the number of iterations that the parallel and the serial scheme require before stopping.

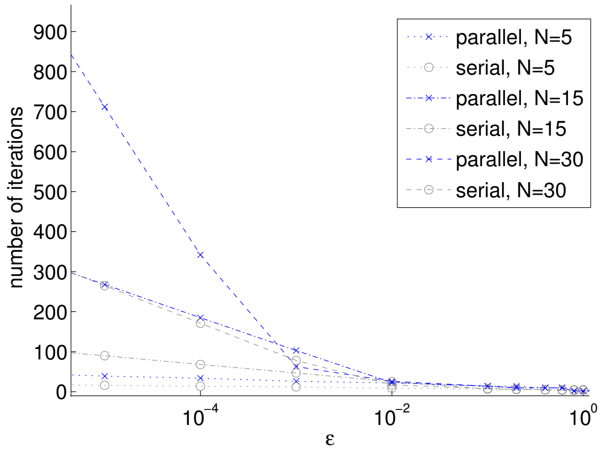

Fig. 12. For varying ε and N, the additional cost of the parallel and the serial scheme compared to the combined overall cost.

smaller prediction horizons, the serial and the parallel scheme perform comparable to the overall scheme. For larger prediction horizons the performance of the parallel scheme deteriorates faster than the serial scheme.

5.3.3. *Varying the stopping tolerance ε*

With increasing stopping tolerance ε the stopping condition will be satisfied within fewer iterations, at the price of a worse solution. Indeed, this characteristic behavior is shown in Fig. 11 and Fig. 12. Fig. 11 shows for $c = 10$, $b = 2c$, varying N, and varying ε, that with ε increasing fewer iterations are required, while Fig. 12 shows that the additional cost of the solution increases when compared to the overall combined scheme. The cost of the serial scheme shows slower deviation from the cost of the overall combined scheme than the cost of the parallel scheme.

6. Conclusions and future research

In this paper we have considered multi-agent model predictive control for the control of large-scale transportation networks, like road traffic networks, power networks, sewer networks, etc. In particular, we have proposed a novel serial scheme for agents to deal with the interconnections between subnetworks. We compared this with an existing parallel scheme and an centralized overall scheme. For the serial and the parallel schemes, the performance of the solution obtained converges toward the performance of the solution obtained by the overall control problem, provided that the overall control problem is convex. We have discussed the schemes theoretically and assessed their performance experimentally by means of simulation studies on a power network.

Although the parallel scheme is more frequently used throughout the literature, for the networks we have considered the proposed serial scheme shows to have preferable properties in terms of solution speed, by requiring fewer iterations, and solution quality, by providing performance closer to the centralized overall control problem.

Future research consists of deriving analytical bounds on the rate of convergence and assessing the performance of the serial and parallel approach for networks with a larger size and different topology. Furthermore, the methods will be extended to situations in which the problem of controlling the transportation network cannot be formulated as a convex problem. In particular we will extend the methods to deal with networks modeled as hybrid systems in which both continuous and discrete dynamics appear, a situation typically appearing when, e.g., continuous flows together with discrete actions are present.

Acknowledgments

This research was supported by the European 6th Framework Network of Excellence "HYbrid CONtrol: Taming Heterogeneity and Complexity of Networked Embedded Systems (HYCON)", contract number FP6-IST-511368, the project "Multi-agent control of large-scale hybrid systems" (DWV.6188) of the Dutch Technology Foundation STW, and the BSIK project "Next Generation Infrastructures (NGI)".

Appendix A. List of most frequent notations

n	number of subnetworks
i	index of an agent or subnetwork
$\mathcal{N}_i$	set of indexes of neighboring agents of agent i
j	index of a neighboring agent, i.e., $j \in \mathcal{N}_i$
$j_{i,q}$	index of qth neighbor of i
m_i	number of neighbors of i
k	control cycle step
$x_{i,k}$	local state of i at step k
$u_{i,k}$	local input of i at step k
$d_{i,k}$	local disturbance of i at step k
$v_{i,k}$	remaining variable of i at step k
$y_{i,k}$	local output variable of i at step k
$A_i, B_{1,i}, B_{2,i}, B_{3,i}$	matrices to describe linear time-invariant state equations
$C_i, D_{1,i}, D_{2,i}, D_{3,i}$	matrices to describe linear time-invariant output equations
N	prediction horizon length
l	sample step within prediction period
$\tilde{a}_k$	$= [\ a_k^{\mathrm{T}}, \ldots, a_{k+N-1}^{\mathrm{T}}\ ]^{\mathrm{T}}$
$\tilde{a}_{i,k}$	$= [\ a_{i,k}^{\mathrm{T}}, \ldots, a_{i,k+N-1}^{\mathrm{T}}\ ]^{\mathrm{T}}$

$w_{\mathrm{in},ji}$	interconnecting input of i with respect to j
$w_{\mathrm{out},ji}$	interconnecting output of i with respect to j
$J_{\mathrm{local},i}(\cdot)$	local objective function
$J_{\mathrm{stage},i}(\cdot)$	local cost per prediction step
Q	matrix for quadratic costs
f	vector for linear costs
$w_{\mathrm{in},i}$	$= [\ w^{\mathrm{T}}_{\mathrm{in},j_{i,1}i},\ldots,w^{\mathrm{T}}_{\mathrm{in},j_{i,m_i}i}\ ]^{\mathrm{T}}$
$w_{\mathrm{out},i}$	$= [\ w^{\mathrm{T}}_{\mathrm{out},j_{i,1}i},\ldots,w^{\mathrm{T}}_{\mathrm{out},j_{i,m_i}i}\ ]^{\mathrm{T}}$
$J_{\mathrm{inter},i}(\cdot)$	objective term to deal with interconnecting constraints
E_i	interconnecting output selection matrix
$(\cdot)^{\mathrm{T}}$	transpose operator
$\lambda_{\mathrm{in},ji}$	Lagrange multiplier associated with interconnecting constraint $w_{\mathrm{in},ji} = w_{\mathrm{out},ij}$
$\lambda_{\mathrm{out},ij}$	Lagrange multiplier associated with interconnecting constraint $w_{\mathrm{in},ji} = w_{\mathrm{out},ij}$
$\|\cdot\|_2$	two norm
$\|\cdot\|_\infty$	infinity norm
$L(\cdot)$	augmented Lagrange function
ε	small positive constant
s	iteration counter
c	positive constant
b	positive scalar
k_{f}	simulation finishing step

References

Acar, L., 1992. Some examples for the decentralized receding horizon control. In: Proceedings of the 31st IEEE Conference on Decision and Control. Tucson, Arizona, pp. 1356–1359.

Aicardi, M., Casalino, G., Minciardi, R., Zoppoli, R., Nov. 1992. On the existence of stationary optimal receding-horizon strategies for dynamic teams with common past information structures. IEEE Transactions on Automatic Control 37, 1767–1771.

Aldeen, M., Marsh, J. F., 1990. Observability, controllability and decentralized control of interconnected power systems. International Journal on Computers and Electrical Engineering 16 (4), 207–220.

Atic, N., Rerkpreedapong, D., Hasanovic, A., Feliachi, A., 2003. NERC compliant decentralized load frequency control design using model predictive control. In: Proceedings on the IEEE Power Engineering Society General Meeting.

Baglietto, M., Parisini, T., Zoppoli, R., 1999. Neural approximators and team theory for dynamic routing: A receding-horizon approach. In: Proceedings of the 38th IEEE Conference on Decision and Control. Phoenix, Arizona, pp. 3283–3288.

Batut, J., Renaud, A., Aug. 1992. Daily generation scheduling optimization with transmission constraints: a new class of algorithms. IEEE Transactions on Power Systems 7 (3), 982–989.

Beccuti, A. G., Morari, M., Jun. 2006. A distributed solution approach to centralized emergency voltage control. In: Proceedings of the 2006 IEEE American Control Conference. Digital Object Identifier 10.1109/ACC.2006.1657251.

Bertsekas, D. P., 1982. Constrained Optimization and Lagrange Multiplier Methods. Academic Press, London, UK.

Bertsekas, D. P., Tsitsiklis, J. N., 1997. Parallel and Distributed Computation: Numerical Methods. Athena Scientific, Nashua, New Hampshire.

Boyd, S., Vandenberghe, L., 2004. Convex Optimization. Cambridge University Press, Cambridge, United Kingdom.

Camacho, E. F., Bordons, C., 1995. Model Predictive Control in the Process Industry. Springer-Verlag, Berlin, Germany.

Camponogara, E., Jia, D., Krogh, B. H., Talukdar, S., Feb. 2002. Distributed model predictive control. IEEE Control Systems Magazine 1, 44–52.

De Schutter, B., van den Boom, T., Hegyi, A., 2002. A model predictive control approach for recovery from delays in railway systems. Transportation Research Record (1793), 15–20.

Dunbar, W. B., Murray, R. M., Apr. 2006. Distributed receding horizon control for multi-vehicle formation stabilization. Automatica 42.

El Fawal, H., Georges, D., Bornard, G., 1998. Optimal control of complex irrigation systems via decomposition-coordination and the use of augmented lagrangian. In: Proceedings of the 1998 International Conference on Systems, Man, and Cybernetics. San Diego, California, pp. 3874–3879.

Elgerd, O. I., Fosha, C., Feb. 1970. Optimum megawatt frequency control of multi-area electric energy systems. IEEE Transactions on Power Apparatus and Systems PAS-89 (4), 556–563.

Fosha, C. E., Elgerd, O. I., Apr. 1970. The megawatt frequency control problem: A new approach via optimal control theory. IEEE Transactions on Power Apparatus and Systems PAS-89 (4), 563–577.

Georges, G., 1999. Decentralized adaptive control for a water distribution system. In: Proceedings of the 3rd IEEE Conference on Control Applications. Glasgow, UK, pp. 1411–1416.

Geyer, T., Larsson, M., Morari, M., Sep. 2003. Hybrid emergency voltage control in power systems. In: Proceedings of the European Control Conference 2003. Cambridge, UK.

Gomez, M., Rodellar, J., Vea, F., Mantecon, J., Cardona, J., 1998. Decentralized predictive control of multireach canals. In: Proceedings of the 1998 IEEE International on Systems, Man, and Cybernetics. San Diego, California, pp. 3885–3890.

Hegyi, A., De Schutter, B., Hellendoorn, J., Mar. 2005. Optimal coordination of variable speed limits to suppress shock waves. IEEE Transactions on Intelligent Transportation Systems 6 (1), 102–112.

Hines, P., Huaiwei, L., Jia, D., Talukdar, S., Mar. 2005. Autonomous agents and cooperation for the control of cascading failures in electric grids. In: Proceedings of the 2005 IEEE International Conference on Networking, Sensing and Control. Tucson, Arizona, pp. 273–278.

Ibraheem, Kumar, P., Kothari, D. P., Feb. 2005. Recent philosophies of automatic generation control strategies in power systems. IEEE Transactions on Power Systems 20 (1), 346–357.

Jia, D., Krogh, B., May 2002. Min-max feedback model predictive control for distributed control with communication. In: Proceedings of the 2002 American Control Conference. Anchorage, Alaska, pp. 4507–4512.

Jia, D., Krogh, B. H., Jun. 2001. Distributed model predictive control. In: Proceedings of the 2001 American Control Conference. Arlington, Virginia, pp. 2767–2772.

Katebi, M. R., Johnson, M. A., 1997. Predictive control design for large-scale systems. Automatica 33 (3), 421–425.

Kawabata, H., Kido, M., Jul. 1982. A decentralized scheme of load frequency control power system. Electrical Engineering Japan 102 (4), 100–106.

Kim, B. H., Baldick, R., May 1997. Coarse-grained distributed optimal power flow. IEEE Transactions on Power Systems 12 (2), 932–939.

Kotsialos, A., Papamichail, I., Margonis, I., Papageorgiou, M., Aug. 2006. Hierarchical nonlinear model-predictive ramp metering control for freeway networks. In: Proceedings of the 11th IFAC Symposium on Control in Transportation Systems. Delft, The Netherlands, pp. 124–129.

Kundur, P., 1994. Power System Stability and Control. McGraw Hill, New York.

Maciejowski, J. M., 2002. Predictive Control with Constraints. Prentice Hall, Harlow, England.

Marinaki, M., Papageorgiou, M., Sep. 2001. Rolling-horizon optimal control of sewer networks. In: Proceedings of the 2001 IEEE International Conference on Control Applications. Mexico City, Mexico, pp. 594–599.

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

Morari, M., Lee, J. H., 1999. Model predictive control: past, present and future. Computers and Chemical Engineering 23, 667–682.

Negenborn, R., De Schutter, B., Hellendoorn, H., Apr. 2006. Multi-agent model predictive control of transportation networks. In: Proceedings of the 2006 IEEE International Conference on Networking, Sensing and Control (ICNSC 2006). Ft. Lauderdale, Florida, pp. 296–301.

Negenborn, R., De Schutter, B., Hellendoorn, J., Jul. 2007. Efficient implementation of serial multi-agent model predictive control by parallelization. Accepted for the 2007 IEEE International Conference on Networking, Sensing, and Control.

Park, Y. M., Lee, K. Y., Sep. 1984. Optimal decentralized load frequency control. Electrical Power Systems Research 7 (4), 279–288.

Qin, S., Badgewell, T., 1997. An overview of industrial model predictive control technology. Chemical Process Control - V 93 (316), 232–256.

Quazza, G., Jul. 1966. Noninteracting controls of interconnected electric power systems. IEEE Transactions on Power Apparatus and Systems PAS-85 (7), 727–741.

Rerkpreedapong, D., Atic, N., Feliachi, A., May 2003. Economy oriented model predictive load frequency control. In: Proceedings of the 2003 Large Engineering Systems Conference on Power Engineering. Canada, pp. 12–16.

Royo, C. B., May 2001. Generalized unit commitment by the radar multiplier method. Ph.D. thesis, Technical University of Catalonia, Barcelona, Spain.

Sawadogo, S., Faye, R. M., Malaterre, P. O., Mora-Camino, F., 1998. Decentralized predictive controller for delivery canals. In: Proceedings of the 1998 IEEE International on Systems, Man, and Cybernetics. San Diego, California, pp. 3380–3884.

Siljak, D. D., 1991. Decentralized Control of Complex Systems. Academic Press, Boston.

Sycara, K. P., 1998. Multiagent systems. AI Magazine 2 (19), 79–92.

Venkat, A. N., Hiskens, I. A., Rawlings, J. B., Wright, S. J., Dec. 2006. Distributed output feedback MPC for power system control. In: Proceedings of the 2006 IEEE Conference on Decision and Control. California, USA, paper ThIP6.9.

Wahlin, B. T., May 2004. Performance of model predictive control on ASCE test canal 1. Journal of Irrigation and Drainage Engineering 130 (3), 227–238.

Weiss, G., 2000. Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence. MIT Press, USA.

Yang, T. C., Cimen, H., Zhu, Q. M., Jan. 1998. Decentralised load-frequency controller design based on structured singular values. Proc. Inst. Elect. Eng. C 145 (1), 7–14.

Yang, T. C., Ding, Z. T., Yu, H., Mar. 2002. Decentralised power system load frequency control beyond the limit of diagonal dominance. International Journal on Electrical Power Energy Systems 24 (3), 173–184.