Multi-Agent Model Predictive Control

with Applications to Power Networks

R.R. Negenborn

Multi-Agent Model Predictive Control

with Applications to Power Networks

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof.dr.ir. J.T. Fokkema,
voorzitter van het College van Promoties,
in het openbaar te verdedigen op dinsdag 18 december 2007 om 10:00 uur
door Rudy Rafaël NEGENBORN,
doctorandus in de informatica,
geboren te Utrecht.

Dit proefschrift is goedgekeurd door de promotoren:
Prof.dr.ir. J. Hellendoorn
Prof.dr.ir. B. De Schutter

Samenstelling promotiecommissie:

Rector Magnificus	voorzitter
Prof.dr.ir. J. Hellendoorn	Technische Universiteit Delft, promotor
Prof.dr.ir. B. De Schutter	Technische Universiteit Delft, promotor
Prof.dr. G.J. Olsder	Technische Universiteit Delft
Prof.dr. J.-J.Ch. Meyer	Universiteit Utrecht
Prof.Dr. G. Andersson	ETH Zürich
Prof.Dr.-Ing. W. Marquardt	RWTH Aachen University
Ir. J.J.M. Langedijk	Siemens Nederland N.V.
Prof.dr. C. Witteveen	Technische Universiteit Delft (reservelid)

This thesis has been completed in partial fulfillment of the requirements of the Dutch Institute of Systems and Control (DISC) for graduate studies. The research described in this thesis was supported by the project "Multi-agent control of large-scale hybrid systems" (DWV.6188) of the Dutch Technology Foundation STW and by an NWO Van Gogh grant (VGP79-99).

TRAIL Thesis Series T2007/14, The Netherlands TRAIL Research School

Published and distributed by: R.R. Negenborn
E-mail: rudy@negenborn.net
WWW: http://www.negenborn.net/mampc/

ISBN 978-90-5584-093-9

Keywords: multi-agent control, model predictive control, power networks, transportation networks.

Printed in The Netherlands

Contents

Chapter 3

Networked hybrid systems

In Chapter 2 we have considered multi-agent control of transportation networks involving only continuous variables and dynamics. In this chapter we consider multi-agent control of hybrid systems, i.e., distributed control of systems with both continuous and discrete dynamics. In Section 3.1 we introduce hybrid systems, illustrate how transportation networks can be seen as hybrid systems, and discuss which issues have to be dealt with when developing multi-agent single-layer MPC approaches for such systems. In Section 3.2 we focus on formulating prediction models of hybrid systems and discuss how transformations can be used to recast descriptions of hybrid systems into systems of linear mixed-integer constraints. In Section 3.3 we then apply these transformations to construct a model of a particular hybrid system. In Section 3.4 we focus on multi-agent control of interconnected hybrid systems and propose an extension of the serial multi-agent single-layer MPC scheme of Chapter 2.

In this chapter we apply the discussed techniques to two applications. In Section 3.3 we consider a decentralized multi-agent single-layer MPC approach for optimization of energy consumption in households. In Section 3.5 we propose an extension of the serial multi-agent approach of Chapter 2 for load-frequency control with discrete generation switching.

Parts of this chapter have been published in [68, 108].

3.1 Transportation networks as hybrid systems

Many of the transportation networks of our interest can be seen as *hybrid systems*. Hybrid systems [4, 104, 143] arise when continuous dynamics are combined with discrete dynamics. The following examples show how particular transportation networks can be seen as hybrid systems:

- In power networks, the transients and the evolution of the voltage and power levels and the demands of generators and users yield continuous dynamics, whereas the activation or deactivation of generators, lines, or users corresponds to discrete dynamics.
- In road traffic networks the flow of the cars through the network can be modeled with continuous dynamics, and elements such as ramp metering, traffic signals, lane closures, route directions, etc., yield discrete dynamics on the system.

- In water networks the evolution of the water levels can be modeled with continuous dynamics, whereas opening and closing of dams, and activating or deactivating of pumps yield discrete dynamics.

More generally speaking, hybrid dynamics are the result of the discrete dynamics caused by, e.g., saturation effects, discrete switching of actuators, discrete controller logic, priorities on control, reaching of physical bounds, etc., in combination with the continuous dynamics of, e.g., flows, pressures, speeds, levels, etc.

Conventional control approaches usually either consider only continuous or only discrete dynamics. The control approaches that do consider discrete and continuous dynamics simultaneously are mostly based on a centralized control paradigm, since multi-agent control has mostly been approached either from a computer science point of view, which focuses on discrete dynamics, or from a control engineering point of view, which focuses on continuous dynamics. Structured control design methods for large-scale hybrid systems are therefore lacking.

In a multi-agent single-layer MPC control structure the network is divided into n subnetworks, each controlled by a single control agent, cf. Section 1.3.2. Each of the control agents uses MPC to determine which actions to take. Each agent hereby uses a prediction model to predict the evolution of its subnetwork under various actuator settings over a certain prediction horizon. For transportation networks that are hybrid systems, all or some of the subnetworks will be hybrid systems. Issues that we address in the following sections are related to:

- Formalizing the hybrid behavior into suitable mathematical models. The control agents have to use prediction models that on the one hand adequately represent the hybrid dynamics, while on the other hand give MPC problems that can be solved efficiently, e.g., by making it possible to use state-of-the-art commercially available optimization problem solvers.

- Making control agents choose local actions that give performance that is as close as possible to overall optimal network performance, when the subnetworks of the control agents are hybrid systems. When the subnetwork that a control agent controls is a hybrid system, the corresponding prediction model will typically contain both continuous and discrete variables. This has as consequence that the MPC optimization problem of a particular control agent will be nonconvex, and that therefore also the overall combined control problem defined in Section 2.4 will be nonconvex. Approaches as discussed in Section 2.4 for coordinating control agents may not give satisfactory performance, and a way has to be found to improve this.

We first focus on the first issue, i.e., modeling of hybrid systems, by discussing how transformations can be used to transform discrete logic into mixed-integer equality and inequality constraints. We then employ these transformations for designing a prediction model used by a decentralized multi-agent single-layer MPC control structure to control household energy consumption. Next, we consider the second issue, i.e., multi-agent control of interconnected hybrid systems, by extending the serial approach of Section 2.4 to deal with hybrid subnetworks. The approach is experimentally assessed on a load-frequency control problem in which generation can be changed in discrete quantities.

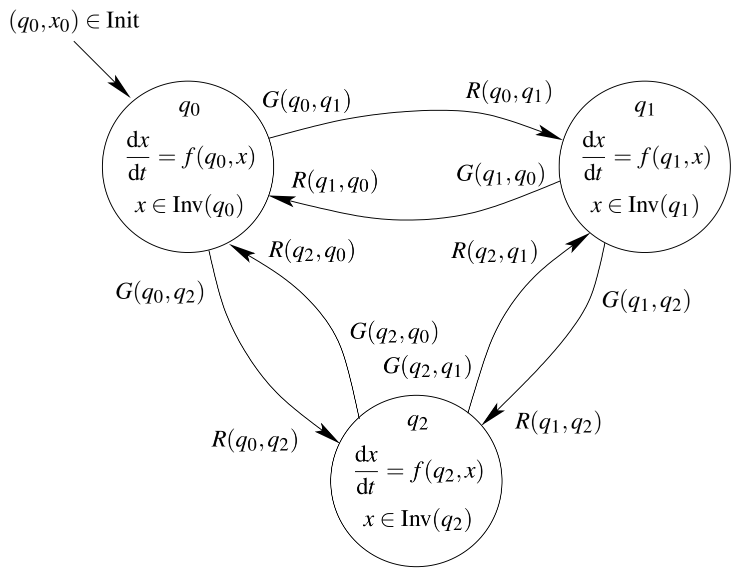


Figure 3.1: Schematic representation of a hybrid automaton.

3.2 Modeling of hybrid systems

There are many ways in which models of hybrid systems can be constructed. Typically the continuous dynamics are represented by systems of differential or difference equations and the discrete dynamics are represented by automata or finite state machines [29, 30]. Combining these two types of models results in hybrid automata [143], a type of model that can represent a large class of hybrid systems.

Figure 3.1 depicts an example of a such a hybrid automaton. Each node represents a *mode* of the system. The modes represent the discrete operating points, i.e., q_0, q_1, and q_2. In this case, each mode is governed by its own continuous dynamics, given by a system of differential equations, e.g., $\frac{dx}{dt} = f(q_1, x)$ for mode q_1. The system can stay in a particular mode as long as the continuous state stays inside the invariant set of that mode, e.g., $x \in \text{Inv}(q_1)$. The system can also transition to a different mode, and in fact has to transition to a new mode if the continuous state x is no longer inside the invariant set. The system can only transition from one mode to another, if the transition between these modes is enabled. The guard set $\mathcal{G}$ indicates for which states x the transition from one mode to another is enabled. The reset set $\mathcal{R}$ indicates which values the states can take on when a transition is made to a new mode. If each sequence of continuous state and mode transitions is uniquely determined only by the initial continuous state and mode, then the hybrid automaton is deterministic. Otherwise, the hybrid automaton is non-deterministic.

Hybrid automata have a large expressibility, in the sense that they can in principle represent the dynamics of any hybrid system. However, this expressibility comes at the price of increased difficulties for analytical studies, simulation, etc. By making assumptions on the possible mode transitions, the dynamics inside the modes, and the guard and the reset sets, different types of models can be defined. Each of these types of models will have different characteristics when it comes to the easiness of performing time-domain simulations, the possibility for analytical analysis, and the range of hybrid systems that can be represented. Some types of models that can be considered as special cases of hybrid automata are timed Petri-nets [34], mixed-logical dynamic models [16], piecewise-affine models [133], max-min-plus scaling models [35], etc. The equivalence of some of these types of models is shown in [57].

3.2.1 Models for MPC control

In the description of dynamics of hybrid systems discrete logic statements are commonly encountered, e.g., in the form of if-then or if-then-else rules. For a deterministic hybrid automaton an example of a discrete logic statements is "if $x \notin \mathrm{Inv}(q_1)$ and $x \in G(q_1,q_2)$, then $q = q_2$ and $x \in R(q_1,q_2)$". This statement means that if continuous state x is not in the invariant set of q_1 anymore and x is in the guard set G guarding the transition from q_1 to q_2, that then the transition to mode q_2 is made, and the continuous state obtains a value from the reset set associated with that transition. Discrete logic can be dealt with when formulating the prediction model of a control agent in the following ways:

- Software can be used that simulates the system, including the discrete logic. This software accepts a starting state and a series of inputs, and delivers an ending state and a series of outputs. Hence, the software is the prediction model of the system. The control agent can include this prediction model using nonlinear constraints in its MPC optimization problem. It can then use nonlinear optimization techniques to solve the nonlinear MPC optimization problem.
- The discrete logic can be transformed into linear equality and inequality constraints. The prediction model of the system will then consist of a system of linear equality and inequality constraints, in the case that the dynamics given fixed discrete dynamics are linear. The control agent can include this prediction model using mixed-integer linear constraints in its MPC optimization problem. It can then use mixed-integer linear or quadratic programming techniques to solve the MPC optimization problem.

In Chapter 4 we discuss the first approach. Below, we discuss the second approach, first from a more theoretical point of view in Section 3.2.2, then from a more applied point of view in Section 3.3.

3.2.2 From discrete logic to linear mixed-integer constraints

In [16, 149] it is shown how discrete logic statements can be transformed into linear mixed-integer equality and inequality constraints, i.e., constraints involving both variables that take on values from a continuous set of values, and variables that take on values from a discrete set of values. As in [16], we denote by $\mathbf{x} \in \mathbb{R}^n$ continuous variables and by $\delta \in \{0,1\}$ a binary logical variable. In addition, we denote by [exp] a logic statement, which has as value the evaluation of an expression exp to true or false. So, $[f(\mathbf{x}) \leq 0]$ evaluates to true when $f(\mathbf{x}) \leq 0$, and to false otherwise.

It would be convenient if these logic statements could be transformed into linear mixed-integer constraints, since optimization problem solvers that know how to deal with these constraints are available. Some useful transformations from logic statements into linear mixed-integer inequality constraints are given by [16]:

$$[f(\mathbf{x}) \leq 0] \wedge [\delta = 1] \text{ is true iff } f(\mathbf{x}) - \delta \leq -1 + \gamma_{\mathrm{m}}(1-\delta) \tag{3.1}$$

$$[f(\mathbf{x}) \leq 0] \vee [\delta = 1] \text{ is true iff } f(\mathbf{x}) \leq \gamma_{\mathrm{M}}\delta \tag{3.2}$$

$$\sim [f(\mathbf{x}) \leq 0] \text{ is true iff } f(\mathbf{x}) \geq \gamma_{\epsilon,\mathrm{mach}} \tag{3.3}$$

$$[f(\mathbf{x}) \leq 0] \Rightarrow [\delta = 1] \text{ is true iff } f(\mathbf{x}) \geq \gamma_{\epsilon,\mathrm{mach}} + (\gamma_{\mathrm{m}} - \gamma_{\epsilon,\mathrm{mach}})\delta \tag{3.4}$$

$$[f(\mathbf{x}) \leq 0] \Leftrightarrow [\delta = 1] \text{ is true iff } \begin{cases} f(\mathbf{x}) \leq \gamma_{\mathrm{M}}(1-\delta) \\ f(\mathbf{x}) \geq \gamma_{\epsilon,\mathrm{mach}} + (\gamma_{\mathrm{m}} - \gamma_{\epsilon,\mathrm{mach}})\delta, \end{cases} \tag{3.5}$$

where $f : \mathbb{R}^{n_{\mathbf{x}}} \to \mathbb{R}$ is linear, $\mathbf{x} \in \mathcal{X}$, $\mathcal{X}$ is a given bounded set, $\gamma_{\epsilon,\mathrm{mach}}$ is a small positive constant, e.g., the machine precision, which indicates when a constraint is considered to be violated, and where

$$\gamma_{\mathrm{M}} = \max_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}) \tag{3.6}$$

$$\gamma_{\mathrm{m}} = \min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}). \tag{3.7}$$

Remark 3.1 Formally $\sim [f(\mathbf{x}) \leq 0]$ is true iff $f(\mathbf{x}) > 0$. However, for numerical reasons optimization problem solvers cannot deal with such a strict inequality. Therefore in (3.3) the strict inequality $f(\mathbf{x}) > 0$ is approximated by the inequality $f(\mathbf{x}) \geq \gamma_{\epsilon,\mathrm{mach}}$. In practice, for a sufficiently small value of $\gamma_{\epsilon,\mathrm{mach}}$ this approximation is typically acceptable. □

As we will see in Section 3.3, as a byproduct of transforming logic statements into mixed-integer constraints, constraints involving products of logical variables and constraints involving products of continuous and logical variables may appear. Although these products are not linear, they can be transformed into linear inequalities. E.g., the product term $\delta_1\delta_2$ can be replaced by an auxiliary binary variable δ_3. The value of variable δ_3 should be 1, when the values of both δ_1 and δ_2 are 1, and 0 otherwise. This behavior can be expressed in a logic statement and corresponding linear inequalities as follows [16]:

$$[\delta_3 = 1] \Leftrightarrow ([\delta_1 = 1] \wedge [\delta_2 = 1]) \text{ is true iff } \begin{cases} -\delta_1 + \delta_3 \leq 0 \\ -\delta_2 + \delta_3 \leq 0 \\ \delta_1 + \delta_2 - \delta_3 \leq 1. \end{cases} \tag{3.8}$$

Also, the product term $\delta f(\mathbf{x})$, for a linear function $f : \mathbb{R}^{n_{\mathbf{x}}} \to \mathbb{R}$ and $\delta \in \{0,1\}$, can be transformed into linear inequalities. The product term $\delta f(\mathbf{x})$ is replaced by an auxiliary variable z. The value of variable z should be $f(\mathbf{x})$ when the value of δ is 1, and 0 otherwise. This behavior can be expressed and transformed into linear inequality constraints as follows [16]:

$$([\delta = 1] \Rightarrow [z = f(\mathbf{x})]) \wedge (\sim [\delta = 1] \Rightarrow [z = 0]) \text{ is true iff } \begin{cases} z \leq \gamma_{\mathrm{M}}\delta \\ z \geq \gamma_{\mathrm{m}}\delta \\ z \leq f(\mathbf{x}) - \gamma_{\mathrm{m}}(1-\delta) \\ z \geq f(\mathbf{x}) - \gamma_{\mathrm{M}}(1-\delta), \end{cases} \tag{3.9}$$

where γ_{M} and γ_{m} are as defined in (3.6)–(3.7). Note that in fact the relations (3.8) and (3.9) transform if-then-else statements into linear inequality constraints.

3.2.3 Mixed-logical dynamic models

A prediction model M based on the transformations discussed above can be cast into mixed-logical dynamic form to obtain a compact representation of the hybrid dynamics as follows [16]:

$$\begin{aligned}
\mathbf{x}(k+1) &= \mathbf{A}\mathbf{x}(k)+\mathbf{B}_1\mathbf{u}(k)+\mathbf{B}_2\boldsymbol{\delta}(k)+\mathbf{B}_3\mathbf{z}(k)\\
\mathbf{y}(k) &= \mathbf{C}\mathbf{x}(k)+\mathbf{D}_1\mathbf{u}(k)+\mathbf{D}_2\boldsymbol{\delta}(k)+\mathbf{D}_3\mathbf{z}(k)\\
\mathbf{E}_2\boldsymbol{\delta}(k)+\mathbf{E}_3\mathbf{z}(k) &\leq \mathbf{E}_1\mathbf{u}(k)+\mathbf{E}_4\mathbf{x}(k)+\mathbf{E}_5,
\end{aligned}$$

where

$$\mathbf{x}(k)=\begin{bmatrix}\mathbf{x}_{\mathrm{c}}(k)\\ \mathbf{x}_{\mathrm{b}}(k)\end{bmatrix} \qquad \mathbf{y}(k)=\begin{bmatrix}\mathbf{y}_{\mathrm{c}}(k)\\ \mathbf{y}_{\mathrm{b}}(k)\end{bmatrix} \qquad \mathbf{u}(k)=\begin{bmatrix}\mathbf{u}_{\mathrm{c}}(k)\\ \mathbf{u}_{\mathrm{b}}(k)\end{bmatrix},$$

are the state, output, and input, respectively, separated into continuous components and binary components, i.e., $\mathbf{x}_{\mathrm{c}}(k) \in \mathbb{R}^{n_{\mathbf{x}_{\mathrm{c}}}}$, $\mathbf{x}_{\mathrm{b}}(k) \in \mathbb{R}^{n_{\mathbf{x}_{\mathrm{b}}}}$, $n_{\mathbf{x}} = n_{\mathbf{x}_{\mathrm{c}}} + n_{\mathbf{x}_{\mathrm{b}}}$, $\mathbf{y}_{\mathrm{c}}(k) \in \mathbb{R}^{n_{\mathbf{y}_{\mathrm{c}}}}$, $\mathbf{y}_{\mathrm{b}}(k) \in \mathbb{R}^{n_{\mathbf{y}_{\mathrm{b}}}}$, $n_{\mathbf{y}} = n_{\mathbf{y}_{\mathrm{c}}} + n_{\mathbf{y}_{\mathrm{b}}}$, $\mathbf{u}_{\mathrm{c}}(k) \in \mathbb{R}^{n_{\mathbf{u}_{\mathrm{c}}}}$, $\mathbf{u}_{\mathrm{b}}(k) \in \mathbb{R}^{n_{\mathbf{u}_{\mathrm{b}}}}$, $n_{\mathbf{u}} = n_{\mathbf{u}_{\mathrm{c}}} + n_{\mathbf{u}_{\mathrm{b}}}$. In addition, $\boldsymbol{\delta}(k)$ are the binary variables and $\mathbf{z}(k)$ are the auxiliary continuous variables.

3.3 Application: Household energy optimization

In this section we consider a decentralized multi-agent single-layer MPC approach for controlling energy in households. We discuss distributed energy resources, formalize the hybrid dynamics of a household in a model, and show how this model can be used for MPC control.

3.3.1 Distributed energy resources

Distributed energy resources, comprising distributed power generators, electricity storage units, and responsive loads, can play a crucial role in supporting the European Union's key policy objectives of market liberalization, combating climate change, increasing the amount of electricity generated from renewable sources, and enhancing energy saving. Large-scale diffusion of distributed energy resources will have a profound impact on the functioning of the electricity infrastructure: It will bring radical changes to the traditional model of generation and supply as well as to the business model of the energy industry [67]. Drivers for distributed energy resources are the generation and sale of electric energy and accompanying goods, such as CO_2 emission rights, and the provision of ancillary services for network operators.

Distributed generation of electricity, e.g., via photo-voltaics, wind turbines, or combined heat and power plants, has a good chance of pervading the electricity infrastructure in the future [67, 120]. Distributed generation offers environmental benefits (e.g., due to the use of renewable energy sources and the efficient use of fossil fuels), reduced investment risks, fuel diversification and energy autonomy, and increased energy efficiency (e.g., due to fewer line losses and co-generation options). In addition, several electricity storage technologies are under development, e.g., lithium-ion batteries and plug-in hybrid electric vehicles [91]. Furthermore, options for load response are foreseen for the future power system [24].

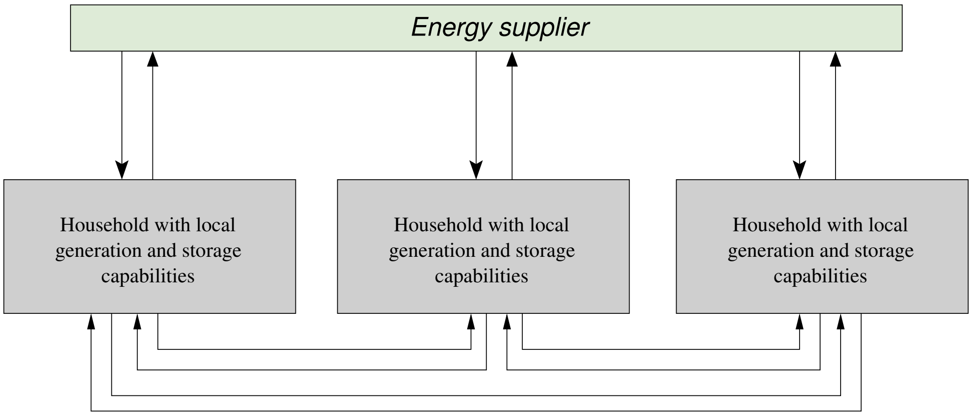


Figure 3.2: Households and an external energy supplier. Households can buy and sell energy to and from an external energy supplier or to and from neighboring households.

With an increase in distributed energy resources combined with more ICT and intelligence in the power network, the options for consumers with respect to energy demand response increase. In this section, we focus on residential distributed energy resources. Households with distributed energy resources operate more independently of energy suppliers, they can devise new contractual arrangements with suppliers and network managers, and they can buy and sell power among one another, and to and from their supplier, as shown in Figure 3.2. As a first step toward developing control structures that are installed in households for optimizing energy usage, we consider an individual household, not taking into account the possibility of energy exchange with neighboring households, i.e., we consider a decentralized multi-agent single-layer control structure[1].

3.3.2 System description

The system under study consists of a household interacting with its energy supplier, as depicted in Figure 3.3. As in conventional households, the household can buy electricity and gas from its energy supplier. In addition to this, the household can sell electricity to the energy supplier. The household can produce this electricity using a micro combined heat and power (μCHP) unit [120]. This unit can simultaneously produce heat and power for the household. It is typically located in a basement, underneath a sink, hanging from a wall, or outside. It can provide various energy needs, such as space and water heating, electricity, and, possibly, cooling.

We assume that the μCHP unit in the household is based on Stirling technology [120]. The unit provides electricity to an electricity storage unit, and heat to a heat storage unit. The μCHP unit consists of a Stirling engine prime mover, conversion unit 1, and an auxiliary burner, conversion unit 2. Conversion unit 1 converts natural gas $z_{\mathrm{g},1}(k)$ (in kWh) into

[1]The control agent that we will develop for control of a household could be located in a physical device such as the Qbox, which will soon become commerically available. See the website of Qurrent, the manufacturer of the Qbox, at http://www.qurrent.com/.

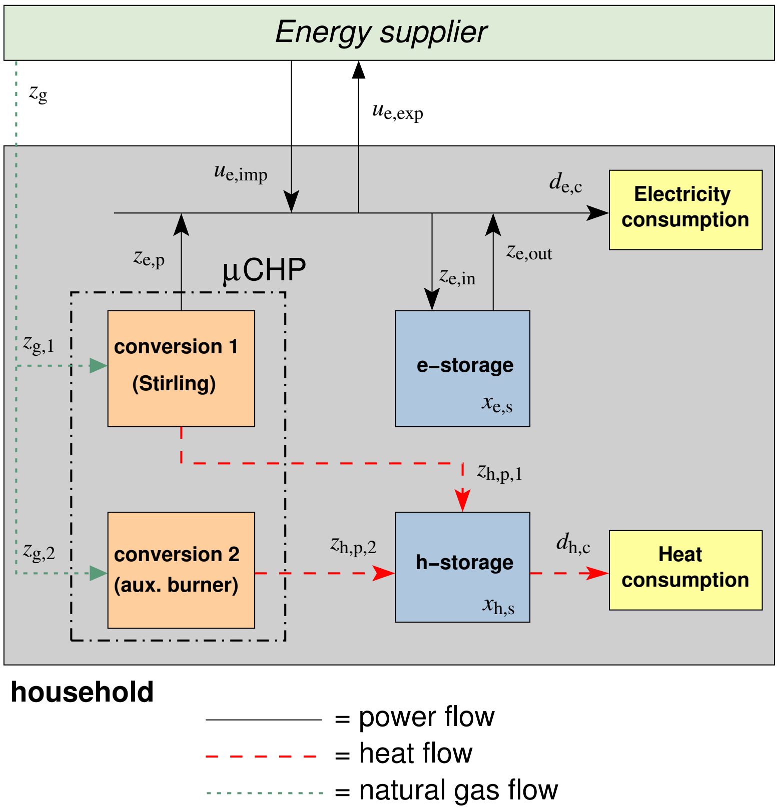


Figure 3.3: Conceptual overview of the system under study [68].

electricity produced $z_{e,p}(k)$ (in kWh) and heat produced $z_{h,p,1}(k)$ (in kWh), with a fixed ratio. The conversion unit can operate in *partial* or *full* mode and has a minimum activation constraint. Conversion unit 2 converts natural gas $z_{g,2}(k)$ (in kWh) to provide additional heat $z_{h,p,2}(k)$ (in kWh). For energy efficiency reasons conversion unit 2 should be used as backup heat generator only. Therefore, priority has to be given to conversion unit 1. Conversion units 1 and 2 are equipped with built-in fixed controllers that are designed to keep the level of the heat storage unit $x_{h,s}(k)$ (in kWh) between certain upper and lower bounds.

The generated heat is supplied to a heat storage unit in the form of hot water. We consider an aggregated heat demand for the household, and therefore make no distinction between heat storage units for, e.g., space heating and sanitation heating. It is therefore also appropriate to assume that there is a single large heat storage unit. Such a configuration is commercially available[2]. The level of the heat storage unit is indicated by the energy $x_{h,s}(k)$ (in kWh) in the heat storage unit. Heat consumption $d_{h,c}(k)$ (in kWh) takes heat from the storage unit, and therefore lowers the level of the heat storage unit $x_{h,s}(k)$. The level of the heat storage unit changes over time depending on the heat produced by the conversion units and the heat consumed.

[2] See, e.g., Gledhill Water Storage, http://www.gledhill.net/.

The generated electricity can be stored in a battery, e.g., a lithium-ion battery, of which the level is indicated by the energy in the battery $x_{\mathrm{e,s}}(k)$ (in kWh). Electricity can flow to and from the battery, represented by $z_{\mathrm{e,in}}(k)$ (in kWh) and $z_{\mathrm{e,out}}(k)$ (in kWh), respectively. In addition to storing electricity, electricity can be used directly by the household for consumption, indicated by $d_{\mathrm{e,c}}(k)$ (in kWh), or it can be sold to the supplier through export, indicated by $u_{\mathrm{e,exp}}(k)$ (in kWh). Electricity can also be imported from the supplier through import, indicated by $u_{\mathrm{e,imp}}(k)$ (in kWh). The level of the electricity storage unit changes over time depending on the electricity produced by conversion unit 1, the electricity imported from or exported to the energy supplier, and the electricity consumed by the household.

System dynamics

Below we formalize the dynamics of the household. As we will see, these dynamics are hybrid, and the transformations from Section 3.2.2 can be used to obtain a prediction model consisting of only linear mixed-integer equality and inequality constraints.

Conversion unit 1 Conversion unit 1 can operate at partial generation or full generation. The control inputs are therefore $u_{1,\mathrm{part}}(k) \in \{0,1\}$ and $u_{1,\mathrm{full}}(k) \in \{0,1\}$, where input $u_{1,\mathrm{full}}(k)$ can only be used when $u_{1,\mathrm{part}}(k) = 1$. Depending on the control inputs, the conversion unit uses a different amount of gas $z_{\mathrm{g},1}(k)$. Conversion unit 1 converts this gas into electricity $z_{\mathrm{e,p}}(k)$ and heat $z_{\mathrm{h,p},1}(k)$. The gas used $z_{\mathrm{g},1}(k)$, the electricity provided to the internal network $z_{\mathrm{e,p}}(k)$, and the heat provided to the heat storage unit $z_{\mathrm{h,p},1}(k)$ are given by:

$$\begin{aligned}
z_{\mathrm{g},1}(k) &= \eta_{\mathrm{g,part}} u_{1,\mathrm{part}}(k) + (\eta_{\mathrm{g,max}} - \eta_{\mathrm{g,part}}) u_{1,\mathrm{full}}(k) \\
z_{\mathrm{e,p}}(k) &= \eta_{\mathrm{e}} z_{\mathrm{g},1}(k) \\
z_{\mathrm{h,p},1}(k) &= (\eta_{\mathrm{tot}} - \eta_{\mathrm{e}}) z_{\mathrm{g},1}(k),
\end{aligned}$$

where $\eta_{\mathrm{g,part}}$ (in kWh) is the gas used when the conversion unit operates partially, $\eta_{\mathrm{g,max}}$ (in kWh) is the gas used when the conversion unit operates at its maximum, η_{e} is the electric efficiency of the unit, and η_{tot} is the total efficiency of the unit, i.e., the electric and the heat efficiency together.

When the conversion unit is in operation the dynamics of the household will be different from when the conversion unit is not in operation. In order to model logic rules relying on such information, a device-in-operation variable that indicates when conversion unit 1 is in operation is used. Based on the actuator setting $u_{1,\mathrm{part}}(k)$, which takes on binary values 0 and 1, the device-in-operation indicator $\delta_{\mathrm{dio},1}(k) \in \{0,1\}$ is defined as:

$$[\delta_{\mathrm{dio},1}(k) = 1] \Leftrightarrow [u_{1,\mathrm{part}}(k) = 1],$$

which can be directly transformed into the linear equality constraint:

$$\delta_{\mathrm{dio},1}(k) = u_{1,\mathrm{part}}(k).$$

Using the device-in-operation variable $\delta_{\mathrm{dio},1}(k)$, the constraint that the full generation can only be switched on after the partial generation $u_{1,\mathrm{full}}(k)$ has been switched on is modeled with the inequality constraint:

$$u_{1,\mathrm{full}}(k) - \delta_{\mathrm{dio},1}(k) \leq 0.$$

Conversion unit 1 has a minimum activation constraint to avoid fast wear and tear of the device due to frequent on and off switching. The minimum activation constraint specifies that when the device has been switched on it has to stay in operation for at least $\eta_{\text{act,min}} \in \mathbb{N}^{+}$ time units, with $\mathbb{N}^{+}$ the positive natural numbers. In order to model the minimum activation constraint, introduce the counter $x_{\text{act}}(k) \in [0, x_{\text{act,max}}]$ (with $x_{\text{act,max}}$ a finite upper bound on the maximum time that a device can be in operation), which counts the number of time units that the device has been in operation so far. The evolution of this variable is given by the relation:

$$x_{\text{act}}(k+1) = \begin{cases} x_{\text{act}}(k)+1 & \text{if } \delta_{\text{dio},1}(k) = 1 \\ 0 & \text{otherwise.} \end{cases}$$

Using (3.9) this relation can be transformed into mixed-integer inequality constraints.

If the activation $x_{\text{act}}(k)$ is 0, then the conversion unit is allowed to stay switched off or to be switched on. However, if the activation $x_{\text{act}}(k)$ is larger than 0, then the conversion unit is not allowed to be switched off, until the activation $x_{\text{act}}(k)$ has reached the minimum activation $\eta_{\text{act,min}}$. Hence, as long as $x_{\text{act}}(k)$ is larger than 0 and smaller than $\eta_{\text{act,min}}$, the value of input $u_{1,\text{part}}(k)$ should stay at its maximum, i.e., 1. After the activation $x_{\text{act}}(k)$ has reached the minimum activation, the input $u_{1,\text{part}}(k)$ is allowed to have a different value again. To model this, introduce a constraint on the minimum value of $u_{1,\text{part}}(k)$ as follows:

$$u_{1,\text{part,min}}(k) \leq u_{1,\text{part}}(k), \tag{3.10}$$

with $u_{1,\text{part,min}}(k) \in \{0,1\}$. Using the activation variable $x_{\text{act}}(k)$ and this constraint we can enforce the minimum activation constraint by adjusting the lower limit $u_{1,\text{part,min}}(k)$ of $u_{1,\text{part}}(k)$ with the relation:

$$[1 \leq x_{\text{act}}(k) \leq \eta_{\text{act,min}} - 1] \Leftrightarrow [u_{1,\text{part,min}}(k) = 1].$$

To transform this relation we introduce auxiliary binary variables $\delta_1(k)$, $\delta_2(k)$, and $\delta_3(k)$ for which it holds that:

$$\begin{aligned} [1 \leq x_{\text{act}}(k)] &\Leftrightarrow [\delta_1(k) = 1] \\ [x_{\text{act}}(k) \leq \eta_{\text{act,min}} - 1] &\Leftrightarrow [\delta_2(k) = 1] \\ [\delta_3(k) = 1] &\Leftrightarrow [\delta_1(k) = 1] \wedge [\delta_2(k) = 1]. \end{aligned}$$

Hence, when $\delta_3(k)$ is equal to 1, then $x_{\text{act}}(k)$ is larger than 0, although it has not yet passed the minimum activation $\eta_{\text{act,min}}$, implying that the conversion unit should be kept in operation. To transform these three relations into mixed-integer inequality constraints, (3.5) and (3.8) are used.

Variable $\delta_3(k)$ is 1 if the device should be kept in operation, and 0 otherwise. This behavior is exactly the same behavior as variable $u_{1,\text{part,min}}(k)$ should have. Therefore, $u_{1,\text{part,min}}(k) = \delta_3(k)$, and the constraint that the conversion unit can only be switched off after an activation of $\eta_{\text{act,min}}$ is enforced by substituting $\delta_3(k)$ for $u_{1,\text{part,min}}(k)$ in (3.10).

A fixed controller is installed in conversion unit 1. This fixed controller is installed to guarantee a minimum level of heat in the heat storage unit. The fixed controller switches the conversion unit on when the level of the heat storage unit $x_{\text{h,s}}(k)$ is lower than a lower limit $\eta_{\text{h,s,lim,min},1}$, and switches it off when the level of the heat storage unit $x_{\text{h,s}}(k)$ is larger than an upper limit $\eta_{\text{h,s,lim,max,1}}$. Let $u_{1,\text{part,tmp}}(k) \in \{0,1\}$ denote the actuator setting that the fixed controller would choose if the minimum activation constraint would not be present. The fixed controller determines the value for this variable as follows:

$$u_{1,\text{part,tmp}}(k) = \begin{cases} 1 & \text{for } x_{\text{h,s}}(k) \leq \eta_{\text{h,s,lim,min,1}} \\ u_{1,\text{part}}(k-1) & \text{for } \eta_{\text{h,s,lim,min,1}} < x_{\text{h,s}}(k) < \eta_{\text{h,s,lim,max,1}} \\ 0 & \text{for } x_{\text{h,s}}(k) \geq \eta_{\text{h,s,lim,max,1}}. \end{cases}$$

To transform this relation auxiliary variables $\delta_4(k)$, $\delta_5(k)$, $\delta_6(k)$, and $\delta_7(k)$ are defined such that:

$$\begin{aligned} [\delta_4(k) = 1] &\Leftrightarrow [x_{\text{h,s}}(k) \leq \eta_{\text{h,s,lim,min,1}}] \\ [\delta_5(k) = 1] &\Leftrightarrow [x_{\text{h,s}}(k) \geq \eta_{\text{h,s,lim,max,1}}] \\ [\delta_6(k) = 1] &\Leftrightarrow [\delta_4(k) = 0] \wedge [\delta_5(k) = 0] \\ \delta_7(k) &= \delta_6(k) u_{1,\text{part}}(k-1). \end{aligned}$$

Using (3.5) and (3.8) these relations are transformed into linear mixed-integer constraints. Given the values for these auxiliary variables, the fixed controller determines the value for $u_{1,\text{part,tmp}}(k)$ as:

$$u_{1,\text{part,tmp}}(k) = 1.\delta_4(k) + 0.\delta_5(k) + \delta_7(k).$$

In determining the actual setting for conversion unit 1, the fixed controller has to respect the minimum activation constraint. Therefore, the value that the fixed controller of conversion unit 1 chooses as input $u_{1,\text{part}}(k)$ to the actuator of conversion unit 1 is not the value of $u_{1,\text{part,tmp}}(k)$ directly, but the value determined as follows:

$$u_{1,\text{part}}(k) = \begin{cases} 1 & \text{if the conversion unit is not allowed to switch off} \\ u_{1,\text{part,tmp}}(k) & \text{otherwise,} \end{cases}$$

which can be written as:

$$u_{1,\text{part}}(k) = 1.\delta_3(k) + (1 - \delta_3(k)) u_{1,\text{part,tmp}}(k),$$

where $\delta_3(k)$ is defined through the minimum activation constraints. This relation can be transformed into linear mixed-integer constraints using (3.8).

Conversion unit 2 Conversion unit 2 has as control input $u_2(k) \in [0, u_{2,\text{max}}]$. Depending on the control input, it uses a different amount of gas $z_{\text{g},2}(k)$ and provides a different amount of heat $z_{\text{h,p},2}(k)$ to the heat storage unit. The gas used $z_{\text{g},2}(k)$ and the heat provided $z_{\text{h,p},2}(k)$ are given by:

$$z_{\text{g},2}(k) = u_2(k) \tag{3.11}$$

$$z_{\text{h,p},2}(k) = \eta_{\text{tot}} z_{\text{g},2}(k). \tag{3.12}$$

A device-in-operation variable $\delta_{\text{dio},2}(k) \in \{0,1\}$ indicating when conversion unit 2 is in operation is defined as:

$$[u_2(k) \geq \gamma_{\epsilon,\text{mach}}] \Leftrightarrow [\delta_{\text{dio},2}(k) = 1].$$

This relation can be converted into linear mixed-integer inequality constraints using (3.5). The device-in-operation variable $\delta_{\mathrm{dio},2}(k)$ is used to enforce that conversion unit 2 is in operation only when conversion unit 1 is in operation through the following constraints:

$$\delta_{\mathrm{dio},2}(k) - \delta_{\mathrm{dio},1}(k) \leq 0. \tag{3.13}$$

A fixed controller is installed in conversion unit 2, similar to the fixed controller as in conversion unit 1. The fixed controller of conversion unit 2 determines an auxiliary actuator setting $u_{2,\mathrm{tmp}}(k) \in \{0,1\}$ as follows:

$$u_{2,\mathrm{tmp}}(k) = \begin{cases} 1 & \text{for } x_{\mathrm{h,s}}(k) \leq \eta_{\mathrm{h,s,lim,min},2} \\ u_{2,\mathrm{tmp}}(k-1) & \text{for } \eta_{\mathrm{h,s,lim,min},2} < x_{\mathrm{h,s}}(k) < \eta_{\mathrm{h,s,lim,max},2} \\ 0 & \text{for } x_{\mathrm{h,s}}(k) \geq \eta_{\mathrm{h,s,lim,max},2}. \end{cases}$$

To transform this relation, auxiliary variables $\delta_9(k)$, $\delta_{10}(k)$, $\delta_{11}(k)$, and $\delta_{12}(k)$ are defined such that:

$$\begin{aligned} [\delta_9(k) = 1] &\Leftrightarrow [x_{\mathrm{h,s}}(k) \leq \eta_{\mathrm{h,s,lim,min},2}] \\ [\delta_{10}(k) = 1] &\Leftrightarrow [x_{\mathrm{h,s}}(k) \geq \eta_{\mathrm{h,s,lim,max},2}] \\ [\delta_{11}(k) = 1] &\Leftrightarrow [\delta_9(k) = 0] \wedge [\delta_{10}(k) = 0] \\ \delta_{12}(k) &= \delta_{11}(k) u_{2,\mathrm{tmp}}(k-1). \end{aligned}$$

Using (3.5) and (3.8) these relations are transformed into linear mixed-integer constraints. The fixed controller now determines the value for the auxiliary actuator setting $u_{2,\mathrm{tmp}}(k)$ as:

$$u_{2,\mathrm{tmp}}(k) = 1.\delta_9(k) + 0.\delta_{10}(k) + \delta_{12}(k).$$

The auxiliary actuator setting $u_{2,\mathrm{tmp}}(k)$ is used by the fixed controller to determine the actual input for conversion unit 2 as:

$$u_2(k) = u_{2,\mathrm{tmp}}(k) \eta_{\mathrm{frac}} u_{2,\max},$$

where η_{frac} is the part of the maximum output $u_{2,\max}$ that is activated when conversion unit 2 is switched on by the fixed controller.

Electricity and heat storage units The electricity and heat storage units are used to store energy. The storage units have a limited capacity. The level of the electricity storage unit $x_{\mathrm{e,s}}(k)$ is determined by the amount of electricity $z_{\mathrm{e,in}}(k)$ that goes into the storage unit, and the amount of electricity $z_{\mathrm{e,out}}(k)$ that is taken out. It is assumed that the charging and discharging of the battery is without energy loss. The dynamics of the level of the electricity storage unit are given by:

$$x_{\mathrm{e,s}}(k+1) = x_{\mathrm{e,s}}(k) + z_{\mathrm{e,in}}(k) - z_{\mathrm{e,out}}(k).$$

The level of the heat storage unit $x_{\mathrm{h,s}}(k)$ is influenced by the heat production of conversion units 1 and 2, i.e., $z_{\mathrm{h,p},1}(k)$ and $z_{\mathrm{h,p},2}(k)$, resepectively. The heat storage unit dynamics are given by:

$$x_{\mathrm{h,s}}(k+1) = x_{\mathrm{h,s}}(k) + z_{\mathrm{h,p},1}(k) + z_{\mathrm{h,p},2}(k) - d_{\mathrm{h,c}}(k).$$

The levels of the electricity and heat storage units are limited by minimum and maximum values, i.e.:

$$x_{\mathrm{e,s,min}} \leq x_{\mathrm{e,s}}(k) \leq x_{\mathrm{e,s,max}}$$
$$x_{\mathrm{h,s,min}} \leq x_{\mathrm{h,s}}(k) \leq x_{\mathrm{h,s,max}}.$$

Power balance A power balance relating the power output of conversion unit 1 $z_{\mathrm{e,p}}(k)$, the input $z_{\mathrm{e,in}}(k)$ and output $z_{\mathrm{e,out}}(k)$ of the electricity storage unit, the electricity consumption $d_{\mathrm{e,c}}(k)$, and electricity bought $u_{\mathrm{e,imp}}(k)$ or sold $u_{\mathrm{e,exp}}(k)$ to the energy supplier, has to hold. This power balance is given by:

$$0 = z_{\mathrm{e,p}}(k) + u_{\mathrm{e,imp}}(k) + z_{\mathrm{e,out}}(k) - u_{\mathrm{e,exp}}(k) - z_{\mathrm{e,in}}(k) - d_{\mathrm{e,c}}(k).$$

3.3.3 MPC problem formulation

We now use the derived model as prediction model M for a control agent controlling the energy flows of a household. The control agent has the task to automatically determine which actions should be taken in order to minimize the operational costs of fulfilling residential electricity and heat requirements, while maintaining the level of the heat storage unit between a desired upper and lower limit, and respecting the operational constraints, including a minimal activation of 2 time units. The control agent uses an MPC strategy such that the control agent can:

- optimize the usage of the heat and electricity storage units;
- take into account the decision freedom due to electricity import and export possibilities, and generation of energy by itself;
- incorporate predictions on residential electricity and heat demands;
- incorporate models of the dynamics and constraints of installed generators and storage units.

MPC scheme

At each control cycle k the control agent makes a measurement of the system state consisting of values for the level of the heat storage unit $x_{\mathrm{h,s}}(k)$, the level of the electricity storage unit $x_{\mathrm{e,s}}(k)$, and the activation counter $x_{\mathrm{act}}(k)$. Then the control agent determines values for the control inputs $u_{1,\mathrm{full}}(k)$, $u_{\mathrm{e,imp}}(k)$, and $u_{\mathrm{e,exp}}(k)$ by solving the MPC optimization problem that minimizes an objective function, subject to the prediction model M and initial constraints. Note that with respect to the conversion units, the control agent only determines $u_{1,\mathrm{full}}(k)$, since the values for $u_{1,\mathrm{part}}(k)$ and $u_2(k)$ are determined by the fixed controllers installed in the conversion units.

Objective function The main objective of the control agent is to minimize the daily operational costs of residential energy use. These costs depend on the price p_{f} (euro/kWh) for gas consumption, the price $p_{\mathrm{imp}}(k)$ (euro/kWh) at which electricity can be bought, and the price p_{exp} (euro/kWh) at which electricity can be sold. Note that in principle, the prices for gas, electricity import and electricity output vary over the day. However, as a first step we assume that the price for gas consumption and power export are constant, whereas the price for importing electricity varies over the day.

In addition to minimizing the daily operational cost, the control agent should also maintain the level of the heat storage unit between the desired upper and lower limit. This goal is included as a soft constraint by penalizing an auxiliary variable $z_{\mathrm{aux}}(k+l) \geq 0$, for $l = \{1,\ldots,N\}$, with a large positive cost p_{soft}. This auxiliary variable $z_{\mathrm{aux}}(k+l)$ is defined such that:

$$z_{\mathrm{aux}}(k+l) = \begin{cases} x_{\mathrm{h,s}}(k+l) - \eta_{\mathrm{h,s,lim,max}} & \text{for } x_{\mathrm{h,s}}(k+l) \geq \eta_{\mathrm{h,s,lim,max}} \\ 0 & \text{for } \eta_{\mathrm{h,s,lim,min}} < x_{\mathrm{h,s}}(k+l) < \eta_{\mathrm{h,s,lim,max}} \\ \eta_{\mathrm{h,s,lim,min}} - x_{\mathrm{h,s}}(k+l) & \text{for } x_{\mathrm{h,s}}(k+l) \leq \eta_{\mathrm{h,s,lim,min}}, \end{cases}$$

which in combination with the minimization of the term $p_{\mathrm{soft}} z_{\mathrm{aux}}(k+l)$ can also be written as:

$$\eta_{\mathrm{h,s,lim,min}} - z_{\mathrm{aux}}(k+l) \leq x_{\mathrm{h,s}}(k+l) \leq \eta_{\mathrm{h,s,lim,max}} + z_{\mathrm{aux}}(k+l).$$

The cost function at control cycle k over a prediction horizon of N control cycles, including the cost for the soft constraints, is defined as:

$$J = \sum_{l=0}^{N-1} \Big(p_{\mathrm{f}} \big(z_{\mathrm{g},1}(k+l) + z_{\mathrm{g},2}(k+l) \big) + p_{\mathrm{imp}}(k+l) u_{\mathrm{e,imp}}(k+l) \\ - p_{\mathrm{exp}} u_{\mathrm{e,exp}}(k+l) + p_{\mathrm{soft}} z_{\mathrm{aux}}(k+1+l) \Big).$$

Note that p_{soft} should not be chosen too larger, since otherwise minimizing $z_{\mathrm{aux}}(k+l)$ has too much weight.

Prediction model The prediction model M that the control agent uses is based on the relations that describe the system model as given in Section 3.3.2, specified over the prediction horizon. Hence, the prediction model M consists of a large system of linear mixed-integer equality and inequality constraints. The values of the parameters of the prediction model are given in Table 3.1.

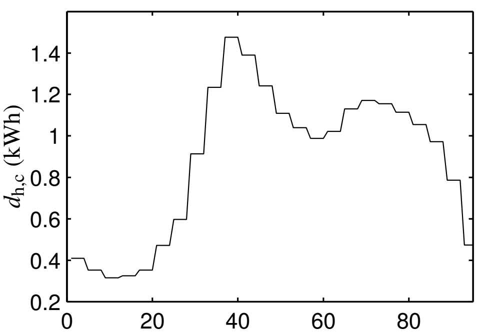
Initial constraints The initial constraints for $k = 1$ are:

$$\begin{aligned} x_{\mathrm{e,s}}(k) &= \bar{x}_{\mathrm{e,s}}(k) \\ x_{\mathrm{h,s}}(k) &= \bar{x}_{\mathrm{h,s}}(k) \\ x_{\mathrm{act}}(k) &= \bar{x}_{\mathrm{act}}(k) \\ u_{1,\mathrm{part}}(k-1) &= \bar{u}_{1,\mathrm{part}}(k-1) \\ u_{2,\mathrm{tmp}}(k-1) &= \bar{u}_{2,\mathrm{tmp}}(k-1), \end{aligned}$$

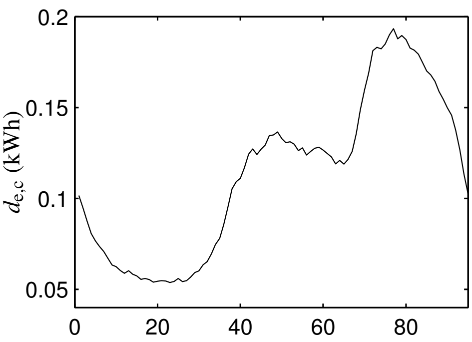
where the variables with a bar are known, e.g., through measurements.

parameter	value	parameter	value
$u_{2,\text{max}}$	4.9383	$\eta_{\text{g,max}}$	1.8333
$x_{\text{act,max}}$	1.10^6	$\eta_{\text{g,part}}$	0.9167
$x_{\text{e,s,max}}$	2	η_{tot}	1.0125
$x_{\text{e,s,min}}$	0	$\eta_{\text{h,s,lim,max}}$	8.1278
$x_{\text{h,s,max}}$	9.1728	$\eta_{\text{h,s,lim,max,1}}$	6.9667
$x_{\text{h,s,min}}$	0	$\eta_{\text{h,s,lim,max,2}}$	5.2250
$\gamma_{\epsilon,\text{mach}}$	1.10^{-8}	$\eta_{\text{h,s,lim,min}}$	2.3222
$\eta_{\text{act,min}}$	2	$\eta_{\text{h,s,lim,min,1}}$	4.0639
η_{e}	0.15	$\eta_{\text{h,s,lim,min,2}}$	2.9028
η_{frac}	0.6		

Table 3.1: Values of the parameters of the household system.



(a) Heat demand data.



(b) Electricity demand data.

Figure 3.4: Energy demand data for average Dutch household on January 29. One time unit corresponds to 15 minutes.

Solving the optimization problem The MPC optimization problem is a mixed-integer linear programming problem. It is linear, since the objective function and all constraints are linear and it is mixed integer, since the problem involves continuous and discrete variables. For solving the optimization problem at each control cycle we use the ILOG CPLEX v10.0 [71] linear mixed-integer programming solver through the Tomlab v5.7 interface [66] in Matlab v7.3 [98].

3.3.4 Simulations

To illustrate the operation of the proposed controller, we perform experiments for a particular winter day, January 29, 2006. For this day, average residential electricity and aggregated heat demand profiles have been created with 2006 data from 'EnergieNed', the Dutch Federation of Energy Companies. Figures 3.4(a) and 3.4(b) show the heat and electricity demand profiles of an average household on this day. Given such information, the control agent of

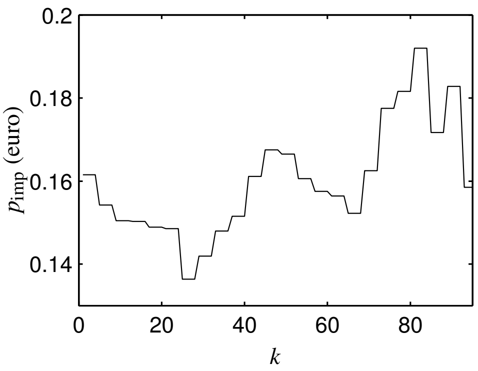


Figure 3.5: Electricity import price per kWh for January 29, 2006. One time unit corresponds to 15 minutes.

the household determines every 15 minutes new actions by solving its MPC problem at that time. To set up the control problem prices for electricity import, electricity export, and gas consumption have to be calculated first.

Price calculation

The variable electricity import price $p_{\mathrm{imp}}(k)$ is calculated as follows. The Dutch Central Bureau of Statistics states a total electricity tariff for small consumers for 2006 of 194 euro/MWh[3] (household class: single tariff, 3000 kWh). The variable part of the total tariff (including energy and VAT taxes) is around 90 % of the total tariff[4], so this becomes 0.1746 euro/kWh. The variable supply part of the total tariff accounts for 32 % of the total tariff[3]. For this variable supply part we have substituted Dutch power exchange values taken from the Amsterdam Power Exchange data. In this way import prices as shown in Figure 3.5 were derived. For the value of the feedback tariff p_{exp} we have taken average 'EnergieNed' data for 2006, which gives 0.0601 euro/kWh.

The gas price p_{f} is determined as follows. At the website of the Dutch Central Bureau of Statistics, a total gas tariff for small consumers of 552 euro/1000 m^3 is given (for consumer class: 2000 m^3). According to the ECN website, 91 % of the gas tariff is variable (including taxes). This leads to a gas price of 0.50232 euro/m^3.

Simulations

Below we first illustrate the operation of the proposed MPC control agent for a particular setting of the prediction horizon length N. After that, we vary the length of the prediction horizon to see how this influences the performance over a day. We will then not only consider a household with fixed controllers in the conversion units installed, but also a household without these fixed controllers. This gives more freedom to the MPC control agent and is expected to improve the performance.

[3] See http://www.cbs.nl/, Dutch central bureau of statistics.
[4] See http://www.energie.nl/, Energy Research Center of The Netherlands (ECN).

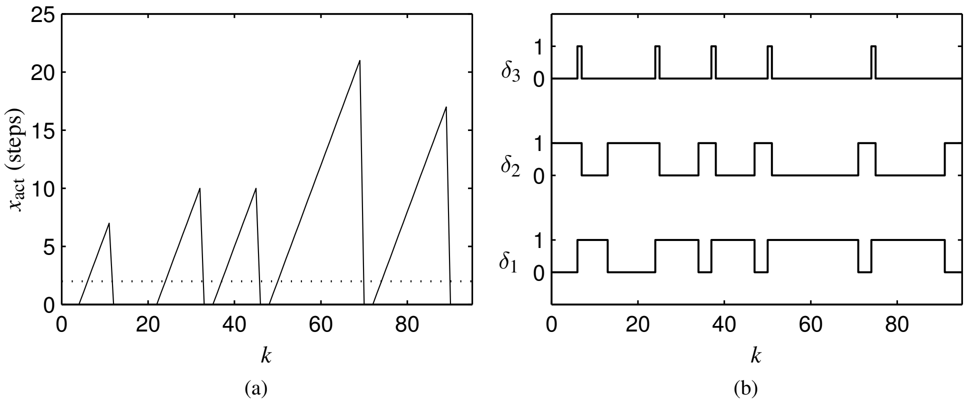


Figure 3.6: (a) Activation time $x_{act}(k)$ of conversion unit 1. The dotted horizontal line indicates the minimal activation time. (b) Evolution of δ_1, δ_2, and δ_3.

In principle the longer the prediction horizon is, the better the performance becomes. However, in practice the time required to solve the mixed-integer optimization problem restricts the length of the prediction horizon that can practically be used. To illustrate the operation of the proposed approach, we therefore below first consider a prediction horizon with length $N = 16$. The initial values for the simulation of the household are taken as:

$$\begin{aligned}
\bar{x}_{\mathrm{e,s}}(k) &= 0\\
\bar{x}_{\mathrm{h,s}}(k) &= 5.806\\
\bar{x}_{\mathrm{act}}(k) &= 0\\
\bar{u}_{1,\mathrm{part}}(k-1) &= 0\\
\bar{u}_{2,\mathrm{tmp}}(k-1) &= 0.
\end{aligned}$$

Results for $N = 16$

Figure 3.6(a) shows the activation time of conversion unit 1. Conversion unit 1 is switched on 5 times throughout the day, and stays in operation at least 2 time units. Hence, the constraints on the minimal activation time of 2 time units is respected. Figure 3.6(b) shows the evolution of the variables $\delta_1(k)$, $\delta_2(k)$, and $\delta_3(k)$ throughout the day. It is easy to verify that indeed, when conversion unit 1 is brought into operation, $\delta_3(k)$ becomes 1, and when conversion unit 1 has been in operation for at least 2 time units, $\delta_3(k)$ becomes 0 again.

Figure 3.7 shows the level of the heat storage unit. The fixed controllers installed in the conversion units should switch on the conversion units depending on the level of the heat storage unit. Figure 3.8(a) depicts the binary variables $\delta_4(k)$, $\delta_5(k)$, $\delta_6(k)$, and $\delta_7(k)$, which are used to indicate when conversion unit 1 should be switched on partially. In addition, Figure 3.8(b) shows the binary values used for determining the actuator values of conversion units 1 and 2. It is observed that, indeed, when the level of the heat storage unit reaches one of the lower limits, the respective conversion unit is switched on, whereas when the level reaches one of the upper limits, the respective conversion unit is switched

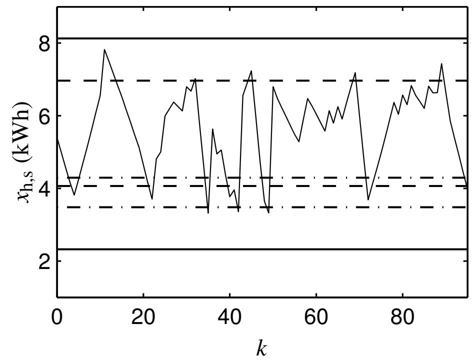


Figure 3.7: The level of the heat storage unit $x_{h,s}(k)$. The dashed and dashed-dotted lines indicate upper and lower activation bounds of the fixed controllers. The solid horizontal lines indicate physical upper and lower bounds.

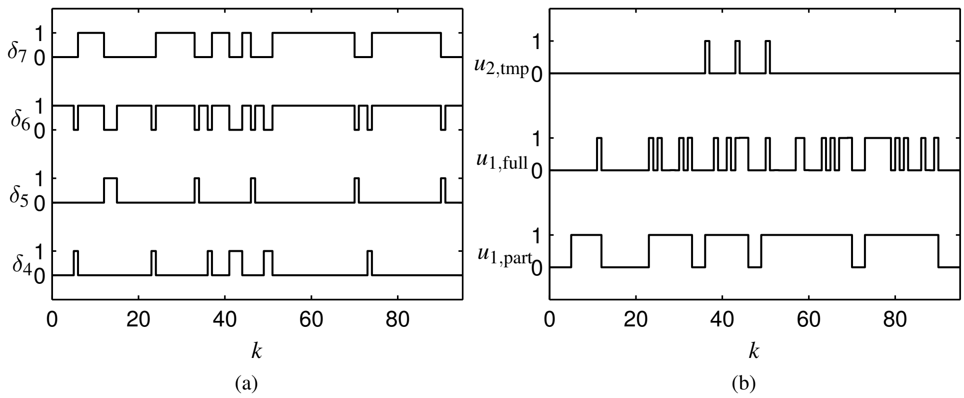


Figure 3.8: (a) Evolution of $\delta_4(k)$, $\delta_5(k)$, $\delta_6(k)$. (b) Evolution of the binary variables associated with the actuators of conversion units 1 and 2.

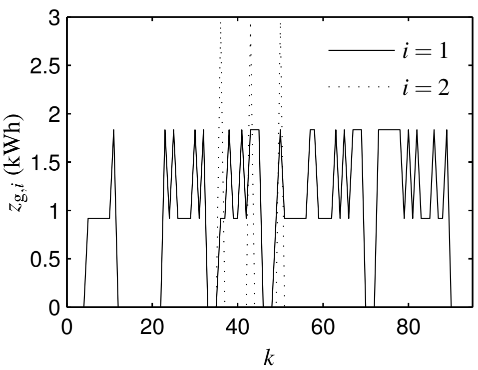


Figure 3.9: The gas consumed by the conversion units.

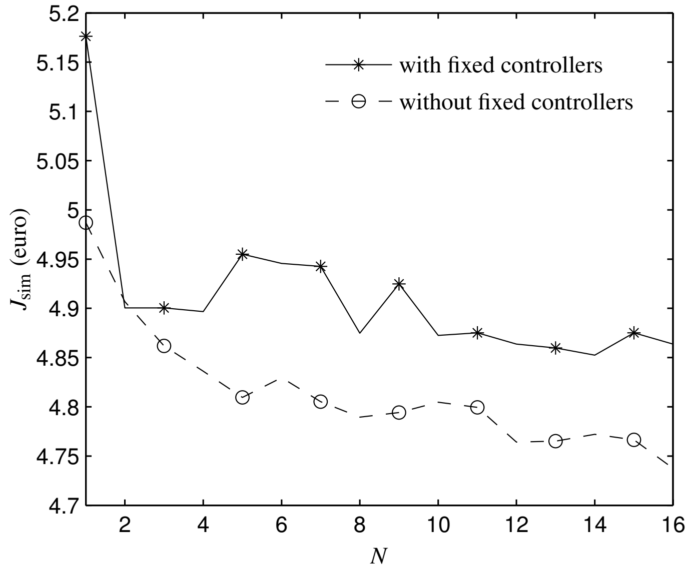


Figure 3.10: Performance J_{sim} for varying prediction horizon lengths N, both for the scenario in which the fixed controllers is installed, and for the scenario in which the fixed controllers are not installed.

off. Hence, the fixed controllers installed in the conversion units operate as they should. In addition, the MPC control agent decides to switch conversion unit 1 into full operation a number of times. When this happens, the MPC control agent has ensured that conversion unit 1 is already operating partially. Figure 3.9 shows the gas consumed by the conversion units, resulting from the actuator settings as chosen by the fixed controllers and the MPC control agent.

Results for varying prediction horizon lengths

We now consider the performance of the MPC control agent under varying lengths of the prediction horizon N. We consider two scenarios: the scenario considered so far, i.e., the scenario in which the MPC control agent controls the household including fixed controllers in the conversion units, and a scenario in which the fixed controllers in the conversion units are not present. In this second scenario, the MPC control agent has more decision freedom, since it can in the second scenario determine by itself when conversion unit 1 and conversion unit 2 should be switched on or off. Note that although the MPC control agent has this additional decision freedom, the prioritizing constraint for using conversion unit 1 before conversion unit 2, the constraint that conversion unit 1 should operate partially before switching to full operation, and the minimum activation time constraint for conversion unit 1 are still present.

Figure 3.10 shows the cost J_{sim} defined over the full simulation period for varying prediction horizon lengths[5]. For both scenarios, there is a general trend that as the prediction horizon length increases, the performance increases as well. However, since the control agent does not take into account the energy consumption patterns and electricity price fluctuations after its prediction horizon, it can choose actions that are not optimal over the full simulation. Therefore, in our case it is not strictly necessary that the performance increases with a longer prediction horizon. We also observe this in Figure 3.10. From the figure we also observe that if the fixed controllers are not present, that then, indeed, the MPC control agent can expoit the the increased decision freedom. This results in a higher performance for the scenario in which the fixed controllers are not installed.

Discussion

With a longer prediction horizon, the number of binary variables increases linearly. This also implies that the computations involved in solving the corresponding MPC optimization problem increase. The household system that we consider does not go to a stable or steady state, since the electricity and heat consumption continuously keep varying. Therefore, in principle the prediction horizon should be taken over the same time span as information about consumption and prices are available. However, due to the computational requirements, this is currently not practical. In order to make computations involving prediction horizons of larger lengths approaches have to be investigated that somehow reduce the number of binary variables and possibly aggregate information regarding energy usage at prediction steps further away.

In this section we have considered energy control of an individual household, as a first step toward cooperative energy control of several interconnected households. The next step could consist of modeling interconnections between households, and developing a scheme that makes control agents of individual households obtain agreement on the values of the variables involved in modeling these interconnections. In the next section we go more into the issues involved in dealing with such interconnections.

3.4 Control of interconnected hybrid subnetworks

In the previous section we have assumed that the subnetworks, viz. the households, are independent of each other. In this section we do not make this assumption anymore, but instead allow for the subnetworks to be interconnected. Let therefore a transportation network be divided into n subnetworks. The subnetworks are interconnected as in Section 1.3.2, hence, typically the interconnections are physical links between subnetworks over which commodity flows from one subnetwork into another. Assume that each subnetwork has a control agent assigned to it. If the overall combined MPC control problem is convex, then the agents can use the multi-agent single-layer MPC approaches of Section 1.3.2.

[5] In order to compare the performance of the control for the two case studies a shrinking horizon [137] has been taken. In the shrinking horizon approach, initially the original prediction horizon N is taken, but as soon as predictions would go over the actual simulation time span, the prediction horizon will be reduced. If no shrinking horizon is taken, then comparing the performance for varying N based on the performance over 1 day is not fair, since the control agent using a larger N will at the end of the day already takes into account what will happen the next day, whereas the control agent using a smaller N will not consider this, since it optimizes over a shorter term. This will have an influence on the actions chosen at the end of the day and therefore on the performance.

In that case, the control agents locally determine in a number of iterations control actions that are overall optimal. However, when the subnetworks are hybrid systems and modeled with both continuous and discrete variables, then the overall control problem will not be convex. It is then the question what difficulties arise due to this nonconvexity, and how the approaches of Section 1.3.2 can be extended to give at best solutions that are close to or equal to overall optimal solutions, and at least solutions that are feasible solutions.

3.4.1 Hybrid subnetwork models

In Section 3.2 we have developed means to transform the dynamics of hybrid systems into linear mixed-integer equality and inequality constraints, i.e., mixed-logical dynamic models. Here, we consider a subclass of this type of models, namely those models for which the discrete dynamics are caused by inputs that can take on values from a discrete set only. In addition, we assume that all other variables, including the interconnecting variables between subnetworks, are continuous variables. Note that this type of models is an extension of the type of models considered in Chapter 2, since we now allow discrete inputs. An example of a situation in which the considered type of models appears in transportation networks is, e.g., in road traffic networks a situation in which local actions consist of discrete speed limit settings and interconnecting constraints between subnetworks are expressed in terms of continuously modeled car flows. In power networks an example of such a situation is, e.g., a situation in which local actions consist changing of power generation or consumption in discrete quantities and interconnecting constraints between subnetworks involve continuous amounts of power flowing between the subnetworks.

Remark 3.2 There are two different types of discrete inputs:

1. discrete inputs that have a direct meaning as a quantity since they are represented as numbers, typically taking on values from a set of integer or real numbers, e.g., $\{0, 0.2, \ldots, 1.0\}$;
2. discrete inputs that only have a symbolic meaning, taking on values from a set of symbolic values, e.g., $\{\text{red}, \text{yellow}, \text{green}\}$.

Although these are different types of discrete inputs, note that, however, the second class of discrete inputs can typically be transformed into the first class of inputs, and vice versa. □

Hence, assume that a network is divided into n subnetworks and that the dynamics of each subnetwork $i \in \{1, \ldots, n\}$ are given by a deterministic linear discrete-time time-invariant model, with noise-free outputs:

$$\begin{aligned} \mathbf{x}_i(k+1) &= \mathbf{A}_i \mathbf{x}_i(k) + \mathbf{B}_{1,i} \mathbf{u}_i(k) + \mathbf{B}_{2,i} \mathbf{d}_i(k) + \mathbf{B}_{3,i} \mathbf{v}_i(k) \\ \mathbf{y}_i(k) &= \mathbf{C}_i \mathbf{x}_i(k) + \mathbf{D}_{1,i} \mathbf{u}_i(k) + \mathbf{D}_{2,i} \mathbf{d}_i(k) + \mathbf{D}_{3,i} \mathbf{v}_i(k), \end{aligned} \tag{3.14}$$

where at time step k, for subnetwork i, $\mathbf{x}_i(k) \in \mathbb{R}^{n_{\mathbf{x}_i}}$ are local states, $\mathbf{u}_i(k) \in \mathcal{U}_i$ (with $\mathcal{U}_i$ a finite set of discrete values) are local inputs, $\mathbf{d}_i(k) \in \mathbb{R}^{n_{\mathbf{d}_i}}$ are known local exogenous inputs, $\mathbf{y}_i(k) \in \mathbb{R}^{n_{\mathbf{y}_i}}$ are local outputs, $\mathbf{v}_i(k) \in \mathbb{R}^{n_{\mathbf{v}_i}}$ are remaining variables influencing the local dynamical states and outputs, e.g., variables of neighboring subnetworks, and $\mathbf{A}_i \in \mathbb{R}^{n_{\mathbf{x}_i} \times n_{\mathbf{x}_i}}$, $\mathbf{B}_{1,i} \in \mathbb{R}^{n_{\mathbf{x}_i} \times n_{\mathbf{u}_i}}$, $\mathbf{B}_{2,i} \in \mathbb{R}^{n_{\mathbf{x}_i} \times n_{\mathbf{d}_i}}$, $\mathbf{B}_{3,i} \in \mathbb{R}^{n_{\mathbf{x}_i} \times n_{\mathbf{v}_i}}$, $\mathbf{C}_i \in \mathbb{R}^{n_{\mathbf{y}_i} \times n_{\mathbf{x}_i}}$, $\mathbf{D}_{1,i} \in \mathbb{R}^{n_{\mathbf{y}_i} \times n_{\mathbf{u}_i}}$, $\mathbf{D}_{2,i} \in \mathbb{R}^{n_{\mathbf{y}_i} \times n_{\mathbf{d}_i}}$, and $\mathbf{D}_{3,i} \in \mathbb{R}^{n_{\mathbf{y}_i} \times n_{\mathbf{v}_i}}$ determine how the different variables influence the local state and output of subnetwork i.

3.4.2 Non-convergence due to the discrete inputs

Suppose that we would setup the MPC control problems as in Section 1.3.2, but now based on the model including discrete inputs, i.e., (3.14). This means that the optimization problems become mixed-integer programming problems. In addition, for fixed values of the integer variables, the optimization problems are convex.

If we would use the schemes of Section 1.3.2 to solve the multi-agent control problem based on the models including the discrete inputs, it may be the case that the agents cannot come to agreement on the values of the interconnecting variables, while choosing locally optimal discrete inputs. A non-converging sequence can arise of values of the interconnecting variables on which the agents do not reach agreement.

In the original approaches, i.e., the serial and the parallel multi-agent single-layer MPC schemes with convex overall MPC problems, a control agent i receives the information from each neighboring agent $j \in \mathcal{N}_i$ regarding the values that neighboring agent j would like the interconnecting variables with respect to agent i to have. Then, control agent i processes this information by updating its interconnecting objective function $J_{\mathrm{inter},i}$, and determines which values for the discrete inputs and interconnecting variables it prefers itself.

In the continuous case, the new values for the inputs and interconnecting variables will usually be slightly different from the values communicated in earlier iterations. However, when the inputs are discrete, the values for the *inputs* cannot slightly change, but only in discrete jumps. Hence, when a neighboring agent j suggests slightly different values for the *interconnecting* variables, control agent i will first include these values in its interconnecting objective function. After control agent i has solved its optimization problem using these new values, it will typically have obtained slightly changed values for the interconnecting variables, while having obtained values for the discrete inputs that are the same as the values at the previous iteration. So, the values of the discrete inputs will typically not change at each iteration, but only when the interconnecting objective function $J_{\mathrm{inter},i}$ has reached such a level that switching to different discrete inputs is beneficial.

The relatively large jumps in the values of the discrete inputs have as a consequence that the values for the interconnecting variables can significantly change as well. A control agent will therefore then suggest rather different values for the interconnecting variables to its neighboring agents. This may cause that for another control agent after some more iterations a certain threshold of the interconnecting objective function has been reached, making it better for that agent to switch the values of its discrete inputs. Due to this mechanism, a series of discrete jumps in the values of discrete inputs can emerge that prevents the iterations from terminating. We will see an example of this behavior in Section 3.5.

3.4.3 Possible extensions of the original schemes

There are several ways in which the original schemes of Section 1.3.2 could be extended in order to break such a series of non-converging discrete jumps. Below we discuss some of these alternatives, based on straightforward extensions of the original schemes. We consider the following extensions:

1. Increasing the accuracy threshold The accuracy threshold $\gamma_{\epsilon,\mathrm{term}}$ is used in the stopping condition to determine when the iterations should stop. It is linked to the maximum allowable violation of the interconnecting constraints. Therefore, if this threshold is increased, the iterations will stop sooner since the values of interconnecting variables involved in an interconnecting constraint are allowed further apart from each other. However, this can obviously lead to predictions of the subnetwork that do not reflect the evolution of the physical subnetwork, and therefore to sub-optimally chosen inputs. In addition, it is a priori unknown to which value the accuracy threshold should be increased. If the increase is not large enough, the iterations may still continue.

2. Refining the discretization By making the discretization of the discrete inputs finer over the iterations, at some point the discretization will be fine enough to let the iterations converge to values for the interconnecting variables that make the stopping condition satisfied. By making a finer discretization for the discrete inputs, the changes in the discrete inputs from one iteration to another will be smaller, hence, approximating the case when there are only continuous inputs. In practice, however, the discretization of the discrete inputs may be given, and may not be adjustable. In that case the finer discretization can be rounded to the closest original discrete value. However, rounding of values has some consequences, as discussed in the next approach.

3. Relaxing and rounding The extreme case of refinement of the discretization appears when the discrete inputs are relaxed to continuous inputs, as is done, e.g., in [15]. In this case, the original schemes can be applied. At termination of the iterations, the resulting values for the continuous inputs can then be rounded to the closest discrete values for the discrete inputs. However, in particular when making predictions over a longer horizon this rounding can lead at least to sub-optimality and sometimes even to infeasibility. This is due to the fact that in general a rounded input has a different influence on the evolution of the subnetwork over a time step when compared to the influence that a continuous input would have. So in practice the evolution of the subnetwork will be different from the predictions made using the prediction model in the optimization.

4. Fixing the integer inputs The discrete inputs can be fixed once the non-converging series of values of the discrete inputs has been detected. The discrete inputs can be fixed to the locally most optimal values, or they can be fixed to the most frequently appearing values over a predefined number of earlier iterations. The remaining overall optimization problem will then become convex and the values of the other variables will converge to values that are optimal given the fixed discrete variables. In addition, at the end of the iterations the interconnecting constraints will be satisfied and thus the agents will have agreed on how the internetwork variables should evolve over the prediction horizon. Furthermore, the agents will have determined inputs that are feasible, and the agreements regarding the values for the interconnecting constraints will be fulfilled when these inputs are implemented. However, the fixed discrete variables may be sub-optimal from a network-wide perspective, and determining when the non-converging series of discrete values arises is a hard problem.

5. Increasing the penalty coefficient The penalty coefficient γ_c can be increased to a very high value once the non-converging series of discrete values has been detected. A large value for the penalty coefficient γ_c places all emphasis on obtaining satisfied interconnecting constraints, and the discrete inputs that come with this can then be implemented. However, it may not be known a priori what the value of the penalty coefficient γ_c should have in order to give convergence and in addition it is hard to determine when the non-converging series of discrete values appears. Therefore, inspired by [20], instead of increasing the penalty coefficient γ_c abruptly when the non-converging series of discrete values has been detected, the penalty coefficient γ_c can be increased in steps several times over the iterations. By increasing the penalty coefficient γ_c in steps, the agents get some time to try to converge to values for the interconnecting variables that satisfy the stopping condition. If this convergence does not happen within a certain number of iterations, then the penalty coefficient γ_c is increased again.

Discussion Comparing the alternatives, the main disadvantage of the alternatives based on increasing the accuracy threshold, and relaxing or refining of the discretization and then rounding, is that the values for the interconnecting variables observed in the system will be significantly different from those determined during the optimization. For the alternative based on increasing the accuracy threshold the reason for this disadvantage is that during the optimization the accuracy required on satisfying the interconnecting constraints is lowered, and thus the values that different control agents assign to particular interconnecting variables are allowed to be further apart. For the alternative based on relaxing or refining of the discretization and then rounding, the reason for this disadvantage is that the control agents have reached agreement on values for the interconnecting variables for a particular set of inputs, whereas a different set of inputs will be implemented on the system. The alternatives based on fixing the integer inputs and increasing the penalty coefficient do not suffer from this disadvantage.

The alternative based on fixing the integer inputs requires that it can be detected when the integer inputs have to be fixed and it requires a strategy to determine to which values the inputs should be fixed. It is not straightforward to implement such strategies. The alternative based on increasing the penalty coefficient does not have to address these issues. However, for this alternative it has to be determined at which frequency the penalty coefficient should be increased, and with which factor. The settings that give the best performance will be problem specific and therefore require tuning. From the alternatives discussed, this last alternative has the most natural way of dealing with the non-converging behavior, by emphasizing over the iterations more and more that a solution should be obtained with interconnecting constraints that are satisfied. The predictions that each control agent therefore makes of its subnetwork are accurate at termination of the iterations.

Below we use the alternative based on increasing the penalty coefficient to formulate a multi-agent single-layer MPC approach for interconnected hybrid systems.

3.4.4 Serial and parallel single-layer hybrid MPC approaches

For control of interconnected hybrid systems, in which the subnetworks are linear time-invariant systems with discrete inputs as modeled using (3.14), and the MPC overall control problem is convex for fixed values of the integer variables, we propose the serial and parallel scheme of Section 1.3.2, with the extension that the penalty coefficient γ_c varies over the iterations, i.e., extension 5 above. Hence, the original serial and parallel scheme are followed in the sense that the agents perform local optimization steps and communication, but the way in which the information from neighboring agents is included in updating the interconnecting objective function is different. Instead of using a fixed penalty coefficient, an iteration-varying penalty coefficient is taken. Every $N_{\Delta s}$ iterations, the penalty coefficient $\gamma_{\rm c}$ is multiplied by $\gamma_{\Delta\rm c}$, with $\gamma_{\Delta\rm c} > 1$.

Remark 3.3 The approaches proposed in this section for multi-agent MPC control of the assumed class of systems follow from rather straightforward extensions of the original approaches of Section 1.3.2. More complex extensions could be the result of combining the original approaches of Section 1.3.2 with optimization techniques for integer programming, such as distributed branch and bound or ADOPT [101]. In an iterative way of alternating between the distributed branch and bound and the approaches of Section 1.3.2, the distributed branch and bound approach could determine values for the integer variables, after which the integer values can be fixed, and the resulting convex overall problem can be solved using the approaches of Section 1.3.2. Such an approach could potentially address a larger class of systems than assumed here, although that remains to be investigated. □

In the following section we perform experiments with the proposed scheme on a load-frequency control problem with discrete power generation.

3.5 Application: Discrete-input load-frequency control

We consider the load-frequency control problem as defined in Section 2.5. In this load-frequency control problem a power network is divided into n subnetworks, each equipped with power generation and consumption capabilities. A control agent is assigned to each subnetwork. The objective of each control agent is to keep frequency deviations at a minimum after load disturbances. In order to achieve this objective each control agent can adjust the power generation in its subnetwork. In the original problem definition of Section 2.5, power generation was considered as a continuous input. Here, we assume that power generation can be adjusted in discrete amounts, hence, power generation is considered as a discrete input. Such discrete power generation is present, e.g., if generators can be switched on or off, or if actuators on the generator can take on values only from a discrete set of values. Furthermore, also load shedding, which can be seen in a way as negative power generation, is typically done in discrete amounts.

3.5.1 Network setup

For illustrative purposes, we consider a network consisting of 2 subnetworks, as shown in Figure 3.11. The dynamics and parameter of the subnetworks are as described in Section 2.5, with the exception that the inputs can only take on discrete values from the set $\{-1.0, -0.9, \ldots, 0.9, 1.0\}$.

3.5.2 Control setup

The control agents controlling subnetworks 1 and 2 use the objective function as defined in Section 2.5. The mixed-integer optimization problem that each control agent solves at an iteration is solved using the quadratic mixed-integer solver of ILOG CPLEX v10 [71], which we use through the Tomlab v5.7 [66] interface in Matlab v7.3 [98].

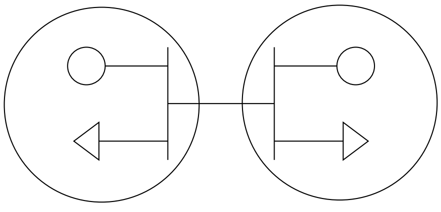


Figure 3.11: Network consisting of 2 subnetworks. Each subnetwork has generation and consumption capabilities.

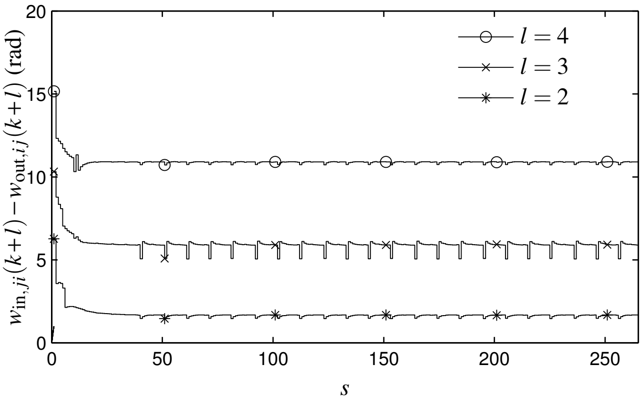


Figure 3.12: Results without *using the extended version of the serial MPC scheme. Per iteration the values of the interconnecting input variable of subnetwork 1 for prediction steps 2, 3, and 4 are shown. The values of the variable for the prediction steps 2, 3, and 4 are shifted with +5, +10, and +15, respectively.*

3.5.3 Simulations

To show the non-converging series of discrete values of the inputs, consider the experiment in which we take a prediction horizon with length $N = 5$ steps, an accuracy threshold $\gamma_{\epsilon,\mathrm{term}} = 0.0001$, and an initial penalty coefficient $\gamma_{\mathrm{c}}(0)$ of 1. The penalty coefficient $\gamma_{\mathrm{c}}(s)$ is updated every $N_{\Delta s} = 50$ iterations, with a factor of $\gamma_{\Delta \mathrm{c}} = 1.5$. The initial state of the network is $x_{\Delta f,1}(0) = 0$, $x_{\Delta\delta,1}(0) = 0$, $x_{\Delta f,2}(0) = 0$, and $x_{\Delta\delta,2}(0) = -1.0745$.

3.5.4 Results

When the control agents do not use the adjustment of the penalty term γ_{c}, then the non-converging series of discrete values appears, as illustrated in Figures 3.12 and 3.13 for the serial approach. The figures illustrate that as the control agents exchange information, the value of the interconnecting input of control agent 1 changes, also when the values of the discrete inputs do not change. At the moments that the discrete inputs change, a clear jump is also observed in the value of the interconnecting input.

When the control agents use the penalty term increments extension, then the iterations converge, as illustrated in Figures 3.14, 3.15, and 3.16. It can be seen that in this case as the penalty coefficient $\gamma_{\mathrm{c}}(s)$ increases, the number of jumps in the discrete inputs reduces, and

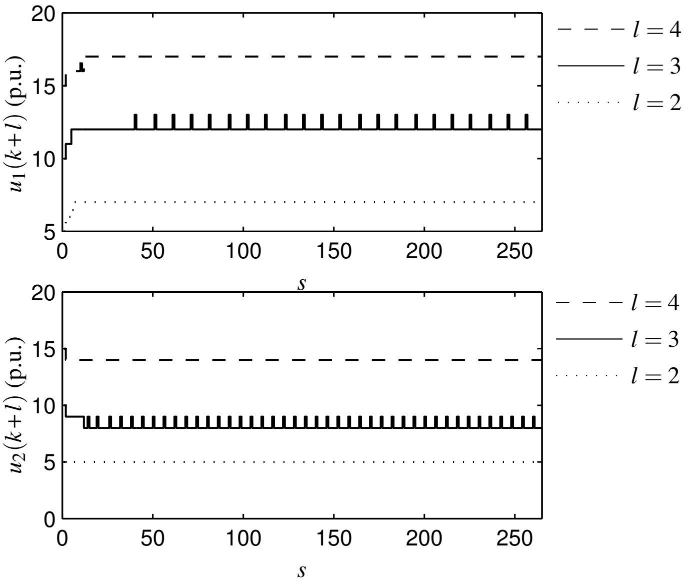


Figure 3.13: Results without *using the extended version of the serial MPC scheme. The values of the discrete inputs chosen by the agents of subnetworks 1 (top) and 2 (bottom), respectively, for prediction steps 2, 3, and 4 are shown. The values of the inputs for the prediction steps 2, 3, and 4 are shifted with +5, +10, and +15, respectively. In addition, the values of the inputs are scaled (before shifting) to take on integer values between -10 and 10.*

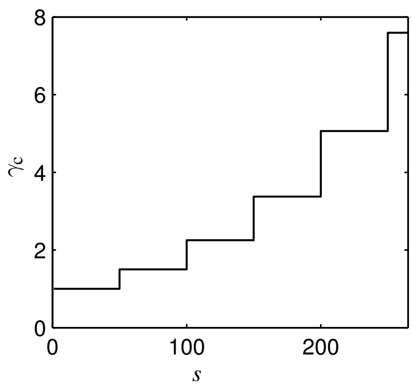


Figure 3.14: Evolution of penalty coefficient γ_c using the extended version of the serial MPC scheme.

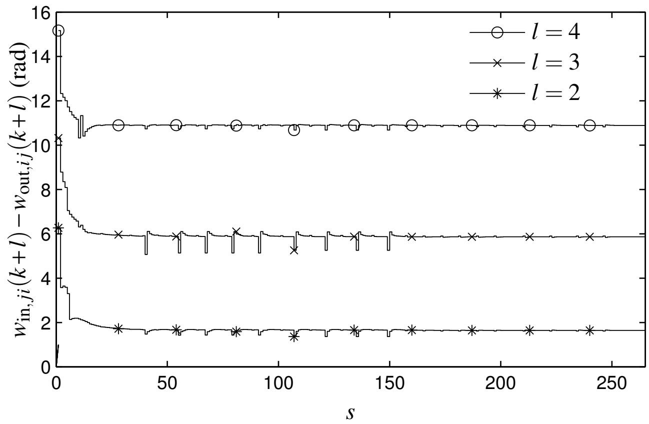


Figure 3.15: Interconnecting variable resulting from using the extended version of the serial MPC scheme. The values of the variable for the prediction steps 2, 3, and 4 are shifted with +5, +10, and +15, respectively.

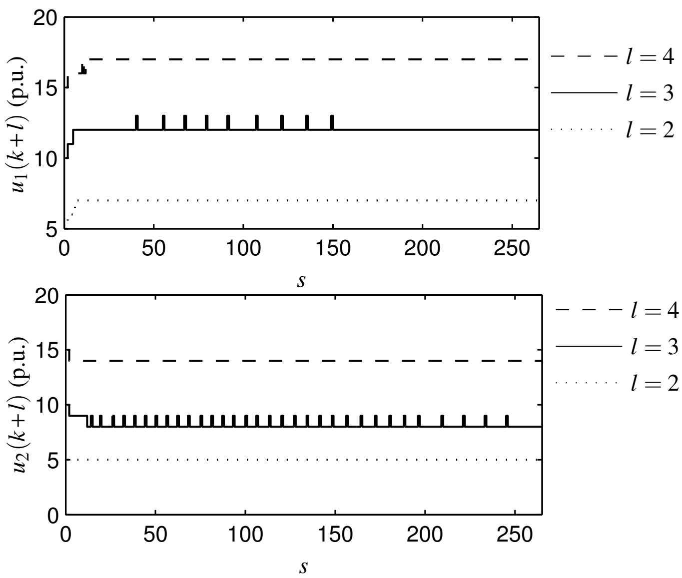


Figure 3.16: Inputs resulting from using the extended version of the serial MPC scheme. The values of the inputs for the prediction steps 2, 3, and 4 are shifted with +5, +10, and +15, respectively. In addition, the values of the inputs are scaled (before shifting) to take on integer values between -10 and 10.

that ultimately convergence is obtained. It is worth noting that the inputs that are chosen by the control agents are the same as those that would have been chosen by a centralized overall control agent.

3.6 Summary

In this chapter we have discussed multi-agent MPC control of transportation networks modeled as interconnected hybrid systems. In this setting, the network is divided into a number of subnetworks, each being controlled by a control agent that uses a model of its subnetwork and MPC to determine its actions.

We have first focused on modeling of hybrid systems and discussed how logic statements, which commonly appear in the description of hybrid systems, can be transformed into linear mixed-integer equality and inequality constraints. Then, we have illustrated the use of the transformations to construct a prediction model for an a single MPC control agent. Subsequently, we have focused on multi-agent control of networks consisting of subnetworks that are modeled as hybrid systems. We have focused on a particular type of hybrid subnetworks, viz. subnetworks with linear time-invariant dynamics that accept inputs that take on values from a discrete set of values only. Furthermore, we have discussed the problems that arise when the serial and parallel scheme of Chapter 2 would be applied to this type of system without modification. Moreover, we have discussed several alternative extensions of the original schemes to deal with these problems, and we have chosen one extension that results in control agents choosing feasible integer inputs, based on accurate subnetwork predictions. Several issues still have to be addressed in future research, including among others investigating formally whether the proposed scheme converges, determining formally what the quality of the solutions is, and determining when the penalty coefficient should be increased and with what value it should be increased. In addition, how to combine distributed optimization problem solvers for continuous and integer variables should be investigated.

In this chapter we have applied the topics discussed on two applications: energy control in households, and load-frequency control with discrete generation switching. For the energy control in households application we have used the transformations to derive a model for a household equipped with its own power generation (via a micro combined heat and power unit) and storage capabilities (via a water tank and a battery). As a first step toward a control structure in which multiple control agents, each representing a single household, jointly control the energy usage in a district, we have proposed a decentralized multi-agent single-layer MPC approach in which the control agents only consider their own household and no communication with other control agents takes place. In the application of the load-frequency control with discrete generation switching we have considered how the proposed extension of the serial scheme of Chapter 2 performs when the subnetworks do have interconnections, and the respective control agents do communicate with one another. We have illustrated that the extension proposed for dealing with non-convergence of the iterations of the MPC scheme can make the iterations converge.

In this chapter, as well as in Chapter 2, we have focused on issues particular to single-layer control, i.e., control in which the control agents have equal authority relationships with respect to one another. In Chapters 4 and 5 we focus more on how to take into account also control agents with different authority relationships.

Bibliography

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

[2] M. Aicardi, G. Casalino, R. Minciardi, and R. Zoppoli. 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, November 1992.

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

[4] P.J. Antsaklis and A. Nerode, editors. Special issue on hybrid systems. *IEEE Transactions on Automatic Control*, 43(4), April 1998.

[5] Power Systems Test Case Archive. Parameters of the IEEE 57-bus grid. http://www.ee.washington.edu/research/pstca/.

[6] K. J. Åström and B. Wittenmark. *Computer-Controlled Systems*. Prentice-Hall, Upper Saddle River, New Jersey, 1997.

[7] K. J. Åström, H. Elmqvist, and S. E. Mattsson. Evolution of continuous-time modeling and simulation. In *Proceedings of the 12th European Simulation Multiconference*, pages 9–18, Manchester, UK, June 1998.

[8] N. Atic, D. Rerkpreedapong, A. Hasanovic, and A. Feliachi. NERC compliant decentralized load frequency control design using model predictive control. In *Proceedings on the IEEE Power Engineering Society General Meeting*, Toronto, Canada, July 2003.

[9] C. Audet and J. E. Dennis Jr. Pattern search algorithms for mixed variable programming. *SIAM Journal on Optimization*, 11(3):573–594, 2000.

[10] C. Audet and J. E. Dennis Jr. Analysis of generalized pattern searches. *SIAM Journal on Optimization*, 13(3):889–903, 2007.

[11] T. Başar and G. J. Olsder. *Dynamic Non-Cooperative Game Theory*. Academic Press, London, UK, 1998.

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

[13] P. Barton and C. Pantelides. Modeling of combined discrete/continuous processes. *AIChE Journal*, 40(6):966–979, 1994.

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

[15] A. G. Beccuti and M. Morari. A distributed solution approach to centralized emergency voltage control. In *Proceedings of the 2006 IEEE American Control Conference*, pages 3445–3450, Minneapolis, Minnesota, June 2006.

[16] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–427, March 1999.

[17] J. Bernussou and A. Titli. *Interconnected Dynamical Systems: Stability, Decomposition and Decentralisation*. North-Holland Publishing Company, Amsterdam, The Netherlands, 1982.

[18] D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, Beltmore, Massachusetts, 2003.

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

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

[21] P. R. Bhave and R. Gupta. *Analysis of Water Distribution Networks*. Alpha Science International, Oxford, UK, 2006.

[22] L. G. Bleris, P. D. Vouzis, J. G. Garcia, M. G. Arnold, and M. V. Kothare. Pathways for optimization-based drug delivery. *Control Engineering Practice*, 15(10):1280–1291, October 2007.

[23] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, UK, 2004.

[24] S. D. Braithwait. Real-time pricing and demand response can work within limits. *Natural Gas and Electricity*, 21(11):1–9, 2005.

[25] M. W. Braun, D. E. Rivera, M. E. Flores, W. M. Carlyle, and K. G. Kempf. A model predictive control framework for robust management of multi-product, multi-echelon demand networks. *Annual Reviews in Control*, 27:229–245, 2003.

[26] K. E. Brenan, S. L. Campbell, and L. R. Petzold. *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*. SIAM, Philadelphia, Pennsylvania, 1996.

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

[28] E. Camponogara, D. Jia, B. H. Krogh, and S. Talukdar. Distributed model predictive control. *IEEE Control Systems Magazine*, 1:44–52, February 2002.

[29] C. G. Cassandras and S. Lafortune. *Introduction to Discrete Event Systems*. Kluwer Academic Publishers, Boston, Massachusetts, 1999.

[30] C. G. Cassandras, S. Lafortune, and G. J. Olsder. Introduction to the modelling, control and optimization of discrete event systems. In A. Isidori, editor, *Trends in Control: A European Perspective*, pages 217–291. Springer-Verlag, Berlin, Germany, 1995.

[31] A. J. Conejo, F. J. Nogales, and F. J. Prieto. A decomposition procedure based on approximate newton directions. *Mathematical Programming, Series A*, 93(3):495–515, December 2002.

[32] A. R. Conn, K. Scheinberg, and P. L. Toint. Recent progress in unconstrained nonlinear optimization without derivatives. *Mathematical Programming*, 79(1–3):397–414, 1997.

[33] C. F. Daganzo. *Fundamentals of Transportation and Traffic Operations*. Pergamon Press, New York, New York, 1997.

[34] R. David. Modeling of dynamic systems by Petri nets. In *Proceedings of the 1st European Control Conference*, pages 136–147, Grenoble, France, July 1991.

[35] B. De Schutter and T. J. J. van den Boom. Model predictive control for max-min-plus-scaling systems. In *Proceedings of the 2001 American Control Conference*, pages 319–324, Arlington, Virginia, June 2001.

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

[37] W. B. Dunbar and R. M. Murray. Model predictive control of coordinated multi-vehicle formations. In *Proceedings of the 41st IEEE Conference on Decision and Control*, pages 4631–4636, Las Vegas, Nevada, December 2002.

[38] W. B. Dunbar and R. M. Murray. Distributed receding horizon control for multi-vehicle formation stabilization. *Automatica*, 42(4):549–558, April 2006.

[39] Dynasim. Dymola – User's Manual. Technical report, Dynasim AB, Lund, Sweden, 2004.

[40] A. Edris, R. Adapa, M. H. Baker, L. Bohmann, K. Clark, K. Habashi, L. Gyugyi, J. Lemay, A. S. Mehraban, A. K. Meyers, J. Reeve, F. Sener, D. R. Torgerson, and R. R. Wood. Proposed terms and definitions for flexible AC transmission system (FACTS). *IEEE Transactions on Power Delivery*, 12(4):1848–1853, October 1997.

[41] H. El Fawal, D. Georges, and G. Bornard. 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*, pages 3874–3879, San Diego, California, 1998.

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

[43] Elkraft Systems. Power failure in Eastern Denmark and Southern Sweden on 23 September 2003 – preliminary report on the course of events. Technical report, Elkraft Systems, Holte, Denmark, 2003.

[44] H. Elmqvist, F. E. Cellier, and M. Otter. Object-oriented modeling of hybrid systems. In *Proceedings of the European Simulation Symposium*, pages xxxi–xli, Delft, The Netherlands, October 1998.

[45] B. Fardanesh. Future trends in power system control. *IEEE Computer Applications in Power*, 15(3):24–31, July 2002.

[46] R. G. Farmer and P. M. Anderson. *Series Compensation of Power Systems*. PBLSH, Encinitas, California, 1996.

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

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

[49] T. Geyer, M. Larsson, and M. Morari. Hybrid emergency voltage control in power systems. In *Proceedings of the European Control Conference 2003*, Cambridge, UK, September 2003.

[50] P. E. Gill, W. Murray, and M. A. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM Journal on Optimisation*, 12(4):979–1006, 2002.

[51] G. Glanzmann and G. Andersson. FACTS control for large power systems incorporating security aspects. In *Proceedings of X SEPOPE*, Florianopolis, Brazil, May 2006.

[52] G. Glanzmann and G. Andersson. Using FACTS devices to resolve congestions in transmission grids. In *Proceedings of the CIGRE/IEEE PES International Symposium*, San Antonio, Texas, October 2005.

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

[54] A. H. González, D. Odloak, and J. L. Marchetti. Predictive control applied to heat-exchanger networks. *Chemical Engineering and Processing*, 45(8):661–671, August 2006.

[55] E. González-Romera, M. Á. Jaramillo-Morán, and D. Carmona-Fernández. Forecasting of the electric energy demand trend and monthly fluctuation with neural networks. *Computers and Industrial Engineering*, 52:336–343, April 2007.

[56] G. C. Goodwin, M. M. Seron, R. H. Middleton, M. Zhang, B. F. Hennessy, P. M. Stone, and M. Menabde. Receding horizon control applied to optimal mine planning. *Automatica*, 42(8):1337–1342, August 2006.

[57] W. P. M. H. Heemels, B. De Schutter, and A. Bemporad. Equivalence of hybrid dynamical models. *Automatica*, 37(7):1085–1091, July 2001.

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

[59] D. J. Hill. Nonlinear dynamic load models with recovery for voltage stability studies. *IEEE Transactions on Power Systems*, 8(1):166–176, February 1993.

[60] D. J. Hill, Y. Guo, M. Larsson, and Y. Wang. Global control of complex power systems. In G. Chen, D. J. Hill, and X. Yu, editors, *Bifurcation Control: Theory and Applications*, Lecture Notes in Control and Information Sciences, pages 155–187. Springer, Berlin, Germany, 2003.

[61] P. Hines, L. Huaiwei, D. Jia, and S. Talukdar. 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*, pages 273–278, Tucson, Arizona, March 2005.

[62] N. G. Hingorani and L. Gyugyi. *Understanding FACTS concepts and technology of flexible AC transmission systems*. IEEE Press, New York, New York, 2000.

[63] I. A. Hiskens and K. Mitsumoto. Dynamical systems benchmark library. URL: http://psdyn.ece.wisc.edu/IEEE_benchmarks/, 2005.

[64] Y. C. Ho, P. B. Luh, and G. J. Olsder. A control-theoretic view on incentives. In *Proceedings of the 19th IEEE Conference on Decision and Control*, pages 1160–1170, Albuquergue, New Mexico, December 1980.

[65] K. Holmström, A. O. Göran, and M. M. Edvall. User's guide for Tomlab /SNOPT, December 2006.

[66] K. Holmström, A. O. Göran, and M. M. Edvall. User's guide for Tomlab /CPLEX, June 2007.

[67] M. Houwing, A. N. Ajah, P. M. Herder, and I. Bouwmans. Addressing uncertainties in the design and operation of residential distributed energy resources: Case study

of a micro-CHP system. In *Proceedings of the 10th Conference on Process Integration, Modelling and Optimisation for Energy Saving and Pollution Reduction*, Ischia Island, Italy, June 2007.

[68] M. Houwing, R. R. Negenborn, P. Heijnen, B. De Schutter, and J. Hellendoorn. Least-cost model predictive control of residential energy resources when applying μCHP. In *Proceedings of Power Tech 2007*, Lausanne, Switzerland, July 2007. Paper 291.

[69] G. Hug-Glanzmann, R. R. Negenborn, G. Andersson, B. De Schutter, and J. Hellendoorn. Multi-area control of overlapping areas in power systems for FACTS control. In *Proceedings of Power Tech 2007*, Lausanne, Switzerland, July 2007. Paper 277.

[70] Ibraheem, P. Kumar, and D. P. Kothari. Recent philosophies of automatic generation control strategies in power systems. *IEEE Transactions on Power Systems*, 20(1): 346–357, February 2005.

[71] ILOG. CPLEX. URL: http://www.ilog.com/products/cplex/, 2007.

[72] R. Irizarry-Rivera and W. D. Seider. Model-predictive control of the Czochralski crystallization process. Part I. Conduction-dominated melt. *Journal of Crystal Growth*, 178(4):593–611, 1997.

[73] N. Jenkins, R. Allan, P. Crossley, D. Kirschen, and G. Strbac. *Embedded Generation*. TJ International, Padstow, UK, 2000.

[74] D. Jia and B. Krogh. Min-max feedback model predictive control for distributed control with communication. In *Proceedings of the 2002 American Control Conference*, pages 4507–4512, Anchorage, Alaska, May 2002.

[75] D. Jia and B. H. Krogh. Distributed model predictive control. In *Proceedings of the 2001 American Control Conference*, pages 2767–2772, Arlington, Virginia, June 2001.

[76] D. Karlsson and D. J. Hill. Modelling and identification of nonlinear dynamic loads in power systems. *IEEE Transactions on Power Systems*, 9(1):157–163, February 1994.

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

[78] H. Kawabata and M. Kido. A decentralized scheme of load frequency control power system. *Electrical Engineering Japan*, 102(4):100–106, July 1982.

[79] T. Keviczky, F. Borrelli, and G. J. Balas. A study on decentralized receding horizon control for decoupled systems. In *Proceedings of the 2004 American Control Conference*, volume 6, pages 4921–4926, Boston, Massachusetts, June 2004.

[80] B. H. Kim and R. Baldick. A comparison of distributed optimal power flow algorithms. *IEEE Transactions on Power Systems*, 15(2):599–604, May 2000.

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

[82] P. Kundur. *Power System Stability and Control*. McGraw-Hill, New York, New York, 1994.

[83] S. Leirens, J. Buisson, P. Bastard, and J.-L. Coullon. A hybrid approach for voltage stability of power systems. In *Proceedings of the 15th Power Systems Computation Conference*, Liège, Belgium, August 2005. Paper 291.

[84] R. M. Lewis and V. Torczon. Pattern search methods for linearly constrained minimization. *SIAM Journal on Optimization*, 10(3):917–941, 2000.

[85] R. M. Lewis and V. Torczon. A globally convergent augmented Lagrangian pattern search algorithm for optimization with constraints and simple bounds. *SIAM Journal on Optimization*, 12(4):1075–1089, 2002.

[86] R. M. Lewis and V. Torczon. Pattern search algorithms for bound constrained minimization. *SIAM Journal on Optimization*, 9(4):1082–1099, 1999.

[87] R. M. Lewis, V. Torczon, and M. W. Trosset. Direct search methods: then and now. *Journal of Computational and Applied Mathematics*, 124(1–2):191–207, December 2000.

[88] G. Lodewijks. *Dynamics of Belt Systems*. PhD thesis, Delft University of Technology, The Netherlands, 1996.

[89] Z. Lukszo, M. P. C. Weijnen, R. R. Negenborn, B. De Schutter, and M. Ilić. Challenges for process system engineering in infrastructure operation and control. In W. Marquardt and C. Pantelides, editors, *16th European Symposium on Computer Aided Process Engineering and 9th International Symposium on Process Systems Engineering* (Garmisch-Partenkirchen, Germany, July 2006), volume 21 of *Computer-Aided Chemical Engineering*, pages 95–100. Elsevier, Amsterdam, The Netherlands, 2006.

[90] Z. Lukszo, M. P. C. Weijnen, R. R. Negenborn, and B. De Schutter. Tackling challenges in infrastructure operation and control using multi-level and multi-agent control. Technical Report 07-027, Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands, 2007. Submitted to a journal.

[91] E. Lysen, S. Van Egmond, and S. Hagedoorn. Opslag van elektriciteit: Status en toekomstperspectief voor Nederland. Technical Report NEO 0268-05-05-01-002, Utrecht Centrum voor Energieonderzoek – SenterNovem, Utrecht, The Netherlands, 2006. In Dutch.

[92] J. Machowski, J. Bialek, and J. R. Bumby. *Power System Dynamics and Stability*. John Wiley & Sons, New York, New York, 1997.

[93] J. M. Maciejowski. *Predictive Control with Constraints*. Prentice-Hall, Harlow, UK, 2002.

[94] Y. Majanne. Model predictive pressure control of steam networks. *Control Engineering Practice*, 13(12):1499–1505, December 2005.

[95] A. Manzoni, A. S. de Silva, and I. C. Decker. Power systems, dynamics simulation using object-oriented programming. *IEEE Transactions on Power Systems*, 14(1): 249–255, 1999.

[96] R. Martí. Multi-Start Methods. In F. Glover and G. A. Kochenberger, editors, *Handbook of Metaheuristics*, chapter 12, pages 355–368. Springer, New York, New York, 2006.

[97] MathWorks. Genetic Algorithm and Direct Search Toolbox 2 – User's Guide, 2007.

[98] Mathworks. Matlab. URL: http://www.mathworks.com/, 2007.

[99] S. E. Mattsson, H. Elmqvist, and M. Otter. Physical system modeling with Modelica. *Control Engineering Practice*, 6(4):501–510, April 1998.

[100] M. D. Mesarovic, D. Macko, and Y. Takahara. *Theory of Hierarchical Multilevel Systems*. Academic Press, New York, New York, 1970.

[101] P. J. Modi, W. M. Shen, M. Tambe, and M. Yokoo. ADOPT: Asynchronous Distributed Constraint Optimization with quality guarantees. *Artificial Intelligence*, 161 (1-2):149–180, January 2005.

[102] M. Morari and J. H. Lee. Model predictive control: past, present and future. *Computers and Chemical Engineering*, 23(4):667–682, 1999.

[103] J. J. Moré and S. J. Wright. *Optimization Software Guide*. SIAM, Philadelphia, Pennsylvania, 1993.

[104] A. S. Morse, C. C. Pantelides, S. Sastry, and J. M. Schumacher, editors. Special issue on hybrid systems. *Automatica*, 35(3), March 1999.

[105] I. R. Navarro, M. Larsson, and G. Olsson. Object-oriented modeling and simulation of power systems using modelica. In *Proceedings of the IEEE Power Engineering Society Winter Meeting*, pages 790–795, Singapore, January 2000.

[106] R. R. Negenborn, B. De Schutter, M.A. Wiering, and H. Hellendoorn. Learning-based model predictive control for Markov decision processes. In *Proceedings of the 16th IFAC World Congress*, Prague, Czech Republic, July 2005. Paper 2106 / We-M16-TO/2.

[107] R. R. Negenborn, B. De Schutter, and H. Hellendoorn. Multi-agent model predictive control of transportation networks. In *Proceedings of the 2006 IEEE International Conference on Networking, Sensing and Control (ICNSC 2006)*, pages 296–301, Fort Lauderdale, Florida, April 2006.

[108] R. R. Negenborn, B. De Schutter, and H. Hellendoorn. Multi-agent model predictive control for transportation networks with continuous and discrete elements. In *Proceedings of the 11th IFAC Symposium on Control in Transportation Systems*, pages 609–614, Delft, The Netherlands, August 2006.

[109] R. R. Negenborn, B. De Schutter, and J. Hellendoorn. Multi-agent model predictive control for transportation networks: Serial versus parallel schemes. In *Proceedings of the 12th IFAC Symposium on Information Control Problems in Manufacturing (INCOM 2006)*, pages 339–344, Saint-Etienne, France, May 2006.

[110] R. R. Negenborn, A. G. Beccuti, T. Demiray, S. Leirens, G. Damm, B. De Schutter, and M. Morari. Supervisory hybrid model predictive control for voltage stability of power networks. In *Proceedings of the American Control Conference 2007*, pages 5444–5449, New York, New York, July 2007.

[111] R. R. Negenborn, B. De Schutter, and J. Hellendoorn. Efficient implementation of serial multi-agent model predictive control by parallelization. In *Proceedings of the 2007 IEEE International Conference on Networking, Sensing, and Control*, pages 175–180, London, UK, July 2007.

[112] R. R. Negenborn, B. De Schutter, and J. Hellendoorn. Multi-agent model predictive control for transportation networks: Serial versus parallel schemes. Technical Report 07-024, Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands, 2007. To appear in Engineering Applications of Artificial Intelligence.

[113] R. R. Negenborn, S. Leirens, B. De Schutter, and J. Hellendoorn. Supervisory nonlinear MPC for emergency voltage control using pattern search. Technical Report 07-025, Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands, 2007. Submitted to a journal.

[114] P. G. Neumann. Widespread network failures. *Communications of the ACM*, 50(2): 112, 2007.

[115] J. Nocedal and S. Wright. *Numerical Optimization*. Springer Series in Operations Research. Springer-Verlag, New York, 1999. ISBN 0-387-98793-2.

[116] F. J. Nogales, F. J. Prieto, and A. J. Conejo. Multi-area AC optimal power flow: A new decomposition approach. In *Proceedings of the 13th Power Systems Control Conference (PSCC)*, pages 1201–1206, Trondheim, Germany, 1999.

[117] S. Ochs, S. Engell, and A. Draeger. Decentralized vs. model predictive control of an industrial glass tube manufacturing process. In *Proceedings of the 1998 IEEE Conference on Control Applications*, pages 16–20, Trieste, Italy, 1998.

[118] M. Otter, H. Elmqvist, and S. E. Mattson. Hybrid modeling in Modelica based on the synchronous data flow principle. In *Proceedings of the International Symposium on Computer Aided Control System Design*, pages 151–157, Kohala Coast-Island of Hawai'i, Hawai'i, August 1999.

[119] Y. M. Park and K. Y. Lee. Optimal decentralized load frequency control. *Electrical Power Systems Research*, 7(4):279–288, September 1984.

[120] M. Pehnt, M. Cames, C. Fischer, B. Praetorius, L. Schneider, K. Schumacher, and J. Vob. *Micro Cogeneration: Towards Decentralized Energy Systems*. Springer, Berlin, Germany, 2006.

[121] L. R. Petzold. A description of DASSL - A differential/algebraic system solver. In *Proceedings of the 10th World Congress on System Simulation and Scientific Computation*, pages 430–432, Montreal, Canada, August 1983.

[122] P. C. Piela, T. G. Epperly, K. M. Westerberg, and A. W. Westerberg. ASCEND: an object-oriented computer environment for modeling and analysis: The modeling language. *Computers and Chemical Engineering*, 15(1):53–72, January 1991.

[123] S. Piñón, E. F. Camacho, B. Kuchen, and M. Peña. Constrained predictive control of a greenhouse. *Computers and Electronics in Agriculture*, 49(3):317–329, December 2005.

[124] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes*. Cambridge University Press, Cambridge, UK, 2007.

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

[126] D. Rerkpreedapong, N. Atic, and A. Feliachi. Economy oriented model predictive load frequency control. In *Proceedings of the 2003 Large Engineering Systems Conference on Power Engineering*, pages 12–16, Montreal, Canada, May 2003.

[127] C. B. Royo. *Generalized Unit Commitment by the Radar Multiplier Method*. PhD thesis, Technical University of Catalonia, Barcelona, Spain, May 2001.

[128] P. W. Sauer and M. A. Pai. *Power System Dynamics and Stability*. Prentice-Hall, London, UK, 1998.

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

[130] N. I. Shaikh and V. Prabhu. Model predictive controller for cryogenic tunnel freezers. *Journal of Food Engineering*, 80(2):711–718, May 2007.

[131] M. G. Singh and A. Titli. *Systems Decomposition, Optimisation and Control*. Pergamon Press, Oxford, UK, 1978.

[132] J.-E. Skog, K. Koreman, and B. Pääjärvi. The Norned HVDC cable link – A power transmission highway between Norway and The Netherlands. In *Proceedings of Energex 2006*, Stavanger, Norway, June 2006.

[133] E. D. Sontag. Nonlinear regulation: The piecewise linear approach. *IEEE Transactions on Automatic Control*, 26(2):346–358, April 1981.

[134] K. Staňková, M. C. J. Bliemer, and G. J. Olsder. Inverse Stackelberg games and their application to dynamic bilevel optimal toll design problem. In *Proceedings of the 12th International symposium on dynamic games and applications*, Sophia Antipolis, France, July 2006.

[135] K. P. Sycara. Multiagent systems. *AI Magazine*, 2(19):79–92, 1998.

[136] The Modelica Association. Modelica - A Unified Object-Oriented Language for Physical Systems Modeling – Language specification. URL: http://www.modelica.org/documents/ModelicaSpec22.pdf, 2005.

[137] M. M. Thomas, J. L. Kardos, and B. Joseph. Shrinking horizon model predictive control applied to autoclave curing of composite laminate materials. In *Proceedings of the American Control Conference*, pages 505–509, Baltimore, Maryland, June 1994.

[138] V. Torczon. On the convergence of pattern search algorithms. *SIAM Journal on Optimization*, 7(1):1–25, 1997.

[139] UCTE. Final report of the investigation committee on the 28 September 2003 blackout in Italy. Technical report, Union for the Coordination of Transmission of Electricity (UCTE), Brussels, Belgium, 2003.

[140] UCTE. Final report system disturbance on 4 November 2006. Technical report, Union for the Coordination of Transmission of Electricity (UCTE), Brussels, Belgium, 2006.

[141] U.S.-Canada Power System Outage Task Force. Final report on the August 14, 2003 blackout in the United States and Canada: causes and recommendations. Technical report, April 2004.

[142] T. Van Cutsem and C. Vournas. *Voltage Stability of Electric Power Systems*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998.

[143] A. J. van der Schaft and J. M. Schumacher. *An Introduction to Hybrid Dynamical Systems*, volume 251 of *Lecture Notes in Control and Information Sciences*. Springer-Verlag, London, 2000.

[144] A. N. Venkat, I. A. Hiskens, J. B. Rawlings, and S. J. Wright. Distributed output feedback MPC for power system control. In *Proceedings of the 45th IEEE Conference on Decision and Control*, San Diego, California, December 2006.

[145] D. D. Šiljak. *Decentralized Control of Complex Systems*. Academic Press, Boston, Massachusetts, 1991.

[146] W. Wang, D. E. Rivera, and K. G. Kempf. Model predictive control strategies for supply chain management in semiconductor manufacturing. *International Journal of Production Economics*, 107(1):56–77, May 2007.

[147] G. Weiss. *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. MIT Press, Cambridge, Massachusetts, 2000.

[148] Wikipedia. List of famous wide-scale power outages. URL: http://en.wikipedia.org/wiki/List_of_power_outages, 2007.

[149] H. P. Williams. *Model Building in Mathematical Programming*. Wiley, New York, New York, 1993.

[150] M. H. Wright. Direct search methods: Once scorned, now respectable. In D. F. Griffiths and G. A. Watson, editors, *Numerical Analysis 1995*, pages 191–208. Addison Wesley, Harlow, UK, 1996.

[151] T. C. Yang, H. Cimen, and Q. M. Zhu. Decentralised load-frequency controller design based on structured singular values. *IEE Proceedings Generation, Transmission and Distribution*, 145(1):7–14, January 1998.

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

Samenvatting

Multi-Agent Modelgebaseerd Voorspellend Regelen met Toepassingen in Elektriciteitsnetwerken

Transportnetwerken, zoals elektriciteitsnetwerken, verkeersnetwerken, spoornetwerken, waternetwerken, etc., vormen de hoekstenen van onze moderne samenleving. Een soepele, efficiënte, betrouwbare en veilige werking van deze netwerken is van enorm belang voor de economische groei, het milieu en de leefbaarheid, niet alleen wanneer deze netwerken op de grenzen van hun kunnen moeten opereren, maar ook onder normale omstandigheden. Aangezien transportnetwerken dichter en dichter bij hun capaciteitslimieten moeten werken, en aangezien de dynamica van dergelijke netwerken alsmaar complexer wordt, wordt het steeds moeilijker voor de huidige regelstrategieën om adequate prestaties te leveren onder alle omstandigheden. De regeling van transportnetwerken moet daarom naar een hoger niveau gebracht worden door gebruik te maken van nieuwe geavanceerde regelstrategieën.

Elektriciteitsnetwerken vormen een specifieke klasse van transportnetwerken waarvoor nieuwe regelstrategieën in het bijzonder nodig zijn. De structuur van elektriciteitsnetwerken is aan het veranderen op verschillende niveaus. Op Europees niveau worden de elektriciteitsnetwerken van individuele landen meer en meer geïntegreerd door de aanleg van transportlijnen tussen landen. Op nationaal niveau stroomt elektriciteit niet langer alleen van het transmissienetwerk via het distributienetwerk in de richting van bedrijven en steden, maar ook in de omgekeerde richting. Daarnaast wordt op lokaal niveau regelbare belasting geïnstalleerd en kan energie lokaal gegenereerd en opgeslagen worden. Om minimumeisen en -serviceniveaus te kunnen blijven garanderen, moeten *state-of-the-art* regeltechnieken ontwikkeld en geïmplementeerd worden.

In dit proefschrift stellen wij verschillende regelstrategieën voor die erop gericht zijn om de opkomende problemen in transportnetwerken in het algemeen en elektriciteitsnetwerken in het bijzonder het hoofd te bieden. Om het grootschalige en gedistribueerde karakter van de regelproblemen te beheersen gebruiken wij *multi-agent* aanpakken, waarin verschillende regelagenten elk hun eigen deel van het netwerk regelen en samenwerken om de best mogelijke netwerkbrede prestaties te behalen. Om alle beschikbare informatie mee te kunnen nemen en om vroegtijdig te kunnen anticiperen op ongewenst gedrag maken wij gebruik van modelgebaseerd voorspellend regelen (MVR). In de regelstrategieën die wij in dit proefschrift voorstellen, combineren wij multi-agent aanpakken met MVR. Hieronder volgt een overzicht van de regelstrategieën die wij voorstellen en de regelproblemen uit de specifieke klasse van elektriciteitsnetwerken, waarop wij de voorgestelde regelstrategieën toepassen.

Multi-agent modelgebaseerd voorspellend regelen

In een multi-agent regeling is de regeling van een systeem gedistribueerd over verschillende regelagenten. De regelagenten kunnen gegroepeerd worden aan de hand van de autoriteitsrelaties die tussen de regelagenten gelden. Een dergelijke groepering resulteert in een gelaagde regelstructuur waarin regelagenten in hogere lagen meer autoriteit hebben over regelagenten in lagere lagen en waarin regelagenten in dezelfde laag dezelfde autoriteitsrelaties met betrekking tot elkaar hebben. Gebaseerd op de ideeën van MVR bepalen in multi-agent MVR de regelagenten welke actie zij nemen aan de hand van voorspellingen. Deze voorspellingen maken zij met behulp van voorspellingsmodellen van die delen van het algehele systeem die zij regelen. Daar waar de regelagenten in hogere lagen typisch minder gedetailleerde modelen en langzamere tijdschalen beschouwen, beschouwen regelagenten op lagere regellagen typisch meer gedetailleerde modelen en snellere tijdschalen. In dit proefschrift worden de volgende regelstrategieën voorgesteld en bediscussieerd:

- Voor de coördinatie van regelagenten in een regellaag wordt een nieuw serieel schema voor multi-agent MVR voorgesteld en vergeleken met een bestaand parallel schema. In de voorgestelde aanpak wordt aangenomen dat de dynamica van de deelnetwerken alleen uit continue dynamica bestaat en dat de dynamica van het algehele netwerk gemodelleerd kan worden met verbonden lineaire tijdsinvariante modellen, waarin alle variabelen continue waarden aannemen.

- In de praktijk komt het regelmatig voor dat deelnetwerken hybride dynamica vertonen, veroorzaakt door zowel continue als discrete dynamica. We bediscussiëren hoe discrete dynamica gevat kan worden in modellen bestaande uit lineaire vergelijkingen en ongelijkheden en hoe regelagenten dergelijke modellen kunnen gebruiken bij het bepalen van hun acties. Daarnaast stellen wij een uitbreiding voor van de coördinatieschema's voor continue systemen naar systemen met continue en discrete variabelen.

- Voor een individuele regelagent die richtpunten bepaalt voor regelagenten in een lagere regellaag wordt het opzetten van object-georiënteerde voorspellingsmodellen bediscussieerd. Een dergelijk object-georiënteerd voorspellingsmodel wordt dan gebruikt om een MVR-regelprobleem te formuleren. Wij stellen voor om de optimalisatietechniek *pattern search* te gebruiken om het resulterende MVR-regelprobleem op te lossen. Daarnaast stellen wij omwille van de efficiëntie een MVR-regelstrategie voor die gebaseerd is op een gelineariseerde benadering van het object-georiënteerde voorspellingsmodel.

- Regelmatig worden deelnetwerken gedefinieerd op basis van reeds bestaande netwerkregio's. Dergelijke deelnetwerken overlappen meestal niet. Als deelnetwerken echter gebaseerd worden op bijvoorbeeld invloedsgebieden van actuatoren, dan kunnen de deelnetwerken overlappend zijn. Wij stellen een regelstrategie voor voor het regelen van overlappende deelnetwerken door regelagenten in een hogere regellaag.

Multi-agent regelproblemen in elektriciteitsnetwerken

Elektriciteitsnetwerken vormen een specifieke klasse van transportnetwerken waarvoor de ontwikkeling van geavanceerde regeltechnieken noodzakelijk is om adequate prestaties te behalen. De regelstrategieën die in dit proefschrift worden voorgesteld worden daarom aan de hand van toepassing op specifieke regelproblemen uit elektriciteitsnetwerken geëvalueerd. In het bijzonder worden de volgende regelproblemen besproken:

- We beschouwen een gedistribueerd *load-frequency* probleem, wat het probleem is van het dicht bij nul houden van frequentie-afwijkingen na verstoringen. Regelagenten regelen elk hun eigen deel van het netwerk en moeten samenwerken om de best mogelijke netwerkbrede prestaties te behalen. Om deze samenwerking te bewekstellingen gebruiken de regelagenten de seriële of de parallele MVR-strategieën. We beschouwen zowel samenwerking gebaseerd op voorspellingsmodellen die alleen continue variabelen bevatten, als met gebruikmaking van voorspellingsmodellen die zowel continue als ook discrete variabelen bevatten. Met behulp van simulaties illustreren we de prestaties die de schema's kunnen behalen.

- In de nabije toekomst zullen huishoudens de mogelijkheid hebben om hun eigen energie lokaal te produceren, lokaal op te slaan, te verkopen aan een energie-aanbieder en mogelijk uit te wisselen met naburige huishoudens. We stellen een MVR-strategie voor die gebruikt kan worden door een regelagent die het energiegebruik in een huishouden regelt. Deze regelagent neemt in zijn regeling verwachte energieprijzen, voorspelde energieconsumptiepatronen en de dynamica van het huishouden mee. We illustreren de prestaties die de regelagent kan behalen voor een gegeven scenario van energieprijzen en consumptiepatronen.

- Spanningsinstabiliteiten vormen een belangrijke bron van elektriciteitsuitval. Om te voorkomen dat spanningsinstabiliteiten ontstaan is lokaal bij generatielokaties een laag van regelagenten geïnstalleerd. Een dergelijke lokale regeling werkt onder normale omstandigheden goed, maar levert ten tijde van grote verstoringen geen adequate prestaties. In dergelijke situaties moeten de acties van de lokale regelagenten gecoördineerd worden. Wij stellen een MVR-regelagent voor die tot taak heeft deze coördinatie te realiseren. De voorgestelde MVR-strategie maakt gebruik van ofwel een object-georiënteerd model van het elektriciteitsnetwerk ofwel van een benadering van dit model verkregen na linearisatie. We illustreren de prestaties die behaald kunnen worden met behulp van simulaties op een dynamisch 9-bus elektriciteitsnetwerk.

- Regeling gebaseerd op *optimal power flow* (OPF) kan gebruikt worden om in transmissienetwerken de *steady-state* spanningsprofielen te verbeteren, het overschrijden van capaciteitslimieten te voorkomen, en vermogensverliezen te minimaliseren. Een type apparaat waarvoor met behulp van OPF-regeling actuatorinstellingen bepaald kunnen worden zijn *flexible alternating current transmission systems* (FACTS). Wij beschouwen een situatie waarin verschillende FACTS-apparaten aanwezig zijn en elk FACTS-apparaat geregeld wordt door een regelagent. Elke regelagent beschouwt als zijn deelnetwerk dat deel van het netwerk dat zijn FACTS-apparaat kan beïnvloeden. Aangezien de deelnetwerken gebaseerd zijn op beïnvloedingsregio's kunnen verschillende deelnetwerken overlappend zijn. Wij stellen een coördinatie- en communicatieschema voor dat kan omgaan met een dergelijke overlap. Via simulatiestudies op een aangepast elektriciteitsnetwerk met 57 bussen illustreren we de prestaties.

Rudy R. Negenborn

Summary

Multi-Agent Model Predictive Control with Applications to Power Networks

Transportation networks, such as power distribution and transmission networks, road traffic networks, water distribution networks, railway networks, etc., are the corner stones of modern society. A smooth, efficient, reliable, and safe operation of these systems is of huge importance for the economic growth, the environment, and the quality of life, not only when the systems are pressed to the limits of their performance, but also under regular operating conditions. As transportation networks have to operate closer and closer to their capacity limits and as the dynamics of these networks become more and more complex, currently used control strategies can no longer provide adequate performance in all situations. Hence, control of transportation networks has to be advanced to a higher level using novel control techniques.

A class of transportation networks for which such new control techniques are in particular required are power networks. The structure of power networks is changing at several levels. At a European level the electricity networks of the individual countries are becoming more integrated as high-capacity power lines are constructed to enhance system security. At a national level power does not any longer only flow from the transmission network in the direction of the distribution network and onwards to the industrial sites and cities, but also in the other direction. Furthermore, at the local level controllable loads are installed, energy can be generated locally with small-scale generators, and energy can be stored locally using batteries. To still guarantee basic requirements and service levels and to meet the demands and requirements of the users while facing the changing structure of power networks, state-of-the-art control techniques have to be developed and implemented.

In this PhD thesis we propose several new control techniques designed for handling the emerging problems in transportation networks in general and power networks in particular. To manage the typically large size and distributed nature of the control problems encountered, we employ multi-agent approaches, in which several control agents each control their own part of the network and cooperate to achieve the best possible overall performance. To be able to incorporate all available information and to be able to anticipate undesired behavior at an early stage, we use model predictive control (MPC).

Next we give a summary of the control techniques proposed in this PhD thesis and the control problems from a particular class of transportation networks, viz. the class of power networks, to which we apply the proposed control techniques in order to assess their

performance.

Multi-agent model predictive control

In multi-agent control, control is distributed over several control agents. The control agents can be grouped according to the authority relationships that they have among each other. The result is a layered control structure in which control agents at higher layers have authority over control agents in lower layers, and control agents within a control layer have equal authority relationships. In multi-agent MPC, control agents take actions based on predictions that they make using a prediction model of the part of the overall system they control. At higher layers typically less detailed models and slower time scales are considered, whereas at lower layers more detailed models and faster time scales are considered.

In this PhD thesis the following control strategies for control agents at various locations in a control structure are proposed and discussed:

- For coordination of control agents within a control layer a novel serial scheme for multi-agent MPC is proposed and compared with an existing parallel scheme. In the approach it is assumed that the dynamics of the subnetworks that the control agents control are purely continuous and can be modeled with interconnected linear discrete-time time-invariant models in which all variables take on continuous values.

- In practice, the dynamics of the subnetworks may show hybrid dynamics, caused by both continuous and discrete dynamics. We discuss how discrete dynamics can be captured by systems of linear equalities and inequalities and how control agents can use this in their decision making. In addition, we propose an extension of the coordination schemes for purely continuous systems that deals with interconnected linear time-invariant subnetworks with integer inputs.

- For an individual control agent that determines set-points for control agents in a lower control layer, creating object-oriented prediction models is discussed. Such an object-oriented prediction model is then used to formulate an MPC control problem. We propose to use the optimization technique pattern search to solve the resulting MPC control problem. In addition, for efficiency reasons, we propose an MPC control strategy based on a linearization of the object-oriented prediction model.

- Commonly, subnetworks are defined based on already existing network regions. Such subnetworks typically do not overlap. However, when subnetworks are based on, e.g., regions of influence of actuators, then the subnetworks may be overlapping. For multiple control agents in a higher control layer, at which it can be assumed that the behavior of the underlying control layers is static, we propose an MPC strategy for control of overlapping subnetworks.

Multi-agent control problems in power networks

Power networks are a particular class of transportation networks and are subject to a changing structure. This changing structure requires the development of advanced control techniques in order to maintain adequate control performance. The control strategies proposed

in this PhD thesis are applied to and assessed on specific power domain control problems. In particular, we discuss the following power network problems and control approaches:

- We consider a distributed load-frequency control problem, which is the problem of maintaining frequency deviations after load disturbances close to zero. Control agents each control their own part of the network and have to cooperate in order to achieve the best possible overall network performance. The control agents achieve this by obtaining agreement on how much power should flow among the subnetworks. The serial and parallel MPC strategies are employed for this, both when the prediction models involve only continuous variables, and when the prediction models involve both continuous and discrete variables. In simulations we illustrate the performance that the schemes can obtain.

- In the near future households will be able to produce their own energy, store it locally, sell it to an energy supplier, and perhaps exchange it with neighboring households. We propose an MPC strategy to be used by a control agent controlling the energy usage in a household. This control agent takes into account expected energy prices, predicted energy consumption patterns, and the dynamics of the household, including dynamics of local energy generation and storage devices. For a given scenario of energy prices and consumption patterns, the performance that the control agent can achieve are illustrated.

- Voltage instability is a major source of power outages. To prevent voltage instability from emerging, a lower layer of control agents is installed in power networks at generation sites. These agents locally adjust generation to maintain voltage magnitudes. Such local control works well under normal operating conditions. However, under large disturbances such local control does not provide adequate performance. In such situations, the actions of the local control agents have to be coordinated. We propose an MPC control agent that has the task to coordinate the local control agents. The MPC strategy that the agent uses is based on either an object-oriented model of the power network or on a linearized approximation of this model. The object-oriented model includes a model of the physical network and the local control agents. We illustrate the performance of the MPC control agent using the object-oriented model or the linearized approximation via simulations on a dynamic 9-bus power network.

- Optimal power flow control is commonly used to improve steady-state power network security by improving the voltage profile, preventing lines from overloading, and minimizing active power losses. Using optimal power flow control, device settings for flexible alternating current transmission systems (FACTS) can be determined. We consider the situation in which there are several FACTS devices, each controlled by a different control agent. The subnetwork that each control agent considers consists of a region of influence of its FACTS device. Since the subnetworks are based on regions of influence, the subnetworks of several agents may be overlapping. We propose a coordination and communication scheme that takes this overlap into account. In simulation experiments on an adjusted 57-bus IEEE power network the performance of the scheme is illustrated.

Rudy R. Negenborn

Curriculum vitae

Rudy R. Negenborn was born on June 13, 1980 in Utrecht, The Netherlands. He finished his pre-university education (*VWO*) in 1998 at the Utrechts Stedelijk Gymnasium, Utrecht, The Netherlands. After this, Rudy Negenborn started his studies in Computer Science at the Utrecht University, Utrecht, The Netherlands. He received the title of *doctorandus* (comparable with Master of Science) in Computer Science, with a specialization in Intelligent Systems, *cum laude* from this university in 2003. For his graduation project, he performed research on Kalman filtering and robot localization. The research involved in this project was carried out during a one-year visit to the Copenhagen University, Denmark, and was supervised by Prof.Dr.Phil. P. Johansen and Dr. M. Wiering.

Since 2004, Rudy Negenborn has been working on his PhD project at the Delft Center for Systems and Control of Delft University of Technology, The Netherlands. The research of his PhD project has been on multi-agent model predictive control with applications to power networks, and has been supervised by Prof.dr.ir. B. De Schutter and Prof.dr.ir. J. Hellendoorn. During his PhD project, Rudy Negenborn obtained the DISC certificate for fulfilling the course program requirements of the Dutch Institute for Systems and Control. Furthermore, he cooperated with and spent time at various research groups, including the Hybrid System Control Group of Supélec, Rennes, France, and the Power Systems Laboratory and Automatic Control Laboratory of ETH Zürich, Zürich, Switzerland.

Rudy Negenborn's more fundamental research interests include multi-agent systems, hybrid systems, distributed control, and model predictive control. His more applied research interests include applications to transportation networks in general, and power networks in particular.

Since 2004, Rudy Negenborn has been a member of the DISC and of The Netherlands Research School for Transport, Infrastructure, and Logistics (TRAIL). Moreover, from 2004 until 2007, Rudy Negenborn fulfilled the positions of public relations representative and treasurer in the board of Promood, the representative body of the PhD candidates at Delft University of Technology.