Felix Moritz

Stochastic Model Predictive Control of Residential Buildings for Price-based Demand Response Applications

Semester Thesis, Spring 2013

Department:
EEH – Power Systems Laboratory, ETH Zürich

Supervisor:
Evangelos Vrettos, ETH Zürich

Professor:
Göran Andersson, ETH Zürich
Abstract

In this thesis, a model from a former work of a residential building is extended. In the former work, advanced deterministic control algorithms were used to optimize the operation of a building. By using forecasts of the weather, the usage of the building, the price of energy, and applying model predictive control, thermal loads were shifted to reduce the energy costs.

Because this algorithm did not incorporate the stochastic behavior of the weather, a new stochastic model predictive controller (SMPC) is implemented. In a further step, the performance of the SMPC is compared to a state-of-the-art controller which is rule based (RBC) and the deterministic model predictive controller (DMPC) of the former work. In terms of costs and violations of given comfort bounds, the SMPC outperforms the RBC and DMPC. However, this better performance comes at the cost of high computational complexity.
Preface

During my studies for my master’s degree, I had a half a year break to do my civil service. I was lucky and found an interesting work at the ”Zürcher Hochschule für angewandte Wissenschaften”, a university for applied sciences in Wädenswil (CH). My job was to support Professor Markus Hubbuch, who works on facility management. There, I got a first insight in issues related to buildings and especially what huge impact buildings have on our environment. I was not really aware of this before, but got my interest awaken.

When I was back at the university to continue studying for my master’s degree, I was looking for a semester project, which is part of my curriculum. Whilst I was browsing through the available projects, I found a description which tackled exactly this environmental issue related to residential buildings. This project was offered by Evangelos Vrettos, a Ph.D. student in Prof. Anderssions’ group, whom I shortly later contacted for more information. After an introduction and some administrational formalities, I was ready to start working on my project at the Power Systems Laboratory.

Now, when I am looking back at this project after finishing it, I must say it was a lot of work, but also I learned a lot from it. On the one hand, I had some tough times when I was stuck in debugging. But on the other hand, I experienced even better moments when the code was running and producing interesting and reasonable results. The area of convex optimization was rather new for me and it took me some time to get familiar with it, but I could always count on my patient and experienced supervisor to help me and push me back onto the right path.

I would like to thank here my supervisor Evangelos Vrettos. He gave me the possibility to work on this interesting project and supervised me for one semester. Even though his time schedule is rather busy, he always found some time to have a meeting and discuss my problems and results. His supervision has been exemplary.

I would also like to thank my girlfriend Xiaoya Tan. She gave me a lot of motivation during the project, and also spent many lunch breaks together with me.
## Contents

Abstract iii

Preface v

List of Figures ix

List of Tables xi

1 Introduction 1

2 Problem Setup 3
  2.1 Building Model 3
  2.2 Components of the Building 4
    2.2.1 Heat Pump and Heating / Cooling Slabs 5
    2.2.2 Radiator 5
    2.2.3 Electric Water Heater 5
    2.2.4 Lead-acid Battery 5
    2.2.5 Photovoltaic Modules 6
    2.2.6 Window 6
    2.2.7 Grid Connection 6
  2.3 External Parameters 6
    2.3.1 Price of Electricity 6
    2.3.2 Weather 7
    2.3.3 Internal Gains 8
    2.3.4 Domestic Hot Water Withdrawal 9
    2.3.5 Uncontrollable Load 9
  2.4 Goal 10

3 Optimization Formulation 11
  3.1 Augmentation with uncertainty model 11
  3.2 Expand model over time 12
  3.3 Model constraints (hard constraints) 12
  3.4 Deterministic Problem 12
  3.5 Stochastic Problem 13

4 Controllers 15
  4.1 Rule Based Controller (RBC) 15
  4.2 Performance Bound (PB) 16
List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Different layers of the building. The temperature in a layer represents a state. Adapted from [5].</td>
</tr>
<tr>
<td>2.2</td>
<td>The different components of the building.</td>
</tr>
<tr>
<td>2.3</td>
<td>Selling and buying price for electrical energy during summer.</td>
</tr>
<tr>
<td>2.4</td>
<td>Selling and buying price for electrical energy during winter.</td>
</tr>
<tr>
<td>2.5</td>
<td>Ambient temperature and solar global radiation during summer.</td>
</tr>
<tr>
<td>2.6</td>
<td>Ambient temperature and global solar radiation during winter.</td>
</tr>
<tr>
<td>2.7</td>
<td>Internal gains due to equipment and people.</td>
</tr>
<tr>
<td>2.8</td>
<td>Domestic hot water withdrawal.</td>
</tr>
<tr>
<td>2.9</td>
<td>Uncontrollable load during summer.</td>
</tr>
<tr>
<td>2.10</td>
<td>Uncontrollable load during winter.</td>
</tr>
<tr>
<td>5.1</td>
<td>Comparison of the total costs of the different controllers during summer.</td>
</tr>
<tr>
<td>5.2</td>
<td>Comparison of the comfort violations of the different controllers during summer.</td>
</tr>
<tr>
<td>5.3</td>
<td>Heating and cooling of the SMPC with a prediction horizon of 12 h during summer.</td>
</tr>
<tr>
<td>5.4</td>
<td>Power flows from and to the battery and electric water heater of the SMPC with a prediction horizon of 12 h during summer.</td>
</tr>
<tr>
<td>5.5</td>
<td>Comparison of the total costs of the different controllers during winter.</td>
</tr>
<tr>
<td>5.6</td>
<td>Comparison of the comfort violations of the different controllers during winter.</td>
</tr>
<tr>
<td>5.7</td>
<td>Comparing the comfort violations of the different controllers during winter.</td>
</tr>
</tbody>
</table>
List of Tables

5.1 Comparing the simulation times of the different controllers for one week during summer. ......................................................... 24
5.2 Comparing the simulation times of the different controllers for one week during winter. ......................................................... 25
Chapter 1

Introduction

Nowadays, with the progressing global climate warming, energy questions are getting more and more important. To lessen the progressing climate change, not only attention to a sustainable energy production has to be paid, but also to an economical consumption.

A sustainable energy production implies the use of renewables, i.e. besides hydropower also the use of solar and wind energy. These renewables introduce stochastic energy production patterns, which pose new challenges to the power grid. Therefore, to counteract these fluctuations in energy generation, it is essential to have a reliable way to control the demand of energy.

In the year 2012, the total electrical consumption of all the households in Switzerland was 18.3 TWh, which is 31.1 % of the total electrical consumption of Switzerland [1]. Being able to influence such a huge load is very interesting for any grid operator.

In 2012, a master student analyzed the potential cost savings and load shiftings for a residential building in a dynamic price environment. He used modern control algorithms to control the actuators of a building. The conclusion of his work show, that a reduction of the operational costs by 10 % is possible, and the demand of thermal loads can be shifted [5]. In a next step, the results from this master project were further elaborated and presented at the European Control Conference [12].

In this semester project, the simulations of [5] shall be repeated with an extended setup. For the weather predictions, an uncertainty will be introduced, which allows a more realistic evaluation of the potential cost and energy savings. This results in adjusting the Model Predictive Control algorithm of [5] according to [8], to deal with the stochastic behavior of the weather.
Chapter 2

Problem Setup

For this project, we have an identical setup as in [5] and [12]. Therefore, the model of the simulated building will not be introduced as detailed as in the previous work, but it will be summarized in this chapter such that the reader can understand the concept. For deeper understanding, we suggest to study [5] and [12].

All the mathematical models which are introduced here were implemented in Matlab. The simulations are executed with recent weather forecast and measurement data of a measurement station in Zurich (CH) from MeteoSwiss (data from August 2011 to July 2012).

2.1 Building Model

In general, we are looking at a simplified building which originates from the Opticontrol-project [3]. This building is not an existing one, but is virtually located in Zurich (CH). It has one single room which measures 100 $m^2$.

The model separates the room, floor, and ceiling of the building in different layers as can be seen in figure 2.1. Each temperature of one of these layers represents a state. These layers interact with each other i.e. heat is transferred between the different layers. In addition, the temperature in the layers is also influenced by different actuators, the weather, and the usage of the building.
Figure 2.1: Different layers of the building. The temperature in a layer represents a state. Adapted from [5].

2.2 Components of the Building

The model of our building describes different components as can be seen in figure 2.2.

Figure 2.2: The different components of the building.
2.2.1 Heat Pump and Heating / Cooling Slabs

We consider an air-to-water heat pump. Using the ambient air, it heats or cools a water system which circulates inside the layers of the building. Its efficiency is dependent on the ambient air temperature and the temperature of the supply water of the cooling circuit. The coefficient of performance (COP) describes how much energy for heating can be produced with a certain amount of electrical energy. For calculating the COP for heating, we use a linear model from [10]:

\[
COP_{heating} = 5.593 + 0.0569 \cdot T_{ambient} - 0.0661 \cdot T_{supply}
\]  

(2.1)

In addition to the heat pump, there are also slabs for heating and cooling. They are also a system consisting of a heat pump and a water circuit. In comparison to the heat pump model, the efficiency of the heating / cooling slabs is modeled to be independent of the ambient air and supply water temperature and therefore constant over time (\(\eta_{slabs} = 1.02\)). In a real building, there would not be two different systems which use the same physical principle, but they are here present because of modeling reasons.

In the simulations of this semester project, only the heat pump was used for heating, and only the slabs were used for cooling. The maximum electrical power usable for the heat pump was \(P_{HP} = 2.0\) kW and for cooling with the slabs was \(P_{slabs-cooling} = 2.0\) kW.

2.2.2 Radiator

The radiator is a simple resistor heater located directly inside the room. Its efficiency is \(\eta_{rad} = 0.903\). Because of its location inside the room, it can heat up the room with a very small time-constant in comparison to the heat pump.

2.2.3 Electric Water Heater

The electrical water heater heats the domestic hot water for the daily usage. The average water temperature in the heater should be above 55°C to supply the residents at any time with hot water for showering and cleaning. At the same time, this restricts the reproduction of legionella. The peak heating power is \(P_{EWH} = 4.1\) kW. For modelling purposes, the water inside the heater is divided into different water layers according to [11]. Each water layer has a certain temperature, which again is a state of our building model.

2.2.4 Lead-acid Battery

A lead-acid battery allows to store energy. This battery helps to react to the varying price signal. Its nominal capacity is \(C_{bat,N} = 5\) kWh in our simulations. In [13] a variant
of the Kinetic Battery Model from [7] is proposed, which is used for the simulations. The lead-acid battery model describes a two-well system. One well represents the available electrical energy, whereas the other well represents the chemical bound energy. The state of charge of those two wells are each a state in our building model.

2.2.5 Photovoltaic Modules

There are twelve photovoltaic modules installed on the top of the roof. All together they have a peak power of $P_{PV,p} = 1.92$ kW. The model for the photovoltaic modules is adopted from [2]. In this model, the modules are equipped with a DC/DC converter which allows to do a Maximum Power Point Tracking. The power output of the modules depends linearly on the cell temperature and nonlinearly on the solar radiation.

2.2.6 Window

The solar radiation can penetrate the building through a window which faces south. A certain fraction of the solar radiation gets reflected, some part absorbed, and another part enters the room and heats it up from the inside.

2.2.7 Grid Connection

The building is connected to the electrical grid. Energy can be bought from it and also be sold to it. There is no limit on the amount of exchanged energy.

2.3 External Parameters

There are several external parameters influencing our simulations. These are the data inputs for the simulation and cannot be influenced.

2.3.1 Price of Electricity

We have a varying price signal, which is the same for buying and selling electricity to the grid. This price signal is adapted from the work of [9]. It is based on the Swiss spot market prices of 2009 and can be used to shift loads to hours of high infeed of renewable energies. After the complete liberalization of the Swiss energy market, this could be a possible price signal for customers with less than 100 MWh of yearly electrical energy consumption.
Chapter 2. Problem Setup

2.3.2 Weather

The weather has a big influence on our building. The global solar radiation penetrates the window and will heat the room up. It also allows the photovoltaic modules to produce electrical energy.

The ambient temperature heats up or cools down the outer shell of the building. Because the building shell is never completely sealed, there can be tiny cracks and gaps especially close to windows and doors, air can also enter the building. This air influences directly the temperature of the room.

When we compare figures 2.5 and 2.6, we can see that the average ambient temperature during winter is much lower than during summer. The same is true for the global solar radiation.
2.3.3 Internal Gains

We assume that the residents of our building are either at home, or not. Their body radiates heat, and if they are at home radiates it to the room. At the same time they will use some electrical equipment which will also radiate heat. However, the electrical equipment will take on a kind of stand-by mode if nobody is at home and will never be turned of completely.

In figure 2.7, we can see this binary pattern of the residents being at home or not being at home.
2.3.4 Domestic Hot Water Withdrawal

The users of the building will use domestic hot water for showering, bathing, washing hands, doing dishes, etc. For this, we assume an average daily usage of 280 l water and a consumption profile as shown as in figure 2.8, which is adapted from [11].

![Diagram of Domestic Hot Water Withdrawal](image)

2.3.5 Uncontrollable Load

The users will use electrical equipment for entertainment, light, cooking, etc. whilst they are at home. This load cannot be controlled without disturbing the activities of the inhabitants, and therefore it must be provided at any time. The dataset is adapted from [4].

![Diagram of Uncontrollable Load](image)
2.4 Goal

We assume that we are the owner of the building and we want to minimize the running costs. At the same time we do not want any restrictions on our comfort. This means, we want to have at any time domestic hot water to take per example a hot shower, and we do not want to freeze or sweat when we are at home.

Therefore, we define a lower bound on the average water temperature in the electrical water heater:

\[ 55^\circ C < T_{\text{EWH}} \]  

(2.2)

We also define a lower and an upper bound on our room temperature:

\[ 21^\circ C < T_{\text{room}} < 23^\circ C \]  

(2.3)
Chapter 3

Optimization Formulation

To fulfill the goal of Chapter 2.4, we have to describe our building, its components and the external parameters in a mathematical way. By doing this, we end up with states of the building and certain components, inputs which tell us how to control our actuators, and disturbances which influence our building. In Appendix A the states, inputs, and disturbances are listed.

The states will change over time and will be different for every time-step $k$. Depending on the former state ($x_k$), the inputs ($u_k$), and the disturbances on the building ($v_k$), the states of the next time-step ($x_{k+1}$) change:

$$x_{k+1} = A \cdot x_k + B \cdot u_k + B_v \cdot v_k$$

(3.1)

The matrices $A$, $B$ and $B_v$ model the building and are adapted from the former work [5].

3.1 Augmentation with uncertainty model

Whereas the disturbances $v_k$ were known in [5], this is not the case in reality. Usually, there can be a prediction made, but this prediction is always faulty to a certain extend. To account for this error, we introduce a new uncertainty model of our disturbances according to [8, p. 86]:

$$
\begin{align*}
v_k &= \bar{v}_k + \tilde{v}_k \\
\tilde{v}_{k+1} &= F \cdot \tilde{v}_k + w_k \\
w_k &\sim N(0, 1)
\end{align*}
$$

(3.2)

In equation 3.2, $\bar{v}_k$ stands for the prediction, $\tilde{v}_k$ for the prediction error, and $w_k$ for some normally distributed noise. This means, that the prediction error is not only random, but also depends on the former prediction error. Therefore, we need to keep
track of our prediction error by introducing it as additional states:

\[
\begin{bmatrix}
   x_{k+1} \\
   \tilde{v}_{k+1} \\
   \tilde{x}_{k+1}
\end{bmatrix} =
\begin{bmatrix}
   A & B_u F \\
   0 & F \\
   \tilde{A}_k & \tilde{B}_k
\end{bmatrix}
\begin{bmatrix}
   x_k \\
   \tilde{v}_k \\
   \tilde{x}_k
\end{bmatrix} +
\begin{bmatrix}
   B \\
   0 \\
   \tilde{B}_k
\end{bmatrix} u_k +
\begin{bmatrix}
   B_v \\
   0 \\
   \tilde{B}_v
\end{bmatrix} v_k +
\begin{bmatrix}
   1
\end{bmatrix} w_k
\]  

(3.3)

### 3.2 Expand model over time

Because we look at our building-model over a certain prediction horizon \(N\), we can write the relations of the states, inputs, and disturbances in one single equation for all the time steps:

\[
x = Ax_0 + Bu + Hv + Ew
\]  

(3.4)

We can find the matrices \(A, B, H, E\) in [8, Appendix A]:

\[
x = \begin{bmatrix}
   \tilde{x}_1 \\
   \tilde{x}_2 \\
   \vdots \\
   \tilde{x}_N
\end{bmatrix},
\quad u = \begin{bmatrix}
   u_1 \\
   u_2 \\
   \vdots \\
   u_N
\end{bmatrix},
\quad A = \begin{bmatrix}
   \tilde{A}_1 \\
   \tilde{A}_2 \tilde{A}_1 \\
   \vdots \\
   \tilde{A}_N \tilde{A}_{N-1} \cdots \tilde{A}_1
\end{bmatrix},
\quad B = \begin{bmatrix}
   \tilde{B}_1 & 0 & \cdots & \cdots \\
   \tilde{A}_2 \tilde{B}_1 & \tilde{B}_2 & 0 & \cdots \\
   \vdots & \vdots & \ddots & \ddots & 0 \\
   \tilde{A}_N \cdots \tilde{A}_2 \tilde{B}_1 \tilde{A}_N \cdots \tilde{A}_3 \tilde{B}_2 \cdots & \tilde{B}_N
\end{bmatrix}
\]  

(3.5)

The matrices \(H\) and \(E\) can be formed in an analogous way to \(B\). It is important to be aware, that in our case \(A_k\) and \(B_k\) will change over time, which is because of the domestic hot water withdrawal.

### 3.3 Model constraints (hard constraints)

Our model represents a real physical building. Therefore, it has to follow physical laws and limitations. Also, we want to keep certain values bounded due to other reasons, like having a comfortable room temperature.

In order to do that, we introduce hard constraints on the inputs \(u\) and the states \(x\). We can adapt those constraints from [5]. They can be written in two matrix-equations:

\[
Su \leq s \\
Gx \leq g
\]  

(3.6)

### 3.4 Deterministic Problem

In the deterministic case, we know all the disturbances over our prediction horizon perfectly. This means, that \(w_k = 0\) and \(v_k = 0\) for all time steps \(k\).

With the help of equation 3.1 we can reformulate the state constraints in equation 3.6 and get only input constraints. Our new optimization problem looks now like
Chapter 3. Optimization Formulation

this:

\[
\begin{align*}
\text{minimize} & \quad c_d^T u \\
\text{subject to} & \quad S u \leq s \\
& \quad G B u \leq g - G A x_0 - G H v
\end{align*}
\]  

(3.7)

The matrices \( S, G \), the vectors \( s, g \), and the cost vector \( c_{dp} \) can be found in [5] or in [12].

3.5 Stochastic Problem

In comparison to the deterministic problem, \( w_k \) and \( v_k \) are not anymore zero, and we do not need to fulfill the state constraints at all times in the stochastic problem. Therefore, the hard constraints of the states in equation 3.6 can be reformulated to:

\[
P[G_i(\bar{x} + Bu + H\sigma + Ew) \leq g_i] \geq 1 - \alpha_i, \quad \forall i = 1, ..., r
\]

(3.8)

\( \alpha \) can be chosen for each individual state constraint. It defines how often a constraint can be violated, and gives us the opportunity to tune our problem.

With this adjusted state constraints, we get the new stochastic problem:

\[
\begin{align*}
\text{minimize} & \quad c_{sp}^T u \\
\text{subject to} & \quad S u \leq s \\
& \quad P[G_i(\bar{x} + Bu + H\sigma + Ew) \leq g_i] \geq 1 - \alpha_i, \quad \forall i = 1, ..., r
\end{align*}
\]

(3.9)

However, this problem is not easy to solve, because it depends on the normally distributed uncertainty \( \omega \). By using affine disturbance feedback, we can parameterize the control inputs as affine functions of the disturbance sequence:

\[
\begin{align*}
\bar{u} & = M \omega + h \\
M & = \begin{bmatrix}
0 & \cdots & \cdots & 0 \\
m_{2,0} & \ddots & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots \\
m_{nu-N,0} & \cdots & m_{nu-N,N-1} & 0
\end{bmatrix} \in \mathbb{R}^{nu \times N} \\
h & = [h_1^T, ..., h_{nu-N}^T]^T \in \mathbb{R}^{nu \times 1}
\end{align*}
\]

(3.10)

Because this parameterization of the control inputs leads to an unbounded value range due to the normal distribution, we have to relax the hard constraints on the inputs in equation 3.6. We do this in a similar way as for the state constraints, just the probability to satisfy this constraint should be much higher, because it represents a hard constraint:

\[
P[S_j u \leq s_j] \geq 1 - \alpha_{u,j}, \quad \forall j = 1, ..., q
\]

(3.11)
We can now do a deterministic reformulation of our chance constraints to have hard constraints in the form of second-order cone constraints:

\[
P[G_i(Ax_0 + B(Mw + h) + Hv + Ew) \leq g_i] \geq 1 - \alpha_{x,i}, \quad \forall i = 1, ..., r
\]
\[
P[S_ju \leq s_j] \geq 1 - \alpha_{u,j}, \quad \forall j = 1, ..., q
\]
\[
\Downarrow
\]
\[
\phi^{-1}(1 - \alpha_{x,i}) \|G_i(BM + E)\|_2 \leq g_i - G_i(Ax_0 + Bh + Hv)
\]
\[
\phi^{-1}(1 - \alpha_{u,j}) \|S_jM\|_2 \leq s_j - S_jh
\]

(3.12)

Summarized, this leads to a new convex problem which can be solved in an computational efficient way:

\[
\text{minimize} \quad c^T h
\]
\[
\text{subject to} \quad \phi^{-1}(1 - \alpha_{x,i}) \|G_i(BM + E)\|_2 \leq g_i - G_i(Ax_0 + Bh + Hv)
\]
\[
\phi^{-1}(1 - \alpha_{u,j}) \|S_jM\|_2 \leq s_j - S_jh
\]

(3.13)

Whereas, \(\phi^{-1}\) is the quantile function (inverse of the Gaussian cumulative distribution function).

To implement this optimization problem in an easy way in Matlab, the help of the toolbox Yalmip was used [6]. To further solve the actual optimization problem, Yalmip called the solver Gurobi. In a first programming attempt, the solver Gurobi was called without Yalmip. This led to several problems which could not be solved, and therefore the help of Yalmip was used. In Appendix B, a way of reformulating the optimization problem for handing it over to Gurobi directly is written down.
Chapter 4

Controllers

In this project, not only extending [5] with a stochastic approach meant to be done, but also comparing this extended stochastic MPC with other controllers. Four different controllers were compared, which are introduced in this Chapter.

4.1 Rule Based Controller (RBC)

The state of the art controller is a rule based controller. By comparing sensor measurements with reference values, actuators can be switched on or off. This controller does not need any complicated control hardware and is therefore cheap and simple to implement. The performance of the RBC can be looked at as a lower performance bound. Any controller with higher intelligence should be able to beat the RBC, otherwise it would be a lowering of current standards.

The rule based control strategy for the simulations in this thesis has five rules:

- If the room temperature is below the lower bound of the room comfort zone plus a security margin, switch on the heat pump at full rated power by buying energy from the grid.

- If the room temperature is above the upper bound of the room comfort zone minus a security margin, switch on the cooling slabs at full rated power by buying energy from the grid.

- If the average electric water heater temperature is below the lower bound of the EWH comfort zone plus a security margin, switch on the EWH at full rated power by buying energy from the grid.

- Sell all energy produced by the photovoltaic modules to the grid (battery is only charged by energy from the grid).
• Buy all the energy for the uncontrollable loads from the grid.

To directly sell all the energy produced by the photovoltaic modules to the grid is an assumption. There can be a self-consumption of the produced energy, which can have the advantage of stressing the power grid less. There even exist now incentives in Germany for PV self-consumption. Because this is outside of the thesis, we refer the interested reader to [14].

Introducing security margins, or in other words making the comfort zones of the RBC narrower than for the predictive controllers, is necessary to have a fair controller comparison. The idea of the RBC is to guarantee an operation within the bounds of the comfort zones during normal operation. Only a special event, like per example a sudden change of weather, would disturb the system as heavy to violate the comfort zones. Because of a lack of intelligence, the RBC could not take any counter measurements during such a disturbance. But if the RBC used the same comfort zones as the predictive controllers, it would violate the comfort zones even during normal operation as can be easily seen.

For the simulations, a security margin of $T_{\text{margin} - \text{room}} = 0.1 \text{ K}$ was chosen.

### 4.2 Performance Bound (PB)

The performance bound is a model predictive controller with perfect predictions of weather and usage of the building. It is the solution of the optimization in equation 3.7, which minimizes the costs of running our building by not violating any given constraints. The PB gives us a benchmark of the lowest achievable costs given a certain amount of correct information. By adjusting the prediction horizon, we can give the PB more information about the future and it can perform better.

### 4.3 Deterministic Model Predictive Controller (DMPC)

The deterministic model predictive controller is in fact exactly the same controller as the PB. The only difference are the weather predictions given to the controller. The DMPC optimizes the costs by using predictions which have an uncertainty as described in Chapter 3.1. However, the controller assumes that the predictions are perfect and therefore it will probably violate the constraints.

### 4.4 Stochastic Model Predictive Controller (SMPC)

The stochastic model predictive controller solves the optimization problem 3.13. It uses the same predictions as the DMPC containing an uncertainty. But in comparison to the
DMPC, it considers that an error might exist and therefore operates the building in a more conservative way to have lower comfort zone violations.
Chapter 5

Results

To compare the different controllers, one week in summer starting at the 1.8.2011 and one week in winter starting at the 30.1.2012 are simulated with the corresponding weather predictions and weather measurements. In the end, the costs the owner of the building would have spend are summed up. At the same time, the amount of comfort zone violations are counted. It is to mention, that not how much the comfort zones are violated is analyzed, but how often is there a comfort violation.

For all the controllers, different prediction horizons starting from 2 hours up to 12 hours are evaluated. Naturally, the varying prediction horizon has no influence on the performance of the RBC.

5.1 Summer

Figure 5.1 shows the accumulated costs of one week operation of the building during a typical week in summer. Figure 5.2 shows the percentage of hours of this week, during which the constraints were violated by the different controllers. Figure 5.1 shows that the controllers using the weather predictions are considerably better in terms of costs than the traditional rule based controller.

Surprisingly, the performance bound does not guarantee lowest costs at all times. The stochastic and deterministic MPC appear to behave better for several prediction horizons. Looking at figure 5.2, this result can be explained. Because the PB knows the exact future weather, it optimizes the building operation without any violations of the comfort zones. On the other hand, the DMPC uses weather predictions without considering that they contain some uncertainty, and will therefore most probable violate the comfort zones. Its performance depends highly on the data used for the simulations. The smaller the prediction error gets, the closer its performance will be at the performance bound, because both use the same algorithm. However, it can be said by looking at the results of this simulation, that the DMPC might perform in a very unsatisfying way. The constraint violations can become way too high to be acceptable.
In contrary, an explicit upper bound of comfort violations is defined inside the SMPC. For the simulations presented in Chapter 5, a value of 10 % was set as an upper bound of comfort violations. As we can see in figure 5.2, the SMPC keeps away from this limit for all prediction horizons. Also interesting to see is, that the cost savings of the SMPC with a prediction horizon of 12 h are approximately 25 % compared to the RBC during summer.

As an example of how the cooling in summer works, the power flows to the electric water heater, and the power flows to the battery are shown in figures 5.3 and 5.4. The controller for these results is a SMPC with a prediction horizon of 12 h. It can nicely be seen, that there is a shift of load. The power flows to the EWH, to the battery, and from the battery are highly related to the price signal of figure 2.3 and react in a opposite behavior.

![Graph](image.png)

**Figure 5.1:** Comparison of the total costs of the different controllers during summer.
Chapter 6. Results

Figure 5.2: Comparison of the comfort violations of the different controllers during summer.

Figure 5.3: Heating and cooling of the SMPC with a prediction horizon of 12 h during summer.
Chapter 6. *Results*

22

Figure 5.4: Power flows from and to the battery and electric water heater of the SMPC with a prediction horizon of 12 hours during summer.

5.2 Winter

Comparing the performance of the controllers during winter, gives us quite surprising results. In figure 5.5, we can see that the RBC beats in terms of costs all the predictive controllers with a prediction horizon smaller than 6 hours. This cannot be explained at first sight and is counterintuitive.

For explaining these results, we can look for example at the heating operations of the SMPC with a prediction horizon of 4 hours in figure 5.7. It can be seen that this controller uses mainly the radiator to heat the building. The heat pump is used very little even though its efficiency would be much higher. The reason for this is that the radiator is located directly inside the room, but the water supply system of the heat pump is located several layers deep inside the ceiling and the floor. Until the heat from the water supply system has penetrated all the layers and started heating up the room, substantial time has passed. The prediction horizon of the predictive controllers must be big enough, that for a certain amount of electrical energy sent to the heat pump, more heat can diffuse through the layers to the inside of the room than the radiator would produce with the same amount of electrical energy.

Looking at the controllers with a prediction horizon bigger than 6 hours, we can observe similar results as during summer, and this time, the cost savings of the SMPC with a prediction horizon of 12 hours are approximately 20% compared to the RBC.
Figure 5.5: Comparison of the total costs of the different controllers during winter.

Figure 5.6: Comparison of the comfort violations of the different controllers during winter.
Chapter 6. Results

5.3 Simulation Time

All the simulations were executed on a computer of the students room of the Power Systems Laboratory. The computers there are equipped with 8 GB of RAM and an up-to-date quad-core processor from Intel (i7 2600k).

In table 5.1 and 5.2 the times used for simulating one week are listed. It can be seen, that the optimization times for the RBC, PB and the DMPC are reasonable fast for any prediction horizon. However, the SMPC needs much more time to be solved. For a prediction horizon of 12 h, it even is calculating more or less in real-time. This is of course a huge drawback to use the SMPC in a real setup. For this project, we had a simplified model, and used to solve the optimizations advanced software which need expensive licenses and powerful hardware. For a residential building, the costs for the licenses and the costs for running the controller hardware must also be part of the overall comparison.

Table 5.1: Comparing the simulation times of the different controllers for one week during summer.

<table>
<thead>
<tr>
<th>Prediction horizon</th>
<th>2 h</th>
<th>4 h</th>
<th>6 h</th>
<th>8 h</th>
<th>10 h</th>
<th>12 h</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBC</td>
<td>0.4 s</td>
<td>0.4 s</td>
<td>0.4 s</td>
<td>0.4 s</td>
<td>0.4 s</td>
<td>0.4 s</td>
</tr>
<tr>
<td>PB</td>
<td>1.0 s</td>
<td>1.3 s</td>
<td>1.9 s</td>
<td>2.8 s</td>
<td>4.1 s</td>
<td>5.7 s</td>
</tr>
<tr>
<td>DMPC</td>
<td>1.0 s</td>
<td>1.4 s</td>
<td>2.2 s</td>
<td>3.0 s</td>
<td>4.4 s</td>
<td>6.1 s</td>
</tr>
<tr>
<td>SMPC</td>
<td>0.9 h</td>
<td>4.4 h</td>
<td>15.1 h</td>
<td>45.7 h</td>
<td>84.3 h</td>
<td>175.5 h</td>
</tr>
</tbody>
</table>
Table 5.2: Comparing the simulation times of the different controllers for one week during winter.

<table>
<thead>
<tr>
<th>Prediction horizon</th>
<th>2 h</th>
<th>4 h</th>
<th>6 h</th>
<th>8 h</th>
<th>10 h</th>
<th>12 h</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBC</td>
<td>0.5 s</td>
<td>0.5 s</td>
<td>0.5 s</td>
<td>0.5 s</td>
<td>0.5 s</td>
<td>0.5 s</td>
</tr>
<tr>
<td>PB</td>
<td>1.2 s</td>
<td>2.1 s</td>
<td>3.2 s</td>
<td>4.4 s</td>
<td>7.0 s</td>
<td>10.4 s</td>
</tr>
<tr>
<td>DMPC</td>
<td>1.3 s</td>
<td>2.0 s</td>
<td>2.9 s</td>
<td>5.5 s</td>
<td>8.2 s</td>
<td>9.4 s</td>
</tr>
<tr>
<td>SMPC</td>
<td>0.6 h</td>
<td>3.9 h</td>
<td>13.6 h</td>
<td>34.7 h</td>
<td>77.1 h</td>
<td>154.0 h</td>
</tr>
</tbody>
</table>
Chapter 6

Discussion and Outlook

This semester project compared different controllers for optimal building operation in the presence of time-varying electricity prices. Nowadays, the RBC is the state-of-the-art controller for residential buildings, and by looking at the simulation results it will stay the state-of-the-art controller for still some time. We could show, that the DMPC violated the defined comfort zones too often to be an acceptable alternative to the RBC. The SMPC was able to limit those violations by operating the building in a more conservative way, and was at the same time able to save about 20 % of the energy costs in comparison to the RBC. However, the SMPC needs lots of computational power, which is usually, with a typical control hardware of a residential building, not available.

Nevertheless, in the eyes of the author, it would be a possible solution to operate the building with a DMPC if a real implementation should be made. In this case, the comfort zones would need to be tuned and narrowed in a similar way as it was done for the RBC, see chapter 4.1, to lessen the amount of violations.

Further work for this project could include introducing uncertainties for other disturbances than the weather. This is a very challenging task though, because the other uncertainties are not normally distributed, and therefore the approach taken in 3.5 would not work anymore.

Another interesting work would be to analyze the violations of the comfort zones further. In this thesis, the number of violations which occurred over time were summed up to compare the controllers. To have a fair comparison, it would be also necessary to compare how big these violations of the comfort zones are. This comes much closer to how a resident would judge the performance of his heating system.

Investment issues such as battery and PV sizing were ommited in this project due to time limitations. However, they could well be the subject of future work.
Appendix A

Declaration of Variables

A.1 Input Variables

\[ u_1 = \text{power from grid to EWH} / EWH_{\text{max power}} \quad \epsilon [0, 1] \]
\[ u_2 = \text{power from PV to EWH} / EWH_{\text{max power}} \quad \epsilon [0, 1] \]
\[ u_3 = \text{power from battery to EWH} / EWH_{\text{max power}} \quad \epsilon [0, 1] \]
\[ u_4 = \text{power from battery to grid (discharge)} \quad [\text{kW}] \]
\[ u_5 = \text{power from grid to battery (charge)} \quad [\text{kW}] \]
\[ u_6 = \text{power from PV to battery (charge)} \quad [\text{kW}] \]
\[ u_7 = \text{power from battery to uc load} \quad [\text{kW}] \]
\[ u_8 = \text{power from battery to HP} \quad [\text{kW}] \]
\[ u_9 = \text{power from battery to linear HC} \quad [\text{kW}] \]
\[ u_{10} = \text{thermal heating power slabs (linear HC)} \quad [\text{kW}] \]
\[ u_{11} = \text{thermal cooling power slabs (linear HC)} \quad [\text{kW}] \]
\[ u_{12} = \text{thermal heating power radiator (linear HC)} \quad [\text{kW}] \]
\[ u_{13} = \text{thermal heating power HP} \quad [\text{kW}] \]
\[ u_{14} = \text{power from PV to uc load} \quad [\text{kW}] \]
\[ u_{15} = \text{power from PV to HP} \quad [\text{kW}] \]
\[ u_{16} = \text{power from PV to linear HC} \quad [\text{kW}] \]
\[ u_{17} = \text{power from PV to grid} \quad [\text{kW}] \]
\[ u_{18} = \text{power from grid to HP} \quad [\text{kW}] \]
\[ u_{19} = \text{power from grid to linear HC} \quad [\text{kW}] \]
\[ u_{20} = \text{power from grid to uc load} \quad [\text{kW}] \]
A.2 State Variables

\[ x_1 = \text{temperature of incoming water} \quad [\degree C] \]
\[ x_2 = \text{temperature of first layer in EWH (bottom)} \quad [\degree C] \]
\[ \ldots \]
\[ x_{11} = \text{temperature of 10th layer in EWH (top)} \quad [\degree C] \]
\[ x_{12} = \text{ambient temperature of EWH} \quad [\degree C] \]
\[ x_{13} = \text{available charge of battery} \quad [\text{kWh}] \]
\[ x_{14} = \text{bound charge of battery} \quad [\text{kWh}] \]
\[ x_{15} = \text{room temperature (X1)} \quad [\degree C] \]
\[ x_{16} = \text{temperature in building (X2)} \quad [\degree C] \]
\[ \ldots \]
\[ x_{26} = \text{temperature in building (X12)} \quad [\degree C] \]
\[ x_{27} = \text{water supply temperature (X13)} \quad [\degree C] \]
\[ x_{28} = \text{water return temperature (X14)} \quad [\degree C] \]

A.3 Disturbances

\[ v_1 = \text{solar global irradiation} \quad [\text{W/m}^2] \]
\[ v_2 = \text{air temperature} \quad [\degree C] \]
\[ v_3 = \text{internal gains due to people} \quad [\text{W/m}^2] \]
\[ v_4 = \text{internal gains due to equipment} \quad [\text{W/m}^2] \]
\[ v_5 = \text{domestic hot water withdrawal} \quad [\text{l/s}] \]
\[ v_6 = \text{uncontrollable load} \quad [\text{kW}] \]
Appendix B

Solver Gurobi

In a first attempt, it was planned to pass the optimization problem directly to the solver Gurobi. After implementing the code which called Gurobi directly, there was some error which generated faulty results. After many hours of debugging, the easier way was chosen by calling Gurobi over Yalmip. Nevertheless, the idea of how to rewrite the second-order-cone-problem in an acceptable way for Gurobi was developed. To help somebody who encounters a similar problem, this reformulation is attached to the thesis in this Appendix.

B.1 Reformulating Problem

The gurobi-solver accepts problems in predefined syntax. Unfortunately, this differs from how we defined our problem. Let us recall here, what our optimization problem looks like:

\[
\begin{align*}
\text{minimize} & \quad c^T h \\
\text{subject to} & \quad \phi^{-1}(1 - \alpha_{u,j})\|S_j M\| \leq s_j - S_j h \\
& \quad \phi^{-1}(1 - \alpha_{x,i})\|G_i(BM + E)\| \leq g_i - G_i(Ax_0 + Bh + Hv)
\end{align*}
\] (B.1)

The optimization variables are in \( h \) and \( M \).

Gurobi asks for a problem of this form:

\[
\begin{align*}
\text{minimize} & \quad c^T x \\
\text{subject to} & \quad Ax \leq b \\
& \quad 1 \leq x \leq u \\
& \quad 0 \leq x_k
\end{align*}
\] (B.2)
B.2 Write problem in a standard form

At first, we rewrite our problem in the standardized form:

\[
\begin{align*}
\text{minimize} & \quad \tilde{c}^T \xi \\
\text{subject to} & \quad \|A_i \xi + b_i\| \leq f_i^T \xi + d_i, \quad i = 1, \ldots, n_{cu} + n_{cx} \\
\end{align*}
\] (B.3)

To achieve this, we have to write our optimization-variables into a vector \(\xi\).

To begin with, we reformulate the \(M\)-matrix responsible for the disturbance feedback such that we receive a row vector.

\[
M = \begin{bmatrix}
0 & \cdots & \cdots & 0 \\
m_{2,1} & \ddots & \ddots & \vdots \\
\vdots & \ddots & \ddots & \vdots \\
m_{nu \cdot N,1} & \cdots & m_{nu \cdot N, N-1} & 0
\end{bmatrix} \in \mathbb{R}^{n_u \cdot N \times N}
\] (B.4)

\[
\tilde{M} = \begin{bmatrix}
m_{1,1} & m_{2,1} & m_{3,1} & \cdots & m_{1,2} & m_{2,2} & m_{3,2} & \cdots
\end{bmatrix}^T \in \mathbb{R}^{n_u \cdot N^2 \times 1}
\]

Now, we can combine \(\tilde{M}\) with \(h\) to a single row-vector which contains all the optimization-variables:

\[
\xi = \begin{bmatrix}
h \\
\tilde{M}
\end{bmatrix} \in \mathbb{R}^{n_u \cdot N(N+1) \times 1}
\] (B.5)

Accordingly to the new optimization-variables, we also adjust the constraints. This means for the input constraints \((i = 1, \ldots, n_{cu})\):

\[
A_i = \begin{bmatrix} 0 & \tilde{S}_i \end{bmatrix} \\
\tilde{S}_i = \text{blkdiag}(S_i, S_i, \ldots) \\
b_i = 0 \\
f_i = \begin{bmatrix} -S_i / \phi^{-1}(1-\alpha_{u,i}) & 0 \end{bmatrix} \\
d_i = s_i / \phi^{-1}(1-\alpha_{u,i})
\] (B.6)

And for the state constraints \((i = n_{cu} + 1, \ldots, n_{cu} + n_{cx})\):

\[
A_i = \begin{bmatrix} 0 & \tilde{G}_{i-n_{cu}} \end{bmatrix} \\
\tilde{G}_{i-n_{cu}} = \text{blkdiag}(G_{i-n_{cu}}B, G_{i-n_{cu}}B, \ldots) \\
b_i = G_{i-n_{cu}}E \\
f_i = \begin{bmatrix} -G_{i-n_{cu}}B / \phi^{-1}(1-\alpha_{x,i-n_{cu}}) & 0 \end{bmatrix} \\
d_i = g_{i-n_{cu}}G_{i-n_{cu}}(A_{X0} + HV) / \phi^{-1}(1-\alpha_{x,i-n_{cu}})
\] (B.7)
Finally, the cost-vector of the objective-function needs also to be extended:

\[ \tilde{c} = [c^T \quad 0]^T \quad \in \mathbb{R}^{n_u N(N+1)x1} \]  

\[ \text{(B.8)} \]

### B.3 Write standard form to Gurobi’s form

Because of the form Gurobi asks for the second-order cone constraints, it is again necessary to do a change of variables for the optimization-variables. For this we will have for every SOC-constraint \( N_{\text{horizon}} + 1 \) optimization-variables, which we will pass to Gurobi.

So our \( i^{th} \) SOC-constraint looks in Gurobi like this:

\[
\sum_{j=(i-1)\cdot(N+1)+1}^{i\cdot(N+1)-1} x_j^2 \leq x_i^2 (N+1) \quad \text{and} \quad 0 \leq x_i (N+1)
\]

\[ \text{(B.9)} \]

We can write our new Gurobi-optimization-variables in dependence of our old ones, by defining:

\[
\tilde{A}_i = \begin{bmatrix} A_i \\ f_i^T \end{bmatrix}, \quad \tilde{b}_i = \begin{bmatrix} b_i \\ d_i \end{bmatrix}, \quad \tilde{A} = \begin{bmatrix} \tilde{A}_1 \\ \tilde{A}_2 \\ \vdots \\ \tilde{A}_{n_{cu}+n_{cr}} \end{bmatrix}, \quad \tilde{b} = \begin{bmatrix} \tilde{b}_1 \\ \tilde{b}_2 \\ \vdots \\ \tilde{b}_{n_{cu}+n_{cr}} \end{bmatrix}
\]

\[ \text{(B.10)} \]

\[ \Rightarrow \quad x = \tilde{A} \cdot \xi + \tilde{b} \]

\[ \xi = \tilde{A}^{-1} (x - \tilde{b}) \]

\[ \text{(B.11)} \]

And with this, we get a new objective function, dependable only on the Gurobi-optimization-variables:

\[ \Rightarrow \quad \min \tilde{c}^T \xi = \min \tilde{c}^T \tilde{A}^{-1} (x - \tilde{b}) = \min \tilde{c}^T \tilde{A}^{-1} x \]

\[ \text{(B.12)} \]
Bibliography


