Technische Universität Berlin

EINSTEIN CENTER Digital Future

INTERNET of THINGS Smart Buildings

Master's Thesis

IOT-MT 008

Low-level controller for Non-holonomic ominiderectional mobile robot

Lingyuan Yang
Registration number: 0406049

25. August 2019

First Examiner:	Prof. Dr.-Ing. Sergio Lucia
Second Examiner:	Prof. Dr.Ben Juurlink
Advisor:	Aftab Ahmad

Technische Universität Berlin
Einstein Center Digital Future

Chair of Internet of Things for Smart Buildings

Abstract

Building climate control provides two important benefits. First, it allows adjusting temperature and humidity to the desired values, be it comfortable working conditions in an office building, or optimal crop growth conditions in a greenhouse. Second, climate control enables the reduction of energy consumption and costs.

Minimizing the costs of operating a building while at the same time providing right conditions is hard to achieve using conventional control approaches. Model predictive control achieves this objective due to its ability to include predictions like the weather forecast into its computations. Moreover, model predictive control considers constraints like the limits of the actuators.

Nonlinear model predictive control often achieves the best results because climate dynamics are nonlinear and highly complex. Due to neglecting the nonlinear dynamics, the accuracy of linear model predictive control is lower. The lower precision reduces the cost savings and increase constraint violations compared to using nonlinear model predictive control. However, the nonlinear approach needs higher computational requirements than the linear one. Furthermore, deriving a nonlinear model is a time-consuming task and requires expertise.

Hence, we propose augmenting linear model predictive control with Gaussian processes to diminish the loss of accuracy. The Gaussian processes are used to model the neglected nonlinear dynamics and are implemented as a parameter in the linear prediction model. Due to this implementation the advantages of linear model predictive control are preserved while the drawbacks are significantly reduced.

On this work, we show on the example of a greenhouse model that fusing linear model predictive control and machine learning based on Gaussian processes results in a similar performance respective cost savings and avoidance of constraint violations as nonlinear model predictive control.

Deutsche Kurzfassung

Die Regelung der Gebäudeklimatik hat zwei große Vorteile. Erstens erlaubt sie das anpassen der Temperatur und der Luftfeuchtigkeit an die gewünschten Werte. Dies kann beispielsweise in Bürogebäuden geschehen um angenehme Arbeitsbedingungen zu erzeugen oder in einem Gewächshaus um den Pflanzenwachstum zu maximieren. Zweitens ermöglicht die Klimaregelung die Minimierung der Energieverbrauchs und -kosten.

Dabei ist das gleichzeitige Regeln eines Gebäudes und die Kostenminimierung schwer zu erreichen mit konventionellen Regelungsmethoden. Modellprädiktive Regelung kann dieses Ziel erreichen aufgrund ihrer Fähigkeit Prädiktionen wie die Wettervorhersage in ihre Berechnungen einzubeziehen. Desweiteren berücksichtigt modellprädiktive Regelung Bedingungen wie die Grenzwerte der Stellgrößen.

Mit nichtlinearer modellprädiktiver Regelung werden oft die besten Ergebnisse erzielt, weil die Klimadynamik nichtlinear und hochkomplex ist. Da lineare modellprädiktive Regelung die nichtlinearen Anteile vernachlässigt ist ihre Genauigkeit geringer. Die geringere Präzision vermindert die Kosteneinsparungen führt zu vermehrtem Verletzen der Bedingungen im Vergleich zu nichtlinearer modellprädiktiver Regelung. Jedoch erfordert das nichtlineare Verfahren mehr Rechenleistung als das Lineare. Desweiteren ist das Ableiten eines nichtlinearen Modells zeitaufwendig und erfordert weitreichende Fachkenntnisse.

Deshalb wird eine mit Gaußschen Prozessen erweiterte lineare modellprädiktive Regelung vorgestellt. Die Gaußschen Prozesse werden verwendet um die vernachlässigten Nichtlinearitäten abzubilden und somit die Genaugkeit der Prädiktionen zu erhöhen im Vergleich zu konventioneller modellprädiktiver Regelung. Die Gaußschen Prozesse werden als Parameter im linearen Prädiktionsmodell eingefügt. Dadurch bleiben die Vorteile der linearen modellprädiktiven Reglung

erhalten bei gleichzeitiger Verminderung der Nachteile.

In dieser Arbeit wurde gezeigt, dass die Kombination von linearer modellprädiktiver Regelung und Gaußschen Prozessen vergleichbare Ergebnisse erzielt bezüglich Kostenminimierung und Berücksichtigung der Bedingungen wie nichtlineare modellprädiktive Regelung.

Erklärung der Selbstständigkeit

Hiermit versichere ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Magdeburg den 25. August 2019

Benjamin Karg

Contents

Abstract **I**

Deutsche Kurzfassung **III**

1 Introduction **3**

2 Kinematic Model of Non-holonomic omnidirectional mobile robot **5**
2.1 Model overview 5
2.2 Internal air temperature model 7
2.3 Internal air humidity model 8

3 Quadratic ICR optimization problem **11**
3.1 MPC at a glance 11
3.2 Process model 13
3.3 Prediction 14
3.4 Objective function 14

4 Control framework **17**
4.1 Preliminaries 17
4.2 Covariance functions 19
4.3 Hyperparameters 20

5 Fusing predictive and learning-based control **23**
5.1 Augmented controller 23
5.2 Prediction model 25
5.3 Software 26

6 Set-point tracking **29**
6.1 Preliminaries 30
6.2 Pretests 31

6.3 Noise-free tracking 35

7 Economic model predictive control 37
7.1 Preliminaries 38
7.2 Economic climate control 39
7.3 Comparison of set-point tracking MPC and EMPC . . . 42

8 Conclusion 47

List of Figures 49

List of Tables 51

Bibliography 53

A Parameters of the greenhouse models 57
A.1 Physical parameters 57
A.2 Parameters for the linearization 58

Chapter 1

Introduction

Mobile robot has been widely used in the industry, like logistics and palletizing, to decrease the dependence on manpower. Among various structrues of mobile robot, omnidirectinal ones are gaining more and more importance due to their high maneuverability in the task space, which makes them more flexible and adaptive in complex envirement. The omnidirectinal mobile robots in the market now are mostly equiped with Swedish wheel or claster wheel, these kinds of structures are called holonomic omnidirectinal as they have 3 degree of freedom in the joint space w.r.t 3 degree of freedom in the task space. However, their complex wheel structure causes a lot of problem in practice, such as vulnerable to uneven ground and noise etc. To avoid such drawback while make use of the flexibility. We put our sight on the so called non-holonomic omnidirectinal mobile robot.

Such kind of robots are equipped with fully steerable wheels (four in our case) that can be controled individualy. To control this kind of robots effciently without actuator conflict, Instantaneous Center of Rotation(ICR) is always used to moldeling the kinematics. This ICR is defined as being the point in the robot frame that instantaneously does not move in relation to the robot. For our case, the ICR corresponds to the point where the zero motion line of each wheel intersect. In practice, we will calculate the desired ICR position/velocity based on the incoming task space command. And coordinate 4 wheels to orient to the right position to conduct the task space motion as commanded. Hoever, in practice, the incoming task space velocity command are not smooth w.r.t time, which leads to inconsist jump of ICR and thus inconsistency of steer angle of wheels. This process of initialization is a main drawback of this kind

of structure comparing to holonomic omnidirectinal one. Most study recently are focusing on pre-defined trajectories whose velocity has been smoothed in the task space[2][20], to work around such problem. Some study have been done on optimization of velocity command by means of limit the ICR movement in task space[19][21], and acheived impressive result. However, these solutions still have limitations, like in the study [19], a unique ICR point is assumed to exist all the time. Which leads to the control frame work not able to handle the simplest case-linear translation, where the ICR is in representation singularity, and also this method is less robust to external disturbance which might cause ICR not unique any more.

The objective of this paper is to make use of the achievement from previous study, mainly from [19], to do the control on ICR in task space while get rid of the limitation it introduces. To make a low level controller that is able to handle all circumstance and highly responsive to omnidirectinal velocity commands. Thus can be treat as a "real omnidirectinal" structure by the global plannar(navigation)

The remainder of this thesis is organized as follows. Chapter 2 introduce the mobile platform we are working on, its kinematic model and ICR control method. Chapter 3 describes how do we form the Quadratic ICR optimization problem . Chapter 4 describes how the kinematic and above optimization are combined into a control framework and the implementation details.

The expironment result is discussed in **??**.

Chapter 2

Kinematic Model of Non-holonomic omnidirectional mobile robot

Kinematics is the most basic study of how mechanical systems behave. In our case, Kinematics make a connection between joint states(driving rate, steering angle and steering velocity) and platform velocity in the task space. Which is fundmental for our control method. First, a overview of the mobile platform is given (Section 2.1). Afterwards, the Forward and Inverse Kinematic equations (Section 2.3) are presented.

2.1 Model overview

The the platform is a regid squre body with four conventional wheels on each corner. each wheel can be controlled independently (steering/driving).

In the task space, the frame is as shown in Fig. 2.1 we use $\xi = [x, t, \theta]$ to discribe the current platform position and orientation, use $\dot{\xi} = [\dot{x}, \dot{y}, \dot{\theta}]$ and $\ddot{\xi} = [\ddot{x}, \ddot{y}, \ddot{\theta}]$ to discribe the platform velocity and acceleration.

In the following discussion, we will mainly focus on to frames, world frame in which the odometry $\hat{\xi}$ is calculated, and body frame in which the platform velocity $\dot{\hat{\xi}}^b$ and platform acceleration $\ddot{\hat{\xi}}^b$ is calculated, the command from global planer $\dot{\xi}^b$ and $\ddot{\xi}^b$ is also assumed to be in the body frame. The lower case "b" in the upper corner indi-

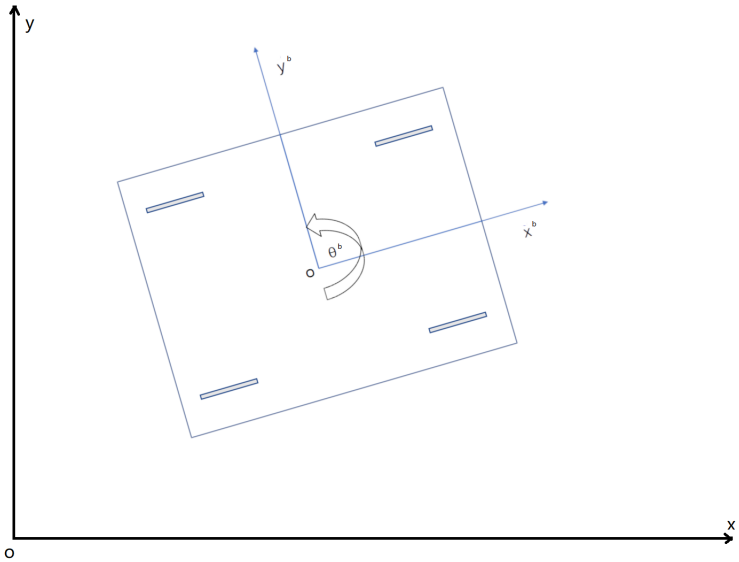


Figure 2.1: Task space

cates this variable is expressed in the body frame. While the notion with "hat" likeˆis used to expressed the measured feedback.

In the joint space we use β_i and $\dot{\beta}_i$ to discribe the steering angle and steering velocity of the wheel, and $\dot{\phi}$ for driving velocity, which is shown in figure Fig. 2.2. The measured feedback from sensor is noted withˆwhile the joint space commands are with out special notion.

$$V_{ven,flux} = D_{ws} \cdot \alpha \cdot U_{ven}^{\beta} + c_{loss} \cdot D_{ws} + c_{leak}, \tag{2.1}$$

where α and β are empirical parameters, c_{loss} is the influence factor of the windspeed when the vents are closed and c_{leak} is the leakage of warmth due to imperfect sealing. In the following two sections the two ODEs are explained. The value of the parameters are gathered in Appendix A.1.

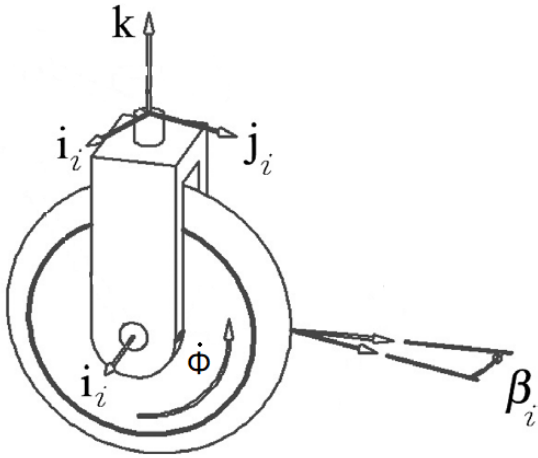


Figure 2.2: Wheel structure

2.2 Internal air temperature model

The greenhouse air temperature is modelled by the following balance:

$$\frac{dX_{T,a}}{dt} = \frac{c_{area,ss} \cdot c_{sph,a} \cdot c_{den,a}}{c_{vol,g}} \cdot (Q_{sol,a} + Q_{cnv,ss-a} + Q_{heat-a} + Q_{cnv-cnd,a-e} - Q_{ven,a-e} - Q_{loss,a-e} - Q_{trp,cr}) , \tag{2.2}$$

where $c_{area,ss}$ is the surface area of the soil, $c_{sph,a}$ is the specific heat and $c_{den,a}$ is the density of the air. The heat fluxes $Q_{(\cdot)}$ are described in the following paragraphs.

The solar radiation absorbed by the air $Q_{sol,a}$ is given by:

$$Q_{sol,a} = c_{asw,a} \cdot V_{tsw,g} \cdot D_{rs,e} \cdot (1 - U_{shd}) , \tag{2.3}$$

where $c_{asw,a}$ is the short-wave absorption coefficient of the greenhouse air. $V_{tsw,g}$ is the short wave heat transmission coefficient depending on the cover, its whitening state and the shading screen.

The heat transfer between the soil surface and the inside air is depending on the temperature difference between soil surface temperature $D_{T,ss}$ and inside air temperature $X_{T,a}$:

$$Q_{cnv,ss-a} = c_{cnv,ss-a} \cdot (D_{T,ss} - X_{T,a}), \tag{2.4}$$

where $c_{cnv,ss-a}$ is the constant convection coefficient.

This process is proportional to the temperature difference between outside air temperature $D_{T,e}$ and inside air temperature $X_{T,a}$:

$$Q_{cnd-cnv,a-e} = c_{cnd-cnv,a-e} \cdot (D_{T,ss} - X_{T,a}), \tag{2.5}$$

where $c_{cnd-cnv,ss-a}$ is the thermal loss coefficient for convection and conduction processes.

The Heat transfer to the outside air due to ventilation and infiltration are described by:

$$\begin{aligned} Q_{ven,a-e} + Q_{loss,a-e} = \\ \frac{c_{den,a} \cdot c_{sph,a}}{c_{area,ss}} \cdot V_{ven,flux} \cdot (X_{T,a} - D_{T,e}). \end{aligned} \tag{2.6}$$

2.3 Internal air humidity model

The internal air humidity is modelled by the following balance:

$$\frac{dX_{H_{a,a}}}{dt} = \frac{c_{area,ss}}{c_{den,a} \cdot c_{vol,g}} \cdot (M_{trp,cr} - M_{ven,a-e} + U_{hum}), \tag{2.7}$$

where the two sources of water vapor are the crop transpiration $M_{trp,cr}$ and the injections of the moisturisation system U_{hum}, while the loss of water vapor is due to exchange of air with the outside through ventilation.

For the water vapour increase due to the crop tomato a balance $M_{trp,cr}$ based on the Penman-Monteith equation is used:

$$\begin{aligned} M_{trp,cr} = \frac{1}{V_{r,trp}} \Big(V_{hsat,a} \\ + \frac{1}{c_{den,a}} \frac{V_{ssvp}}{c_{psyco}} \frac{V_{r,bl}}{2D_{LAI}} \frac{V_{rn,cr}}{V_{lt,vap}} - X_{Ha,a} \Big), \end{aligned} \tag{2.8}$$

where c_{psyco} is the thermodynamic psychometric constant, D_{LAI} is the leaf area index of the tomato crop and $c_{den,a}$ is the densitiy of the inside air.

The thermodynamic psychometric constant c_{psyco} is calculated by:

$$c_{psyco} = \frac{c_p \cdot p}{\lambda_v \cdot MW_{ratio}}, \tag{2.9}$$

where p is the atmospheric pressure, λ_v is the latent heat of water vaporization, c_p is the specific heat of air at constant pressure and MW_{ratio} is the ratio of molecular weight of water vapour and dry air.

The water concentration of the air at saturation $V_{hsat,a}$ is calculated with Teten's formula:

$$V_{hsat,a} = 611.2 \cdot exp\left(\frac{\frac{17.62 \cdot X_{T_{a,a}}}{243.12 + D_{T,e}}}{R_D\left(273.2 + X_{T_{a,a}}\right)}\right), \tag{2.10}$$

where R_D is the gas constant of water vapor.

The transpiration resistance $V_{r,trp}$ is described by:

$$V_{r,trp} = \frac{1}{2D_{LAI}}\left[\left(1 + \frac{V_{ssvp}}{c_{psyco}}\right)V_{r,bl} + V_{r,s}\right], \tag{2.11}$$

where $V_{r,bl}$ is the boundary layer resistance Eq. (2.12) and $V_{r,s}$ is the stomatal resistance Eq. (2.15).

$$V_{r,bl} = 220 \cdot \frac{c_{cl,cr}^{0.2}}{V_{ws,a}^{0.8}} \tag{2.12}$$

where $c_{cl,cr}$ is the characteristic length of the crop leaf and and $V_{ws,a}$ is the average inside air speed:

$$V_{ws,a} = \frac{V_{ven,flux}}{c_{area,p}}, \tag{2.13}$$

where $c_{area,p}$ is the the cross section area the air is streaming through.

The slope of the saturated vapor pressure curve V_{ssvp} is calculated by this empirical balance:

$$V_{ssvp} = 0.04145 \cdot e^{0.060888 \cdot X_{T,a}}. \tag{2.14}$$

Because the stomatal resistance of the tomato crop is mainly affected by global radiation, it can be described by:

$$V_{r,s} = 200\left(1 + \frac{1}{e^{0.05 \cdot V_{tsw,g} \cdot D_{rs,e} - 50}}\right), \tag{2.15}$$

where $V_{tsw,g}$ is the short wave heat transmission coefficient depending on the cover, its whitening state and the shading screen.

$V_{lt,vap}$ is the latent heat of evaporation of water calculated at internal air temperature using:

$$V_{lt,vap} = 4185.5 \cdot (597.0 - 0.56 \cdot X_{T,a}). \tag{2.16}$$

The loss of water vapour due to natural ventilation and infiltration $M_{ven,a-e}$ is described by:

$$\begin{aligned} M_{ven,a-e} = & \frac{c_{den,a}}{c_{area,ss}} \cdot V_{ven,flux} \cdot \left(X_{H_{a,a}} - D_{H_a,e}\right) \\ & + M_{loss,a-e}, \end{aligned} \tag{2.17}$$

where $M_{loss,a-e}$ are constant infiltration losses.

Establishing a (pseudo-) physical greenhouse model was illustrated in this chapter. The model contains nonlinear ODEs for the inside air temperature and humidity. The nonlinear greenhouse model will be used as the prediction model for nonlinear MPC. For linear MPC an analytical linearization will be derived. In the following chapter the theoretical fundamentals of MPC are presented.

Chapter 3

Quadratic ICR optimization problem

An early control method for MIMO systems is the linear quadratic regulator (LQR) for an unconstrained objective function for states and inputs. Model predictive control (MPC) is an advanced control approach evolved from LQR. [14] gives amongst other things a survey on the history of MPC.

Section 3.1 shows the basic principles of MPC. In Section 3.2 the derivation of process models and their requirements for MPC are presented. Section 3.3 shows how system evolution can be described only in dependence of the input trajectory and the current states. The objective, its effect on the control and the consideration of constraints is explained in Section 3.4.

3.1 MPC at a glance

MPC uses knowledge about a complex and often multi-variable system's properties to compute an input trajectory that is optimal with respect to an arbitrary defined objective function while considering the constraints. MPC schemes can be linear and nonlinear. For an further introduction to MPC [17] is proposed. For theory, design and applications of nonlinear MPC [9] and [7] are recommended.

Model predictive control has its roots in optimal control. Basically, a model is used to forecast the system's future behaviour. This knowledge is used to find an optimal trajectory of control inputs. For this computations, MPC needs a process model and an optimization algorithm. MPC consists of four repeating steps::

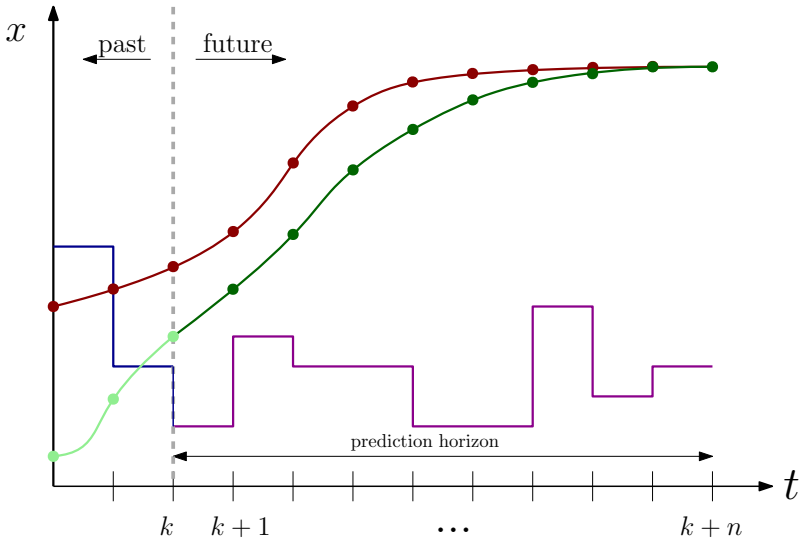


Figure 3.1: Example of a time-discrete MPC scheme: red - reference trajectory, light green - measured state trajectory, dark green - predicted state trajectory, violet - optimal control trajectory

1. Obtain the current state

2. Predict the system evolution

3. Calculate optimal input trajectory

4. Apply optimal input

Fig. 3.1 depicts one step of the MPC algorithm. The objective is that the green line, the system's real state, is as close as possible to the red line, the reference trajectory. Therefore the violet line is the optimal control trajectory of future inputs.

3.2 Process model

To guarantee a good performance of a MPC, the quality of the process model which is used for the predictions is crucial. The model has to capture the major dynamics of the real controlled system otherwise the predicted states may deviate heavily from the real states. Thus, the computed inputs may lead to constraint violations which render the control task infeasible. To derive a process model, two general approaches can be considered.

The first one is first-principles modelling where the model is derived from physical laws. The advantage of this method is that the parameters and their values have a meaning and the relations between the variables can be easily understood. But it demands much knowledge in the specific field of application.

The second approach is system identification which can be generally divided into two groups. In case of grey box system identification known relations and parameters are modelled while unknown ones are estimated via optimization with training data. When black box system identification is applied, a strictly mathematical model with a certain amount of parameters is used, for example a neuronal network or a polynomial.

For all cases described above the obtained models have to be verified for different scenarios by comparing the real system with the process model. The quality of the process model has to be optimized until it meets the requirements for a prediction model. For more information about system identification see [10] and [12].

The augmented MPC scheme presented in Chapter 5 relies on a linear continuous model:

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u}(t), \tag{3.1}$$

where $\mathbf{x}(t)$ are states and $\mathbf{u}(t)$ are the inputs at time t.

However, since for solving the optimal control problem the continuous model is fully discretized at every time step t_s using collocation our example system will be a linear discrete-time one described by:

$$\mathbf{x}_{k+1} = A_{dis}\mathbf{x}_k + B_{dis}\mathbf{u}_k, \tag{3.2}$$

where $\mathbf{x}_k$ are the states and $\mathbf{u}_k$ are the inputs at time step k.

3.3 Prediction

If we want to predict the systems behaviour further than one step into the future, we have a function that is depending on future states. In case of a prediction horizon of two time steps we can insert (3.2) in the first line of (3.3).

$$\begin{aligned}\mathbf{x}_{k+2} &= A_{dis}\mathbf{x}_{k+1} + B_{dis}\mathbf{u}_{k+1},\\ &= A_{dis}\left(A_{dis}\mathbf{x}_k + B_{dis}\mathbf{u}_k\right) + B_{dis}\mathbf{u}_{k+1},\\ &= A_{dis}^2\mathbf{x}_k + A_{dis}B_{dis}\mathbf{u}_k + B_{dis}\mathbf{u}_{k+1}.\end{aligned} \tag{3.3}$$

Because the current state $\mathbf{x}_k$ and the current input $\mathbf{u}_k$ are known, $\mathbf{x}_{k+2}$ is only depending on $\mathbf{u}(T_s)$. The same applies for predicting the state for all future time steps. For the prediction of the states of the n-th future step at time step k we get:

$$\begin{aligned}\mathbf{x}_{k+n} = A_{dis}^n\mathbf{x}_k + A_{dis}^{n-1}B_{dis}\mathbf{u}_k + \dots\\ + A_{dis}B_{dis}\mathbf{u}_{k+n-2} + B_{dis}\mathbf{u}_{k+n-1}.\end{aligned} \tag{3.4}$$

Thus, the future states are only a function of the current state $\mathbf{x}_k$ and the future inputs $\mathbf{u}_{k+i}$ which build the input trajectory.

3.4 Objective function

MPC is an optimal control approach with respect to the objective function. The objective function describes the target of the control mathematically. An example of a quadratic objective for the discrete-time state-space-model (3.2) is

$$J_{dis}(\mathbf{x},\mathbf{u}) = \sum_{k=1}^{n}(\mathbf{r}_k - \mathbf{x}_k)^T Q(\mathbf{r}_k - \mathbf{x}_k) + \mathbf{u}_k^T R\mathbf{u}_k, \tag{3.5}$$

where $\mathbf{r}_k$ is the discrete reference trajectory, n is the prediction horizon, Q is the weighting matrix for states and R is the weighting matrix for the inputs. Large positive values in Q result in a focus on sticking close to the reference trajectory while large positive values in R enforce the minimization of used actuation energy. To find the

optimal input trajectory, the objective hast to be minimized while considering the constraints.

The resulting optimal control problem is described by:

$$\min\ J_{dis}(\mathbf{x}, \mathbf{u}) \tag{3.6a}$$

subject to:

$$\mathbf{x}_{k+1} = A\mathbf{x}_k + B\mathbf{u}_k, \tag{3.6b}$$

$$\mathbf{x}_i \in \mathcal{X}, \mathbf{u}_i \in \mathcal{U}, \tag{3.6c}$$

$$x_0 = x_{\text{init}}, \tag{3.6d}$$

where $\mathcal{X}$ and $\mathcal{U}$ are the compact and bounded spaces for feasible states and feasible inputs.

The state constraints are often implemented as box constraints. Thus, they define the allowed maximum deviation from the reference trajectory. The input constraints often describe the physical limits of the actuators. As well the state constraints as the input constraints can be defined as soft and hard constraints. The violation of hard constraints lead directly to an infeasible optimization problem, whereas the violation of soft constraints add a penalty term to the objective:

$$J_{dis,soft}(\mathbf{x}, \mathbf{u}) = J(\mathbf{x}, \mathbf{u}) + \delta \sum_{k=1}^{n} \mathbf{v}_k, \tag{3.7}$$

where δ is the weight of the penalty and $\mathbf{v}_k$ is the absolute value of the violation at time step k.

(3.5) is a conventional objective function. However, there are many approaches to design a economic objective function for minimizing the costs or maximizing the profit. A maximization problem can be easily transformed to a minimization problem by:

$$\max\ J(\mathbf{x}, \mathbf{u}) = \min\ -J(\mathbf{x}, \mathbf{u}). \tag{3.8}$$

In Chapter 7 an approach to minimize the cost of operating a greenhouse is presented.

Since the models used in the following chapters are continuous the description of a continuous optimal control for (3.1) given by:

$$\min\ J_{cont}(\mathbf{x}(t), \mathbf{u}(t)) \tag{3.9a}$$

subject to:

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u}(t), \tag{3.9b}$$

$$\mathbf{x}(t) \in \mathcal{X}, \mathbf{u}(t) \in \mathcal{U} \tag{3.9c}$$

$$x_0 = x_{\text{init}}, \tag{3.9d}$$

where the objective function for the continuous case is described by:

$$J_{cont}(\mathbf{x}(t), \mathbf{u}(t)) = \int_0^{t_f} (\mathbf{r}(t) - \mathbf{x}(t))^T Q(\mathbf{r}(t) - \mathbf{x}(t)) + \mathbf{u}(t)^T R\mathbf{u}(t)\mathrm{d}t, \tag{3.10}$$

where $\mathbf{r}(t)$ is the continuous reference trajectory and t_f is the prediction horizon.

The relation between the prediction horizon of the continuous and the discrete-time is:

$$t_f = n \cdot t_s. \tag{3.11}$$

In this chapter the model predictive control approach was explained. Moreover, insight was given about prediction models, Thus, the first theoretical foundation of the expanded linear model predictive control scheme derived in Chapter 5 was presented. In the following chapter GPs are presented which build the second theoretical foundation.

Chapter 4

Control framework

Gaussian processes are a kind of supervised machine learning. They can be seen as a generalization of the Gaussian probability distribution. Instead of describing the probability distribution of random variables, the stochastic process governs the properties of functions.

Generally, two kinds of GPs can be distinguished. First, classification is used where the input is mapped to discrete classes. One well-known example is the classification of handwritten digits. Second, regression is applied where the input is mapped to a continuous output. Since this work uses GPs to estimate nonlinear dynamics (Section 5.1) the focus is on GP regression.

A quick introduction to GP classification is given in [4] and to GP regression in [3]. Modelling with GP for a tutorial example is explained in [8]. For more information about GPs [16] is proposed.

In Section 4.1 GPs are explained alongside an academic example for GP regression. Afterwards, the focus is on two more specific parts of a GP model, the kernel (Section 4.2) and his hyperparameters (Section 4.3).

4.1 Preliminaries

Gaussian process regression (GPR) is a stochastic approach to generate a function which is underlying a dataset. It is similar to polynomial (e.g. linear) regression. In both cases a function is returned which allows to compute the most likely values for arbitrary input values. For GPR this function is called mean function. The advantage of GPR is that it furthermore delivers the variance of each value which gives a hint about how reliable the value itself is.

GPR is not completely free-form because a mean and a kernel function have to be chosen leading to a prior GP. A GP is completely specified by its mean function and covariance function. We define the mean function $m(x)$ and the covariance function $k(x, x')$ of a process $f(x)$ as

$$m(x) = E[f(x)], \tag{4.1}$$

$$k(x, x') = E[(f(x) - m(x))(f(x') - m(x'))], \tag{4.2}$$

where $E[f(x)]$ is the expected value and write the GP as

$$f(x) \sim \mathcal{GP}(m(x), k(x, x')). \tag{4.3}$$

For notational simplicity, we will take the mean function to be zero. Our training data, n noisy observations at the points X, are denoted as $\mathbf{y}$. The function values at the n_* arbitrary chosen locations X_* are denoted as $\mathbf{f}_*$. Assuming identically distributed Gaussian noise with variance $\epsilon = \sigma_n^2 I$ for the measurements, the training data is described by:

$$\mathbf{y} = f(X) + \epsilon \tag{4.4}$$

Therefore the prior is

$$cov(\mathbf{y}) = K(X, X) + \sigma_n^2 I, \tag{4.5}$$

where the second summand corresponds the noise term. This leads to the joint distribution

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(X, X) & K(X, X_*) \\ K(X_*, X) & K(X_*, X_*) \end{bmatrix}\right), \tag{4.6}$$

where $K(X, X)$ is the $n \times n$ matrix containing the covariances at all pairs of training points, $K(X, X_*)$ is the $n \times n_*$ matrix containing the covarianes at all pairs of training points and test points, and similarly for $K(X_*, X)$ and $K(X_*, X_*)$.

Deriving the conditional distribution for the test points gives:

$$\mathbf{f}_* \mid X, \mathbf{y}, X_* \sim \mathcal{N}(\bar{\mathbf{f}}_*, cov(\mathbf{f}_*)). \tag{4.7}$$

The most likely values for the test points $\bar{\mathbf{f}}_*$ are described by:

$$\begin{aligned}\bar{\mathbf{f}}_* &\equiv E(\mathbf{f}_* \mid X, \mathbf{y}, X_*) \\ &= K(X_*, X)[K(X, X) + \sigma_n^2 I]^{-1}\mathbf{y}\end{aligned} \tag{4.8}$$

The corresponding covariances are given by:

$$\begin{aligned}cov(\mathbf{f}_*) &= K(X_*, K_*) \\ &- K(X_, X)[K(X, X) + \sigma_n^2 I]^{-1} K(X, X_*).\end{aligned} \tag{4.9}$$

Since $\bar{\mathbf{f}}_*$ is the best estimation for the function value at the test points it used in the augmented model predictive control scheme in Section 5.1. $cov(\mathbf{f}_*)$ is a measure for the uncertainty of the value.

4.2 Covariance functions

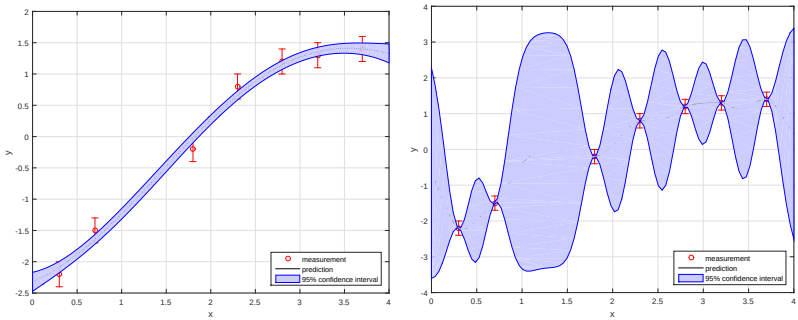
For the way the knowledge is transferred from the training data to the Gaussian process model, the choice of the covariance function, more generally a kernel, is crucial. A kernel is a mapping of a pair of inputs, $\mathbf{x} \in X$, $\mathbf{x}' \in X$ into $\mathbb{R}$. By definition, a kernel has to be symmetric, that means $k(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}', \mathbf{x})$. Given a set of test points $\mathbf{x}_i$, the Gram matrix K with entries $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$ can be computed. If the Gram matrix K fulfills

$$Q(\mathbf{v}) = \mathbf{v}^T K \mathbf{v} \geq 0 \quad \forall \quad \mathbf{v} \in \mathbb{R}^n, \tag{4.10}$$

it is positive semi-definite and therefore a covariance matrix. If additionally $Q(\mathbf{v}) = 0$ only when $\mathbf{v} = 0$, K is called positive definite and is also a covariance matrix. If a covariance function is only a function of $\mathbf{x} - \mathbf{x}'$, it is called stationary. If it is only depending on $|\mathbf{x} - \mathbf{x}'|$, it is called isotropic. One example is the standard exponential kernel

$$k_{SE}(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \cdot exp\left(-\frac{(\mathbf{x} - \mathbf{x}')^2}{2l^2}\right), \tag{4.11}$$

where σ^2 is highest possible correlation and l is the lengthscale parameter. It is possible to add and multiply different covariance functions to generate new ones.



(a) $l = 2.0$ (b) $l = 0.2$

Figure 4.1: Impact of the hyperparameters on the predictions of a GP

4.3 Hyperparameters

Depending on the choice of the covariance function $k(\mathbf{x}, \mathbf{x}')$, a set of hyperparameters θ has to be selected. In case of the squared exponential kernel 4.11 $\theta = \left\{\sigma_f^2, l\right\}$. In most practical applications the majority of the hyperparameters are free, but sometimes fixing a hyperparameter makes sense. For example if a periodic kernel should capture an effect related to the day-night-cycle, fixing its lengthscale to 24 hours seems appropriate.

To determine the optimal values of the free hyperparameters, the maximum likelihood estimation is used. Given the observations x_1, x_2,..., x_n and the covariance function/statistical model $f(x)$, the joint density function for all observations is defined as

$$f(x_1, x_2, ..., x_n) = f(x_1 \mid \theta) \times f(x_2 \mid \theta) \times ... \times f(x_n \mid \theta). \tag{4.12}$$

By looking at this equation from a new perspective, the likelihood

$$\mathcal{L}(\theta; x_1, x_2, ..., x_n) = f(x_, x_2, ..., x_n \mid \theta) = \prod_{i=1}^{n} f(x_i \mid \theta) \tag{4.13}$$

is a function of the hyperparameters θ. Therefore the observations x_i are fixed parameters. Thus, the optimal hyperparameters θ^* can be

derived from

$$\theta^* = \arg\max_{\theta} \quad \mathcal{L}(\theta; x_1, x_2, ..., x_n). \tag{4.14}$$

In practice, often the natural logarithm is used because this simplifies the computation of the optimal hyperparameters while giving the same results. Fig. 4.1 shows the importance of the choice of the hyperparameters. A very short lengthscale leads to massive uncertainties of the predictions.

The previous sections explained what GPs are and how they are completely described by the mean function and covariance function. Moreover, the optimization of the hyperparameters and their effect on the predictions were illustrated.

MPC and GP build the theoretical base for the augmented MPC scheme provided in this work. The following chapter derives the augmented MPC by fusing predictive and learning-based control.

Chapter 5

Fusing predictive and learning-based control

The implementation of the by GPs augmented MPC scheme is made up of three major steps. First, MPC and GPs are conceptionally combined in an augmented MPC scheme in Section 5.1. Second, a linear model for the prediction was derived in Section 5.2. Third, in Section 5.3 the software used for the implementation is highlighted.

In this chapter the time-dependency is not explicitly written due to better readability.

5.1 Augmented controller

The purpose of the augmentation is to control a nonlinear process (5.1) with a MPC scheme based on a linear prediction model (5.2).

$$\dot{\mathbf{x}}_{nl} = f\left(\mathbf{x}, \mathbf{u}, \mathbf{d}\right), \tag{5.1}$$

$$\dot{\mathbf{x}}_{lin} = A\left(\mathbf{d}\right)\mathbf{x} + B\left(\mathbf{d}\right)\mathbf{u} + C\left(\mathbf{d}\right). \tag{5.2}$$

$\mathbf{x}$ denotes the states, $\mathbf{u}$ the inputs and $\mathbf{d}$ the disturbance variables which are treated as parameters. The linear model is expanded with a term estimating the error $\mathbf{w}$:

$$\dot{\mathbf{x}}_{aug} = A\left(\mathbf{d}\right)\mathbf{x} + B\left(\mathbf{d}\right)\mathbf{u} + C\left(\mathbf{d}\right) + \mathbf{w}. \tag{5.3}$$

The error estimator $\mathbf{w}$ is described by a multidimensional Gaussian process

$$\mathbf{w}\left(\mathbf{z}\right) \sim \mathcal{GP}\left(m\left(\mathbf{z}\right), k\left(\mathbf{z}, \mathbf{z}'\right)\right), \tag{5.4}$$

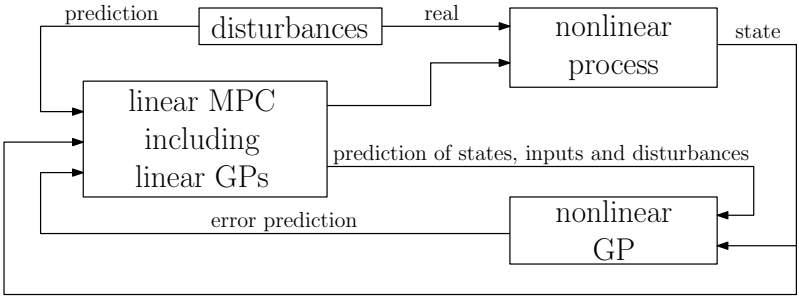


Figure 5.1: Concept of the augmented MPC

where $\mathbf{z}$ is the concatenation of $\mathbf{x}$, $\mathbf{u}$ and $\mathbf{d}$.

The training data of the GPs is obtained by computing the error with 5.5 for a sufficient number of training points $\mathbf{z}_t$ of the input space.

$$\mathbf{w}\left(\mathbf{z}_t\right) = \dot{\mathbf{x}}_{nl}\left(\mathbf{x}_t, \mathbf{u}_t, \mathbf{d}_t\right) - \dot{\mathbf{x}}_{lin}\left(\mathbf{x}_t, \mathbf{u}_t, \mathbf{d}_t\right). \tag{5.5}$$

After the GPs are optimized with the training data, they are added to the linear model.

In our concept two possible implementations are distinguished. If the GP is linear, it is included directly into the optimization. Otherwise it is treated as a parameter precomputed based on the predicted state and input trajectory of the previous optimization step and the known future disturbances. This implementation is similar to a look-up table. This distinction is done to keep the MPC scheme linear because of the advantageous presented in Chapter 1. The whole concept is depicted in Fig. 5.1.

The full augmented MPC description is given by:

$$\min J(\mathbf{x}, \mathbf{u}; \mathbf{d}, \mathbf{w}) \tag{5.6a}$$

subject to:

$$\dot{\mathbf{x}}_{aug} = A\left(\mathbf{d}\right)\mathbf{x} + B\left(\mathbf{d}\right)\mathbf{u} + C\left(\mathbf{d}\right) + \mathbf{w}, \tag{5.6b}$$

$$\mathbf{x} \in \mathcal{X}, \mathbf{u} \in \mathcal{U}, \tag{5.6c}$$

and the optimal input trajectory $\mathbf{u}^*$ is obtained by:

$$\mathbf{u}^* = \arg\min_{\mathbf{u}} J(\mathbf{x}, \mathbf{u}; \mathbf{d}, \mathbf{w}), \tag{5.7}$$

with respect to the constraints (5.6b) and (5.6c).

5.2 Prediction model

An analytical linearization of the nonlinear model described in **??** was used as the prediction model. The nonlinear model is denoted as $f_{nl}(\mathbf{x}, \mathbf{u}; \mathbf{d})$. The linearization at the point $(\mathbf{x}_0, \mathbf{u}_0)$ is described by:

$$\begin{aligned} f_{lin}(\mathbf{x}, \mathbf{u}; \mathbf{d}) = f_{nl}(\mathbf{x}_0, \mathbf{u}_0; d) \\ + \left.\frac{\partial f_{nl}(\mathbf{x}, \mathbf{u}; \mathbf{d})}{\partial x}\right|_{\mathbf{x}_0, \mathbf{u}_0} \cdot (\mathbf{x} - \mathbf{x}_0) \\ + \left.\frac{\partial f_{nl}(\mathbf{x}, \mathbf{u}; \mathbf{d})}{\partial u}\right|_{\mathbf{x}_0, \mathbf{u}_0} \cdot (\mathbf{u} - \mathbf{u}_0). \end{aligned} \tag{5.8}$$

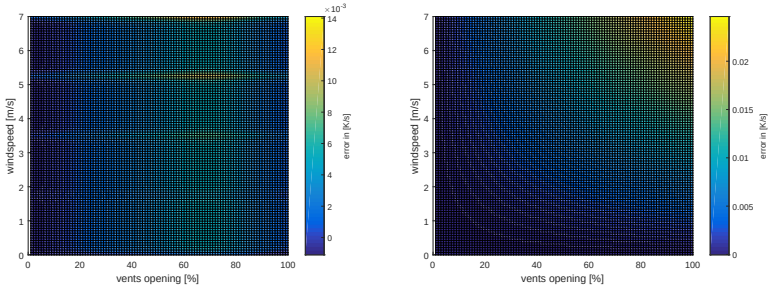
5.8 can also be expressed as a parameter-dependent state-space model 5.2. The parameters for the linearization are given in Appendix A.2.

The linearisation is still parameter-dependent because of the bilinear properties of $f_{nl}(\mathbf{x}_0, \mathbf{u}_0; d)$. Examining a simplified version of 2.3, described by:

$$Q_{rs,sim} = D_{rs,e} \cdot (1 - U_{shd}), \tag{5.9}$$

we can see that the magnitude of the effect of the shades U_{shd} is heavily depending on the solar radiation $D_{rs,e}$. At night ($D_{rs,e} = 0$) the shades have no effect while at noon ($D_{rs,e} \approx 800^{W}/m^2$) they may be the only way to avoid too high temperatures. This is also the reason why prediction models derived from step and impulse responses perform badly when the environmental variables differ from the test scenario.

Another aspect of our prediction model is the error prediction with GPs. Taking every variable into account would result in an input space with eleven dimensions. To reduce the size of the GP identifying the major error sources is necessary. The wind speed has a major effect on the greenhouse inside air temperature. Hence, establishing



(a) Before optimization (b) After optimization

Figure 5.2: GP describing the error for the temperature ODE caused by the linearization in dependence of U_{ven} and D_{ws}

a GP for the wind speed $D_{ws,e}$ which is bilinearly and nonlinearly affected with opening percentage of the vents U_{ven} results in an huge improvement of the linear model as Chapter 6 will show.

Fig. 5.2 shows the GP of the error of the temperature ODE. Before the optimization of the hyperparameters the error seems arbitrary. After the optimization a clear pattern is identifiable. The greater the wind speed and the vents opening the greater the error. Since the system is linearized around $(U_{ven} = 0.1, D_{ws,e} = 0)$ the error growth in dependence of the distance to this point seems valid.

5.3 Software

The MPC was implemented via the use of several frameworks for Python 2.7. **do-mpc** [13], a nonlinear MPC framework with integrated optimization, is the central element. It consists of four modules: model, optimizer, observer and simulator. Due to this modularity switching between different setups, e.g. changing the prediction model, is simple and analysis is simplified.

For the optimization tasks **Ipopt** [24] is employed. It is a primal-dual interior-point algorithm with a filter line-search method for nonlinear programming including second order corrections. **CasADi** [1]

do-mpc

model	optimizer	simulator
CasADi	CasADi + Ipopt + GPy	CasADi

Figure 5.3: Relations between the software tools

is a symbolic software tool for automatic differentiation and dynamic optimization used for the modeling and for the computation of first and second order derivative information. **GPy** [6] is the Gaussian process framework utilized for the augmented MPC. It includes a multitude of models, kernels and likelihoods for modelling stochastic processes and computing their optimal parameters. Fig. 5.3 recaps the relations between the single software tools. The module observer is not mentioned explicitly because full state feedback was used.

In summary, this chapter illustrated the concept of the expanded MPC approach. Moreover, the linear prediction model used for the approach was derived and the software tools to implement the new concept were described. In the following two chapters the newly derived augmented MPC scheme is used for set-point tracking (Chapter 6) and economic MPC (Chapter 7).

Chapter 6

Set-point tracking

The ability to track a static set-point is tested because it gives a measure to compare and quantify the quality of different MPC schemes. Furthermore, the ability of rejecting disturbances can be examined clearly. The progress of the disturbance variables is presented in Fig. 6.1. The data is obtained from the meteorological station Weidling in Austria [23].

The chapter is divided into three sections. Section 6.1 describes the optimal control problem explicitly. Section 6.2 shows three tests illustrating certain properties of conventional MPC and the augmented MPC scheme. Section 6.3 compares the quality of three different prediction models respective to set-point tracking with noise-free data.

Within this chapter four MPC approaches were tested. The approaches vary in the prediction models and if GPs are implemented. Moreover, the kind of kernels used to describe the GPs is distinguished. The nomenclature of the approaches is defined in Table 6.1.

Description	Prediction model	Gaussian processes
NMPC	Nonlinear	-
LMPC	Linear	-
LMPC+LGP	Linear	Linear kernels
LMPC+GP	Linear	Nonlinear kernels

Table 6.1: Nomenclature of the MPC approaches

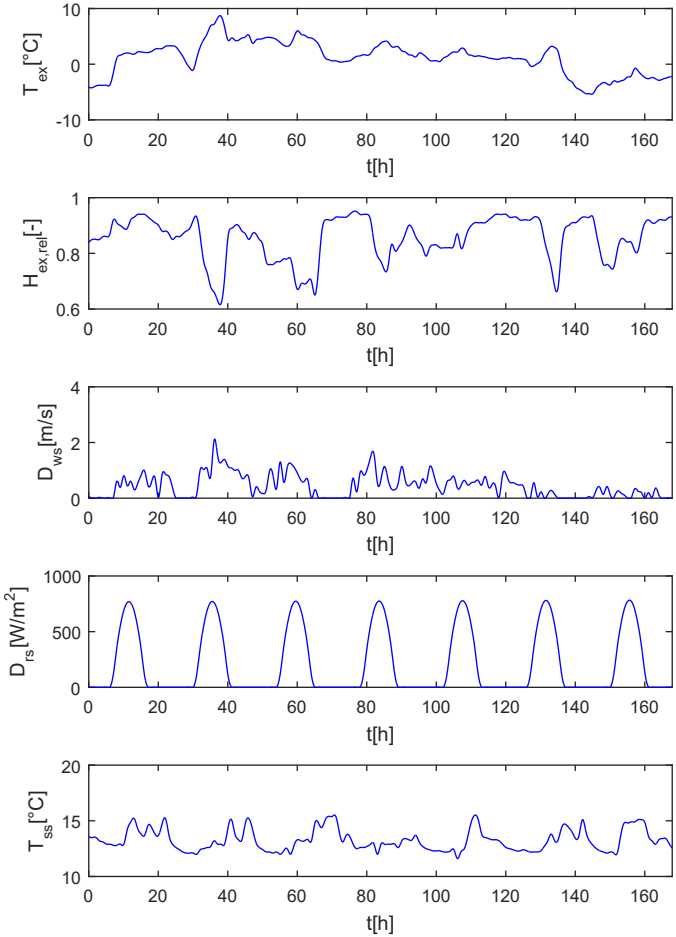


Figure 6.1: Progress of the disturbance variables

6.1 Preliminaries

For the optimal growth of tomato crop an air temperature of 25°C and an relative air humidity of 65% are assumed. That equals at the

given temperature an absolute air humidity of 0.015g water per cubic metre of air. The quadratic objective function used was:

$$J_{sp} = 200 \cdot (25.0 - X_{T,a})^2 + 100 \cdot (0.015 - X_{H_{a,a}})^2, \tag{6.1}$$

which enforces temperature tracking more than humidity tracking of the previously mentioned optimal conditions.

The full description of the set-point tracking MPC is given by:

$$\min \int_0^{t_f} J_{sp} \, \mathrm{d}t \tag{6.2a}$$

subject to:

$$\dot{\mathbf{x}}_{pm}(t) = f(\mathbf{x}(t), \mathbf{u}(t); \mathbf{d}(t)), \tag{6.2b}$$

$$\mathbf{u}(t) \in \mathcal{U}, \tag{6.2c}$$

where $\dot{\mathbf{x}}_{pm}$ is the model equation of the corresponding MPC approach. Since set-point tracking shows the ability to keep a system close to the desired state no state constraints were defined is it can be seen in Table 6.2.

quantities	lower bound	upper bound	unit
$X_{T,a}$	-	-	$^\circ\mathrm{C}$
$X_{H_{a,a}}$	-	-	$\mathrm{kg/m^3}$
U_{ven}	1	100	$\%$
U_{shd}	0	0.7	$-$
U_{heat}	0	1000000	$\mathrm{W/m^2}$
U_{hum}	0	0.2	$^{kg}/_{\mathrm{m^5 \cdot s}}$

Table 6.2: Constraints of the controlled variables $X_{(\cdot)}$ and manipulated variables $U_{(\cdot)}$.

6.2 Pretests

A deep analysis of LMPC+GP was done on noise-free predictions in Section 6.3 because noise-free predictions enable explicitly to quan-

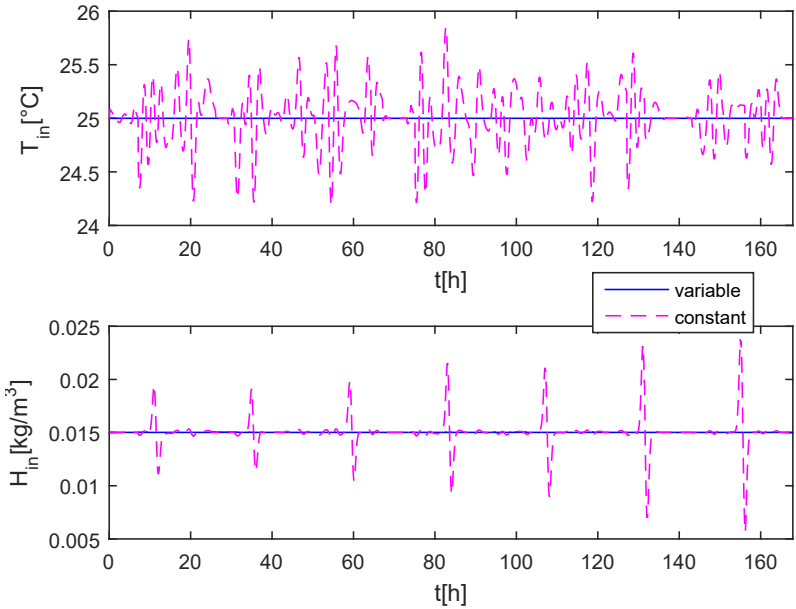


Figure 6.2: Comparison of NMPC with variable and constant disturbance prediction

tify the profit of the LMPC+GP scheme. In this section the three aspects disturbance prediction, error estimation with linear kernels and performance with noisy disturbances are emphasized.

Stressing MPC considering the weather forecast is motivated by being able to reject the effect of disturbance variables. This is illustrated in Fig. 6.2. The settings of the both pictured NMPC approaches differ only in the prediction of the disturbance variables. Including variable and exact predictions in the computations enables the control to reject disturbances in advance and to stick very close to the set-point. Whereas assuming constant disturbance values over the prediction horizon leads to major deviations since the control has to react to the current disturbances.

Due to the fact that GPs with linear kernels can be directly included

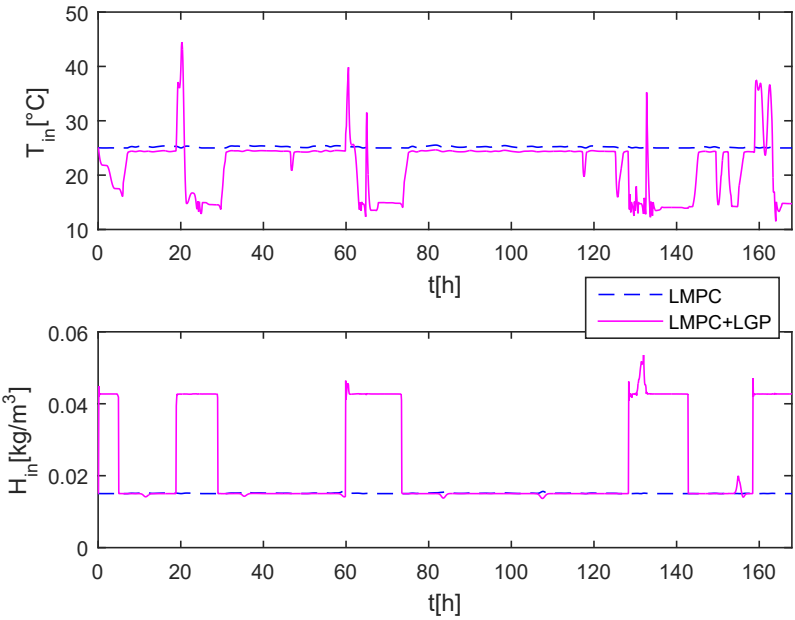


Figure 6.3: Comparison of augmented MPC with solely linear kernels and linear MPC

into the optimization of a linear MPC scheme, this setup denoted as LMPC+LGP was tested. But the linear kernels were not able to model the error as Fig. 6.3 depicts. When the input variables for the GP were in a scope where the linear kernels modelled the error well, the control quality of LMPC+GP was similar to LMPC (e.g. at $t \approx 100h$). But for extrapolating outside their scope the linear kernels led to massive deviations from the set-point (e.g. at $t \approx 65h$). Since the error is the by the linearisation neglected nonlinear dynamics it inherits the nonlinear characteristics. Thus, nonlinear approach is required to model the error properly. Hence, the approach with linear kernels was dropped.

Since signals from sensors are noisy and the weather forecast is non perfect the application of MPC in a real greenhouse has to handle

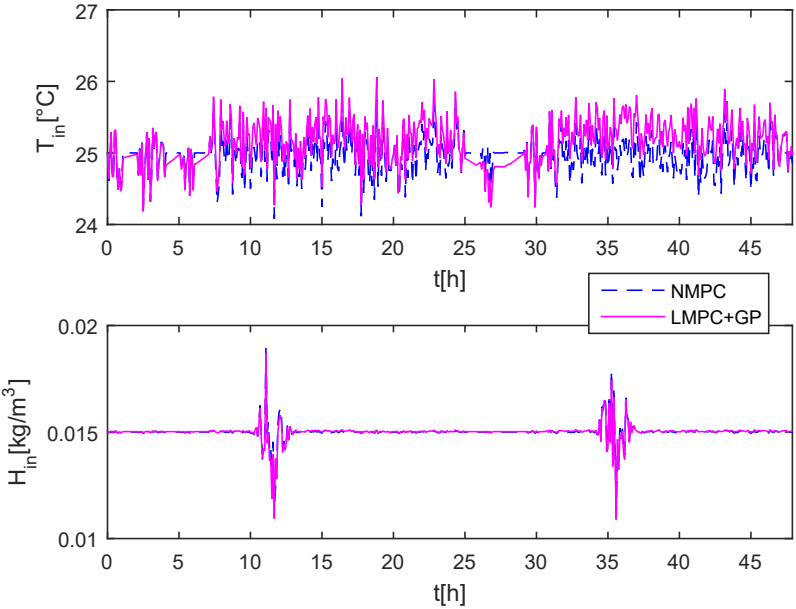


Figure 6.4: Comparison of nonlinear and augmented MPC with noisy predictions

the outcomes. Thus, the ability of LMPC+GP to cope with noisy predictions was checked to simulate a more realistic scenario. Fig. 6.4 depicts the results for NMPC and LMPC+GP in which NMPC serves as the benchmark. Due to the non perfect predictions MPC is not able to compute the optimal input. Therefore, the state trajectories are chattering around the set-points. Since both approaches are able to track the set-points and the results of NMPC and LMPC+GP are matchable it is shown that LMPC+GP is able to handle noisy predictions.

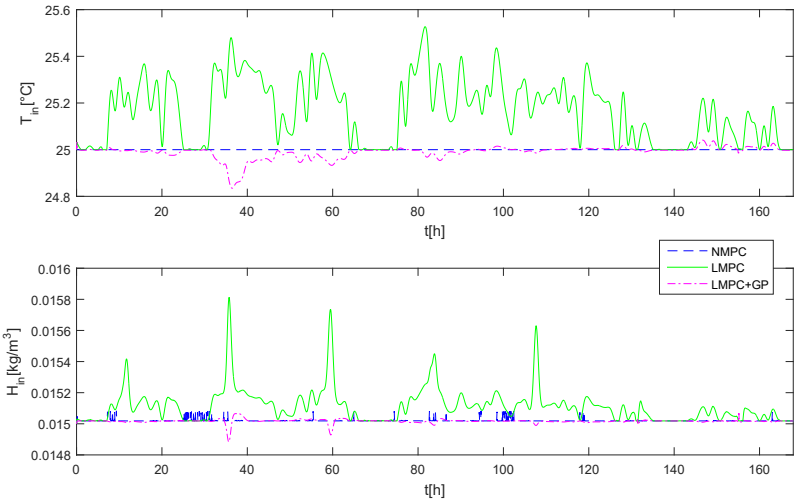


Figure 6.5: State trajectories

6.3 Noise-free tracking

The comparison of NMPC, LMPC and LMPC+GP was done on noise-free predictions. Thus, the benefit of LMPC+GP can be clearly ascertained eliminating the random effect of noise. For all three schemes the prediction horizon was eight hours and the sample period was fixed to five minutes. The aim of achieving a performance with a linear MPC scheme similar to a NMPC scheme was reached.

The progress of the state trajectories is depicted in Fig. 6.5. NMPC is the benchmark for LMPC and LMPC+GP as it uses the nonlinear prediction model which is identical to the simulation model portrayed in **??**. Therefore, the results are optimal with respect to the objective function (6.1).

LMPC and LMPC+GP use the same prediction model except for the error term added in LMPC+GP. By estimating the error with GPs LMPC+GP achieves a diminishing of the square root tracking error by more than 86 % for both temperature and humidity. Also the maximum deviation which can be critical concerning state constraints

could be reduced by 68 % for the temperature and by 85 % for the humidity. The smaller root mean square humidity tracking error of LMPC+GP compared to NMPC is due to the objective function. Since deviations of the temperature are weighted heavier small deviations of the humidity are accepted. Table 6.3 summarizes the results.

quantities		NMPC	LMPC	LMPC+GP
maximum	temp. [°C]	$3.6 \cdot 10^{-4}$	0.5273	0.1651
deviation	hum. [kg/m³]	$8.3 \cdot 10^{-5}$	$8.1 \cdot 10^{-4}$	$1.2 \cdot 10^{-4}$
root mean	temp. [°C]	$1.3 \cdot 10^{-5}$	0.2126	0.0294
square error	hum. [kg/m³]	$2.2 \cdot 10^{-5}$	$1.6 \cdot 10^{-4}$	$2.1 \cdot 10^{-5}$

Table 6.3: Root mean square error and maximum deviation

Chapter 7

Economic model predictive control

Static set-point tracking of the optimal leads to maximal crop growth in the greenhouse. But such an implementation is neither market-driven nor efficient.

Since there is often a region of optimal parameters yet another MPC approach is presented. Assuming a region of optimal greenhouse temperature from 24 °C to 26 °C, operating the greenhouse at the lower bound requires less heating energy. The quadratic objective function of conventional MPC is not able to involve this fact.

Furthermore, renewable energy sources, customer stimulus and many other effects produce regularly changing energy prices which are depicted in Fig. 7.1. The intraday of the electricity tariff was generated covering the major characteristics of the energy market. From 0 a.m. to 4 a.m. when most people are asleep energy is the cheapest while in the early evening when people return from work energy is the most expensive.

One possibility to cope with the requirements of the modern energy market is economic model predictive control (EMPC). An overview of approaches to economic process optimization is provided in [18], [22] and [5]. First, the idea of EMPC is presented in Section 7.1. Then EMPC is applied to the operation of the greenhouse in Section 7.2. In Section 7.3 the results of set-point tracking MPC and EMPC are compared using the augmented model predictive derived in Section 5.1.

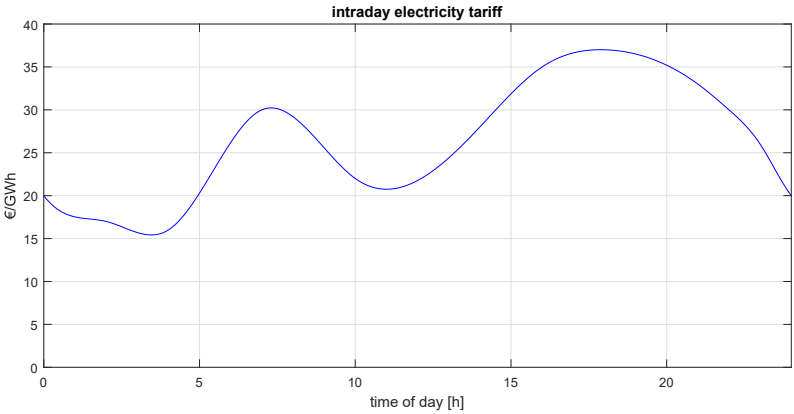


Figure 7.1: Intraday of the electricity tariff

7.1 Preliminaries

EMPC is an off steady-state dynamic process optimization scheme. Instead of tracking a set-point optimal regions and an economic objective function are defined. The aim of EMPC is to optimize the economically efficient operation of a process by maximizing the profit or minimizing the costs. Often the optimal point for operating the process with respect to the objective function is at the limits.

For the greenhouse the state constraints limit the region of parameters for optimal crop growth. The state constraints are implemented as soft constraints. Soft constraints avoid infeasibility for small violations of the limits. Whereas the input constraints are hard constraints because they are physical limits of the actuators. The constraints of the variables are summarized in Table 7.1.

Since heating is the main cost for operating a greenhouse in the winter the objective function for EMPC is described by:

$$J_{eco} = U_{heat} \cdot t_{step} \cdot p_E \cdot c_{area,ss} + p_W \cdot U_{hum}, \tag{7.1}$$

where p_E is the current electricity tariff and p_W is a penalty for water consumption. Those possibly time-variable parameters are denoted

quantities	lower bound	upper bound	unit
$X_{T,a}$	24	26	°C
$X_{H_{a,a}}$	0.01	0.02	$\mathrm{kg/m^3}$
U_{ven}	1	100	%
U_{shd}	0	0.7	–
U_{heat}	0	1000000	$\mathrm{W/m^2}$
U_{hum}	0	0.2	$^{kg}/_{\mathrm{m^5 \cdot s}}$

Table 7.1: Constraints of the controlled variables $X_{(\cdot)}$ and manipulated variables $U_{(\cdot)}$.

as $\mathbf{d}_{eco}(t)$. The first term gives the cost in € for each computation step. The second term is added to minimize the water usage for humidification.

The description of EMPC is very similar to conventional MPC (3.6) and to set-point tracking MPC (6.2). The economic optimal control problem is fully described by:

$$\min \int_0^{t_f} J_{eco}(\mathbf{x}(t), \mathbf{u}(t); \mathbf{d}_{eco}(t))\, \mathrm{d}t \tag{7.2a}$$

subject to:

$$\dot{\mathbf{x}}_{pm}(t) = f(\mathbf{x}(t), \mathbf{u}(t); \mathbf{d}(t)), \tag{7.2b}$$

$$\mathbf{x}(t) \in \mathcal{X},\ \mathbf{u}(t) \in \mathcal{U}. \tag{7.2c}$$

In the following two sections three different MPC approaches are compared. The differences between the approaches and their denomination is given in Table 6.1.

7.2 Economic climate control

For the analysis of EMPC the same scenario was used as for set-point tracking (Fig. 6.1). The figures were limited to 72 hours due to better

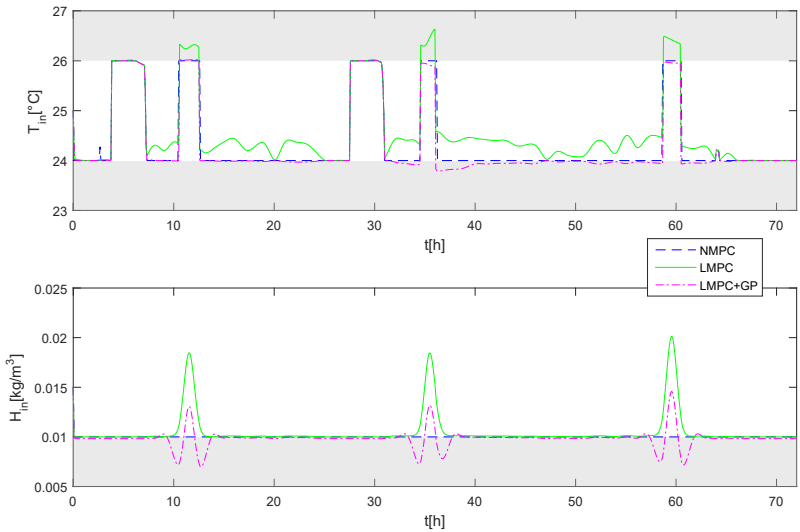


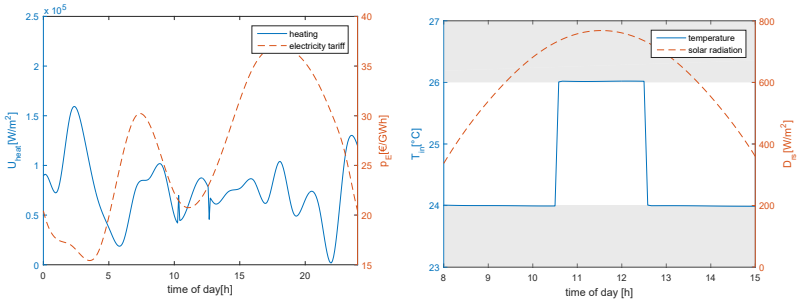
Figure 7.2: State trajectories of EMPC

visualisation. NMPC was used as the benchmark since it includes a perfect prediction model.

Fig. 7.2 shows that LMPC+GP enhances the quality of EMPC compared to LMPC. The temperature trajectory of LMPC+GP stays mostly close to the one of NMPC while LMPC is often deviated. Thus, the root mean square error for the temperature could be reduced by 47 %. This results in a reduction of costs for operating the greenhouse by 1.5 % while providing better crop growth.

In most of the cases there are two peaks for the air temperature within every 24 hours where the temperature is driven to the upper bound. Those peaks are due to capitalizing the influences of time-varying parameters and disturbances. In those situations the air serves as a storage for cheap energy.

The first peak of the day is dedicated to more heating due to low energy prices. The left pane of Fig. 7.3 shows this relation in detail. When the price for energy is the lowest ($t \leq 4\,\mathrm{h}$) the heating reaches



(a) Electricity tariff (b) Solar radiation

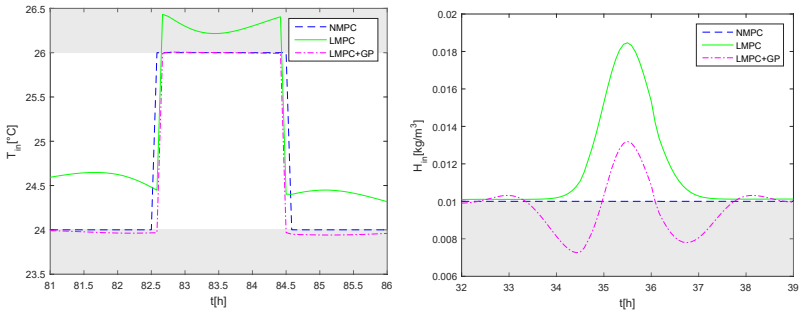
Figure 7.3: Effect of external variables on EMPC with LMPC+GP

its highest values. Throughout the day the heating is significantly lower since the energy is more expensive.

The second peak is dedicated to the gratuitous energy delivered by the sun. Around noon when the solar radiation reaches its peak the air temperature inside the greenhouse rises to the upper bound. This effect is depicted on the right pane of Fig. 7.3.

The left pane of Fig. 7.4 shows a detail of the temperature trajectory around noon ($t = 84\,\mathrm{h}$). Due to underestimating the impact of the solar radiation LMPC is deviated by trend to higher temperatures. This leads to violations of the upper bound. Due to the GPs LMPC+GP is able to estimate the nonlinear effect of the solar radiation. Therefore, the maximum violations of the temperature constraints could be reduced by 63 %.

The humidity trajectories are depicted in Fig. 7.2. Since the humidification system is the main source for the inside air humidity, operating the system at the lower bound minimizes the water usage. The solar radiation is the main influence for the crop transpiration. Since the effect is nonlinear the control of LMPC and LMPC+GP gets worse the bigger the solar radiation becomes. The right pane of Fig. 7.4 shows the humidity trajectories from 8 a.m. to 3 p.m. in detail. LMPC+GP stays closer to the to optimal trajectory of NMPC than LMPC. The GPs enable LMPC+GP to compensate the nonlinear effect while LMPC gets heavily deviated. Hence, the root mean



(a) Temperature (b) Humidity

Figure 7.4: Details of the state trajectories

square error of LMPC+GP is reduced by 43 % compared to LMPC. The violations of the lower bound by LMPC+GP are not critical since the correct temperature is more important for crop growth.

The input trajectories are presented in Fig. 7.5. For all three MPC approaches the trajectories are very similar. The inputs do not reach their upper limits and their progress is smooth. This means both linear prediction models compute reasonable control.

The input trajectories of ventilation U_{ven} and the shades U_{shd} exhibit several flat areas. Both manipulated variables are bilinearly related with disturbance variables. The effect of the bilinearity was explained using the example of (5.9). When the disturbances are zero the inputs have no impact. Thus, they are set to the middle of their physical limits.

7.3 Comparison of set-point tracking MPC and EMPC

The definitions of set-point tracking MPC (6.2) and EMPC (7.2) are very similar. However, the choice of the objective function and the state constraints have a huge impact on the performance of MPC. For the analysis in this section exclusively the expanded MPC scheme

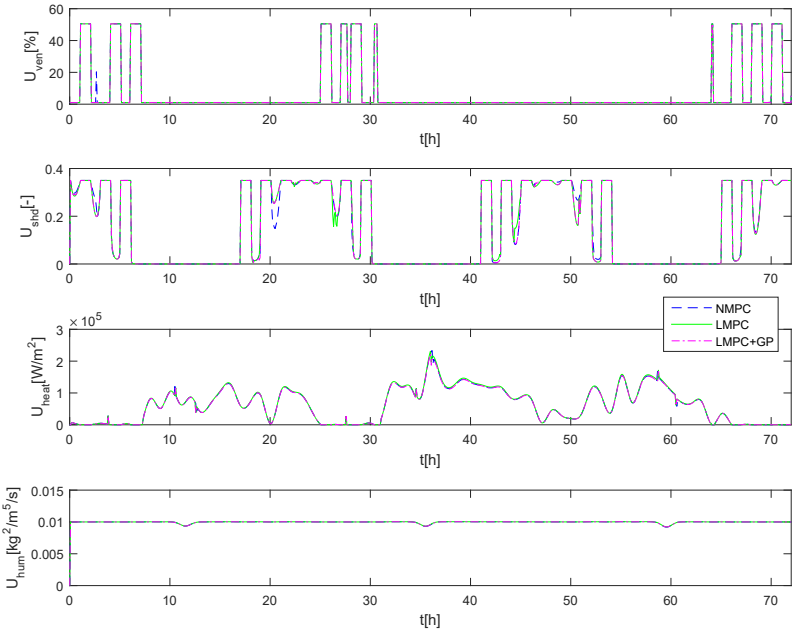


Figure 7.5: Input trajectories of EMPC

(LMPC+GP) was used.

Table 7.3 shows the operating effort of the greenhouse using the control strategies presented Chapter 6 and Chapter 7. Applying EMPC leads to significant savings concerning operating costs compared to set-point tracking MPC, as Fig. 7.6shows.

Due to the consideration of the intraday of the electricity tariff the cost was diminished by 5.54 %. This means every hour more than 6 € can be saved on average. The energy used for heating the greenhouse was decreased by 5.32 %. Also the consumption of water for humidification could be slightly reduced by 0.15 %. Fig. 7.7 illustrates the savings more detailed. On the left the progress of the cost and on the right the progress of the humidification is depicted.

In the current chapter the control strategy EMPC was presented.

quantities		NMPC	LMPC	LMPC+GP
maximum violation	temp. [°C]	-	0.60	0.22
	hum. [kg/m³]	-	-	$2.84 \cdot 10^{-3}$
root mean square error	temp. [°C]	-	0.2815	0.1486
	hum. [kg/m³]	-	0.0024	0.0014
mean	U_{heat} [W/m²]	$5.619 \cdot 10^4$	$5.705 \cdot 10^4$	$5.605 \cdot 10^4$
	U_{hum} [kg²/m⁵·s]	0.01	0.01	0.01
	cost [€/h]	109.35	111.04	109.09

Table 7.2: Comparison of the EMPC schemes

First EMPC was derived and stated. Afterwards, it was shown that LMPC+GP performs better than LMPC and similar to NMPC applied to EMPC. The current section presented the profit EMPC compared to set-point tracking MPC using LMPC+GP as the prediction model. In the concluding chapter the outcomes of this work are summarized and reviewed. Furthermore, directions for future works are provided.

quantities	unit	set-point tracking	economic	savings
mean U_{heat}	[W/m²]	$5.9205 \cdot 10^4$	$5.6051 \cdot 10^4$	5.32 %
mean U_{hum}	[kg²/m⁵·s]	$99.731 \cdot 10^{-3}$	$99.561 \cdot 10^{-3}$	0.15 %
mean cost	[€/h]	115.50	109.09	5.54 %

Table 7.3: Comparison of operating effort of the greenhouse

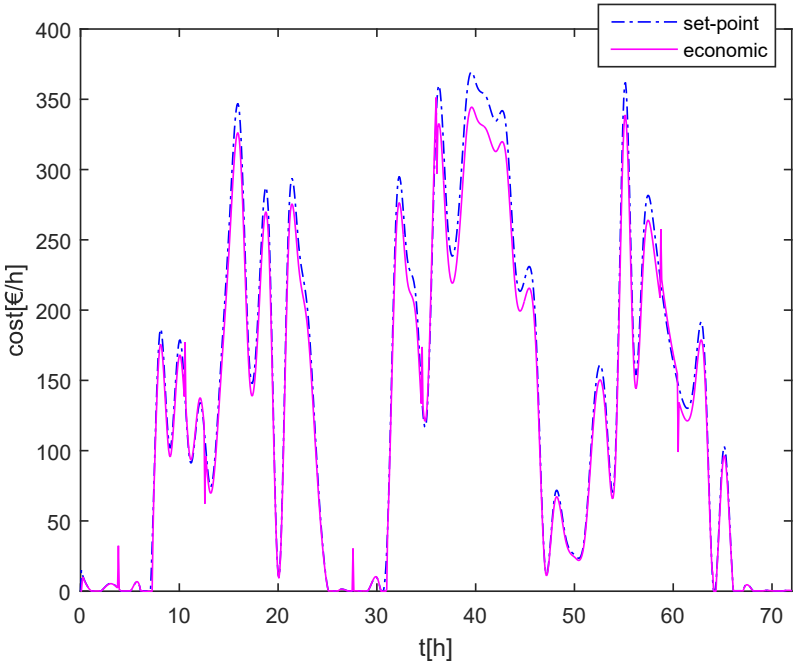
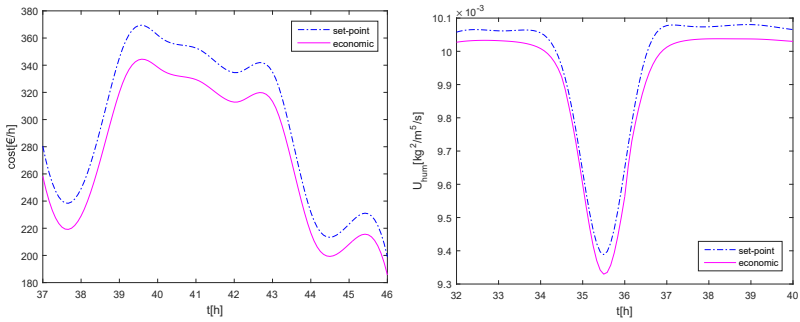


Figure 7.6: Progress of the operating cost



(a) Operating cost (b) Humidification

Figure 7.7: Detailed comparison of the operating effort

Chapter 8

Conclusion

Building climate control is important due to provide the optimal conditions, may it be for working inside an office building or for crop growth. Moreover, building climate control gives the possibility to reduce the costs for operating buildings and to reduce the consumption of energy.

For building climate control the best results are often achieved with nonlinear model predictive control because the climate dynamic is highly complex and nonlinear. Since obtaining a nonlinear prediction model is an time-consuming task and requires expertise, the work presented an augmented linear model predictive control scheme providing similar results to nonlinear model predictive control. The error due to neglecting the nonlinear dynamics with a linear prediction model is estimated with Gaussian processes and included in the computations as a parameter.

It was shown using the example of a greenhouse that error estimation with Gaussian processes improves the quality for the control significantly. For set-point tracking the root mean square error was reduced by nearly one order of magnitude. Applying the augmentation in an economic model predictive control scheme resulted in lower costs for operating the greenhouse while providing better conditions for the crop growth. Thus, the performance of nonlinear model predictive control and linear model predictive control with error estimation via Gaussian processes are comparable. Furthermore, the advantages of economic model predictive control were demonstrated. Controlling the greenhouse climate with economic model predictive control resulted in significant cost savings compared to set-point tracking model predictive control.

One possible improvement for the by Gaussian processes extended linear model predictive control scheme is the estimation of periodic occurring errors as it was shown in [11]. Since disturbance variables like the solar radiation are linked to the day-night-cycle a Gaussian process with periodic kernel could model this error. On-line optimization of the hyperparameters would allow adaption to the current weather situation.

Another direction for further developments is the application of the augmented linear model predictive control scheme on embedded systems. This is possible due to the low computational requirements compared to nonlinear model predictive control. For the implementation **μAO-MPC**, a code generation software package for linear model predictive control, is proposed. For more details see [25].

Finally, [15] presents a more systematic approach to obtain Gaussian process models. The influence of all input dimensions and the relations between those can be gradually examined. By dropping input dimensions and relations with little impact the size of the Gaussian process model is decreased and the interpretability is increased. Thus, the computational expense of the Gaussian process models is reduced.

List of Figures

2.1 Task space 6
2.2 Wheel structure 7

3.1 Example of a time-discrete MPC scheme: red - reference trajectory, light green - measured state trajectory, dark green - predicted state trajectory, violet - optimal control trajectory 12

4.1 Impact of the hyperparameters on the predictions of a GP 20

5.1 Concept of the augmented MPC 24
5.2 GP describing the error for the temperature ODE caused by the linearization in dependence of U_{ven} and D_{ws} . . 26
5.3 Relations between the software tools 27

6.1 Progress of the disturbance variables 30
6.2 Comparison of NMPC with variable and constant disturbance prediction 32
6.3 Comparison of augmented MPC with solely linear kernels and linear MPC 33
6.4 Comparison of nonlinear and augmented MPC with noisy predictions 34
6.5 State trajectories 35

7.1 Intraday of the electricity tariff 38
7.2 State trajectories of EMPC 40
7.3 Effect of external variables on EMPC with LMPC+GP . 41
7.4 Details of the state trajectories 42
7.5 Input trajectories of EMPC 43
7.6 Progress of the operating cost 45

7.7 Detailed comparison of the operating effort 45

List of Tables

6.1 Nomenclature of the MPC approaches 29
6.2 Constraints of the controlled variables $X_{(\cdot)}$ and manipulated variables $U_{(\cdot)}$. 31
6.3 Root mean square error and maximum deviation . . . 36

7.1 Constraints of the controlled variables $X_{(\cdot)}$ and manipulated variables $U_{(\cdot)}$. 39
7.2 Comparison of the EMPC schemes 44
7.3 Comparison of operating effort of the greenhouse . . . 44

A.1 Parameters of the greenhouse model 58
A.2 Parameters for the linearization 58

Bibliography

[1] J. Andersson. *A General-Purpose Software Framework for Dynamic Optimization*. Phd thesis, Arenberg Doctoral School, KU Leuven, Department of Electrical Engineering (ESAT/SCD) and Optimization in Engineering Center, Kasteelpark Arenberg 10, 3001-Heverlee, Belgium, 01.01.2013.

[2] A. Dietrich, T. Wimböck, A. Albu-Schäffer, and G. Hirzinger. Singularity avoidance for nonholonomic, omnidirectional wheeled mobile platforms with variable footprint. In *2011 IEEE International Conference on Robotics and Automation*, pages 6136–6142. IEEE, 2011.

[3] M. Ebden. Gaussian processes for classification: A quick introduction, .

[4] M. Ebden. Gaussian processes for regression: A quick introduction, .

[5] M. Ellis, J. Liu, and P. D. Christofides. *Economic Model Predictive Control: Theory, Formulations and Chemical Process Applications*. Advances in Industrial Control. Springer-Verlag, s.l., 1. aufl. edition, 2016. ISBN 3319411071.

[6] GPy. Gpy: A gaussian process framework in python, since 2012.

[7] A. Grancharova and T. A. Johansen. *Explicit nonlinear model predictive control: Theory and applications*, volume 429 of *Lecture Notes in Control and Information Sciences*. Springer, Berlin, 2012. ISBN 9783642287800.

[8] G. Gray, R. Murray-Smith, K. Thompson, and D. J. Murray-Smith. Tutorial example of gaussian process prior modelling applied to twin-tank system.

[9] L. Grüne and J. Pannek. *Nonlinear model predictive control: Theory and algorithms*. Communications and control engineering. Springer, London, 2011. ISBN 9780857295019.

[10] R. Isermann and M. Münchhof. *Identification of Dynamic Systems: An Introduction with Applications*. Advanced Textbooks in Control and Signal Processing. Springer-Verlag Berlin Heidelberg, Berlin, Heidelberg, 2011. ISBN 9783540788782. doi: 10.1007/978-3-540-78879-9.

[11] E. D. Klenske, M. N. Zeilinger, B. Schölkopf, and H. Philipp. Gaussian process-based preditive control for periodic error correction. *IEEE Transactions on Control Systems Technology*, 24 (1), 2016.

[12] L. Ljung. *System identification: Theory for the user*. Prentice Hall information and system sciences series. Prentice Hall PTR, Upper Saddle River, NJ, 2. ed., 14. printing edition, 2012. ISBN 9780136566953.

[13] S. Lucia, A. Tătulea-Codrean, C. Schoppmeyer, and S. Engell. Rapid development of modular and sustainable nonlinear model predictive control solutions. *Control Engineering Practice*, (60):51–62, 2017.

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

[15] T. A. Plate. Accuracy versus interpretability in flexible modeling: implementing a tradeoff using gaussian process models. *Behaviormetrika*, 26(1):29–50, 1999. ISSN 0385-7417. doi: 10.2333/bhmk.26.29.

[16] C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, Cambridge, Mass, 2006. ISBN 026218253X.

[17] J. B. Rawlings and D. Q. Mayne. *Model predictive control: Theory and design*. Nob Hill Publ, Madison, Wis., 2009. ISBN 9780975937709.

[18] J. B. Rawlings, D. Angeli, and C. N. Bates. Fundamentals of economic model predictive control. *Proceedings of 2012 Conference on Decision and Control*, pages 3851–361, 2012.

[19] M. Sorour, A. Cherubini, P. Fraisse, and R. Passama. Motion discontinuity-robust controller for steerable mobile robots. *IEEE Robotics and Automation Letters*, 2(2):452–459, 2016.

[20] M. Sorour, A. Cherubini, R. Passama, and P. Fraisse. Kinematic modeling and singularity treatment of steerable wheeled mobile robots with joint acceleration limits. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2110–2115. IEEE, 2016.

[21] M. Sorour, A. Cherubini, A. Khelloufi, R. Passama, and P. Fraisse. Complementary-route based icr control for steerable wheeled mobile robots. *Robotics and Autonomous Systems*, 118: 131–143, 2019.

[22] T. Tran, K.-V. Ling, and J. M. Maciejowski. Economic model predictive control - a review. *The 31st International Symposium on Automation and Robotics in Construction and Mining (ISARC 2014)*, 2014.

[23] P. Unfried and M. Unfried. Meteorological station weidling: Climate data. 2016.

[24] A. Wächter and L. T. Biegler. On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Mathematical Programming*, (106(1)):25–27, 2006.

[25] P. Zometa, M. Kögel, and R. Findeisen. muao-mpc: A free code generation tool for embedded real-time linear model predictive control. In *Proc. American Control Conference (ACC), 2013*, pages 5340–5345, Washington D.C., USA, 2013.

Appendix A

Parameters of the greenhouse models

A.1 Physical parameters

parameter	value	unit
$c_{sph,a}$	$1004.5 \cdot 10^3$	$J/\mathrm{kg \cdot K}$
$c_{den,a}$	1.2041	kg/m^3
$c_{vol,g}$	2922	m^3
$c_{area,ss}$	877	m^2
$c_{area,p}$	166.8	m^2
$c_{asw,a}$	0.456	–
$V_{tsw,g}$	1.0	–
$c_{cnv,ss-a}$	11.774	$W/\mathrm{m}^2 \cdot \mathrm{K}$
$c_{cnd-cnv,a-e}$	14.099	$W/\mathrm{m}^2 \cdot \mathrm{K}$
c_{loss}	4.2571	–
c_{leak}	0.01	m/s
α	0.0097	–
β	0.54	–
R_D	461.51	$J/\mathrm{kg \cdot K}$
D_{LAI}	3.25	–
MW_{ratio}	0.622	–

parameter	value	unit
p	101325	Pa
λ_v	$2.26 \cdot 10^6$	J/kg
c_p	$1.006 \cdot 10^3$	$J/\mathrm{kg \cdot K}$
$c_{cl,cr}$	0.2	m
$M_{loss,a-e}$	0.2	$kg^2/\mathrm{m}^5 \cdot \mathrm{s}$

Table A.1: Parameters of the greenhouse model

A.2 Parameters for the linearization

parameter	value	unit
$X_{T_{a,a,0}}$	25.0	°C
$X_{H_{a,a,0}}$	0.015	kg/m^3
$U_{ven,0}$	1.0	%
$U_{shd,0}$	0.0	–
$U_{heat,0}$	0.0	$\mathrm{W/m^2}$
$U_{hum,0}$	0.0	$kg/\mathrm{m}^5 \cdot \mathrm{s}$

Table A.2: Parameters for the linearization