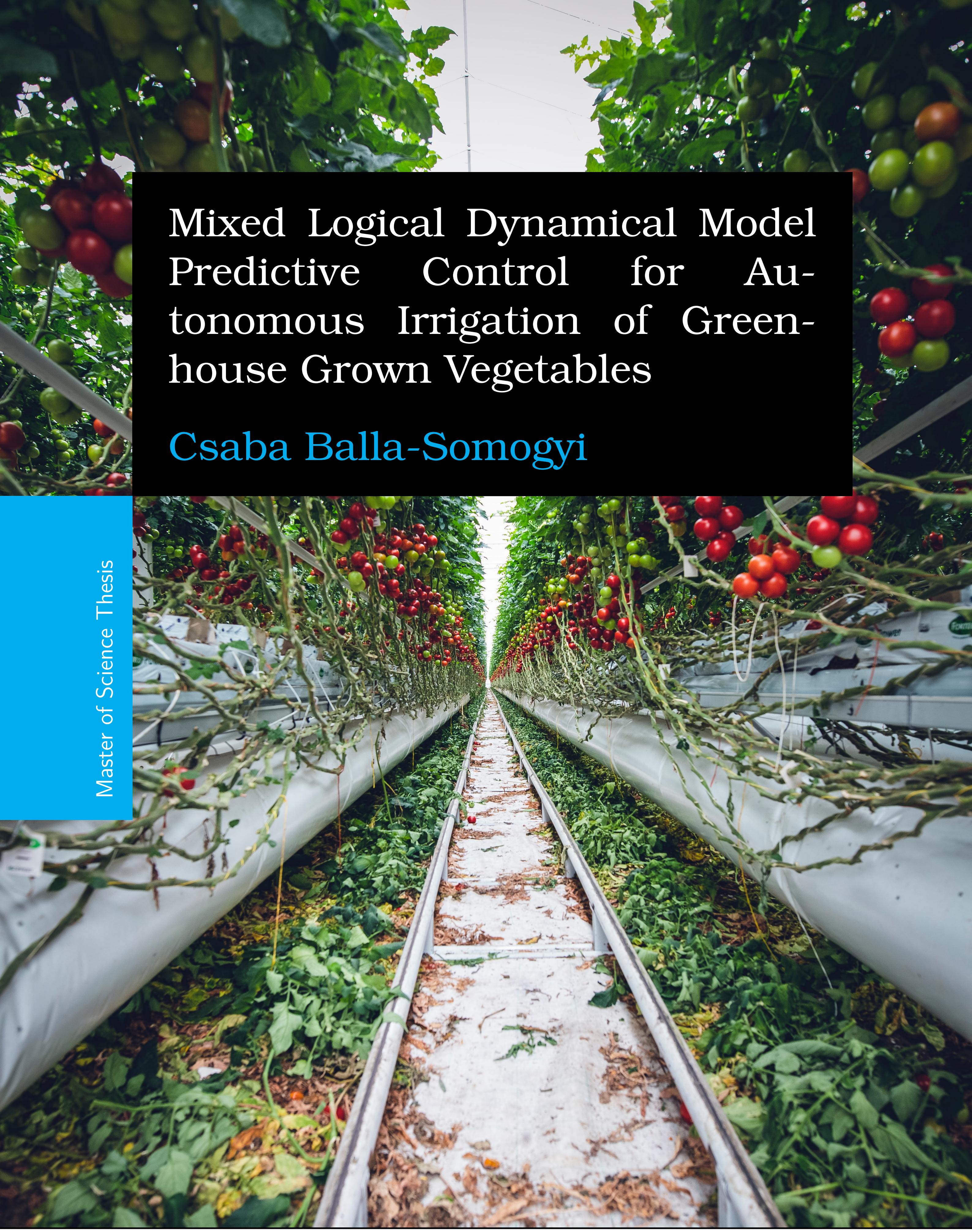

Mixed Logical Dynamical Model Predictive Control for Autonomous Irrigation of Greenhouse Grown Vegetables

Csaba Balla-Somogyi

Master of Science Thesis

Delft University of Technology

Delft Center for Systems and Control

Mixed Logical Dynamical Model Predictive Control for Autonomous Irrigation of Greenhouse Grown Vegetables

Master of Science Thesis

For the degree of Master of Science in Systems and Control at Delft University of Technology

Csaba Balla-Somogyi

August 15, 2022

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of Technology

The work in this thesis was supported by Blue Radix BV. Their cooperation is hereby gratefully acknowledged.

Abstract

Since industrialization, world population is constantly on the rise, and so the demand for nutritious and healthy food is increasing as well. Traditional agriculture is steadily extended with and substituted by crop growing in greenhouses, which has better yield, more resistance against weather extremes, and overall more control over the growing parameters than regular agriculture. Greenhouses are traditionally controlled by expert growers, but currently there are not enough experienced growers in the world, and one grower can only handle a limited number of greenhouses. Autonomous greenhouse control solves this problem by taking over the task of defining setpoints for the low-level climate controllers, so that a full crop cycle can be managed with little effort from the grower's part. Autonomous climate control in greenhouses can be extended by automatic irrigation, which reduces the workload manual calculation of irrigation decisions puts on the growers. Furthermore, it has the possibility of providing better yield than manual control during a crop cycle while reducing the water usage.

This thesis introduces a new predictive irrigation control approach, which uses forecast weather data and climate predictions to create irrigation decisions with the use of a Mixed Logical Dynamical (MLD) Model Predictive Control (MPC) algorithm. The behaviour of the controller can be changed through costs and constraints, which can be defined according to the need of greenhouse growers. The water balance model, constituting from a simple Plant Water Uptake (PWU) model from literature as well as a novel drain model, is used to predict the water content of the growing substrate accurately for 24 hours ahead. The MLD MPC exploits the structure of nonlinearities inside the water balance model to create a Mixed Integer Linear Programming (MILP) optimal control problem, which can be solved using efficient algorithms with guarantee on optimality. The validation results on data of multiple greenhouses show, that the created algorithm can be generalized with little effort. The effectiveness of the control algorithm in open-loop was inspected through simulations.

Table of Contents

List of Figures

List of Tables

List of Assumptions

Preface and Acknowledgements

This document is a part of my Master of Science graduation thesis. The idea of doing my thesis on this subject came after several long discussions with my academic supervisor, Tamas. During the meetings it became clear, that both of us are interested in horticulture, technology, and their combined impact on society. The problem of autonomous irrigation came up during a discussion with the representatives of the company Blue Radix, Rudolf and Peter. The shape of a joint graduation project with TU Delft and Blue Radix was manifested, and so my graduation project internship began in 2021 November at the company. This thesis work is the result of a long process of literature research, data analysis, model fitting and control design which I conducted with the help of both my university and Blue Radix.

I would like to express my gratitude towards my supervisors Tamas and Peter, who were always there if I had questions, with whom I could share my ideas, results and doubts during the whole project. I would like to thank the company Blue Radix and all of their employees for providing me their resources, their knowledge, and the greenhouse datasets. Without their support this thesis work would not have been created. A big thanks is due to my partner, Fanni, who helped with the sentimental aspects of the work. She helped me stay motivated during the project, and she endured all of my complaining, for which I am extremely grateful.

Delft, University of Technology — Csaba Balla-Somogyi
August 15, 2022

Chapter 1

Introduction

In 2017, around 1 out of 9 people was undernourished, which accounts for a total of about 821 million people. Between 2014 and 2017 world hunger was on the rise, reaching levels experienced 10 years before. One of the key drivers of this tendency was identified as the increasing amount of weather extremes, and its effect on food production [1]. Fortunately, protected cultivation is on the rise [2][3]. Beside the obvious protection against weather extremes, vegetable growing greenhouse horticulture provides a more resource-efficient alternative to agriculture, with investments in greenhouses not necessarily increasing the environmental impact, but even giving the possibility for it to be reduced [4].

The total area of greenhouses in 1980 was around 150.000 ha, and by 2019 it grew to almost 500.000 ha [3]. Fresh vegetable export in the world doubled between 2006 and 2016. In 2016 the top three fresh vegetable exporting countries were China, Mexico, and the Netherlands, followed by Spain and Canada. The most exported vegetables include tomatoes, sweet and chili peppers, cucumbers, and onions, and the former three can traditionally be grown in greenhouses. The area of greenhouses shows a growing tendency in almost all vegetable exporting countries around the world. The Netherlands was a bigger fresh vegetable exporter than Spain in 2016, although Spain had 14 times the greenhouse area of the Netherlands [2]. Although the research did not consider vegetables sold internally in the country of production, the efficiency of Dutch horticultural production over other countries clearly shows how technology and expertise can improve on crop yield.

Contrary to the extending industry of horticulture, in the Netherlands the number of greenhouse growers sank by 85% between 1980 and 2017, and the average managed hectares per one grower grew from 0.6 ha to 4 ha [5]. This tendency puts pressure on growers, while considerable amount of their time is spent on repetitive decision making, like irrigation scheduling [6]. A global shortage of skilled greenhouse labor was identified in 2013 [7], and this tendency did not improve later on in 2018 as well [8]. Because of these reasons, now more than ever, advancements in greenhouse automation are needed [9].

As [9] highlighted, one of the leading trends in recently built greenhouses is the inclusion of automation, and not only for climate systems. In the last couple of years, research on autonomous greenhouse control has seen growing popularity so much so, that in the Netherlands the Autonomous Greenhouse Challenge has been organised for the 3rd time in 4 years, challenging industry specialists and academic researchers alike in a practical greenhouse control setup. At the time of writing this paper, the 3rd challenge has been concluded, but only the results of the first two competitions have been published yet [10][11]. Practical viability of developed autonomous climate control algorithms has been shown, with 1 algorithm gaining more profit than commercial cucumber growers [10] and all the five algorithms overperforming commercial tomato growers [11] in real greenhouse growing experiments.

Beside climate control solutions, automated irrigation scheduling is also a well researched field of greenhouse technology. Researches show various benefits, that with advanced methods/strategies for irrigation can be achieved. For example, the use of automated and frequent irrigation scheduling could allow for reduced substrate volume, reducing costs and the environmental impact of disposing of the used substrate media [12]. Significant amount of water can be spared in warmer climates with improved irrigation techniques [13], which would reduce the environmental impact of horticulture. In an efficiency aspect, automated decision making for irrigation could significantly reduce the time growers need to manage irrigation [6]. Using models of plant water uptake and forecast weather conditions (especially solar radiation), irrigation can be applied automatically, before water-stress occurs and hinders photosynthesis for hours ([14], p.51).

Although multiple researches inspected the use of Model Predictive Control (MPC) in agricultural water management [15] [16] [17], only one example of MPC in horticultural irrigation [18] was found in the literature. One similar algorithm came up in the form of event-based generalized predictive control [19], which provided good results with reduced water usage. Because many Plant Water Status (PWS) and Evapotranspiration (ET) models are available in the literature, either simple or accurate, the use of MPC for irrigation decision making can promise good results.

The thesis introduces a new way of irrigation control, which uses predictions on Plant Water Uptake (PWU) and drain to estimate, how the water content of the growing substrate changes given climate predictions. Mixed Logical Dynamical (MLD) modelling is used to incorporate nonlinearities into a linear drain model in a structured way, and thus an accurate water balance prediction model is created. The proposed MLD MPC is a novel approach for irrigation scheduling, with modular structure and easily tuned behaviour for a versatile use in the irrigation of greenhouse grown high-wire crops. The thesis inspects the advantages and drawbacks of the developed MLD model and the MPC controller with different horizons. The results show, that the use of MPC for irrigation automation is viable, because it uses already available greenhouse hardware, it is easy to tune, and it provides irrigation decisions 24 hours ahead, which can greatly improve its acceptance with professional greenhouse growers, because it is easily supervised.

The thesis is structured as follows: Chapter 2 discusses the implementation of a simple PWU uptake model and presents unique drain modelling approaches. Chapter 3 shows the implementation of a linear and a MLD MPC controller. Chapter 4 evaluates the created models and controllers, and discusses the achieved results. Chapter 5 summarizes the findings of the thesis, and proposes further research directions.

1-1 Project scope

The project considers the following scope:

- Plants in their generative growth phase are considered, where biomass and Leaf Area Index (LAI) are nearly constant.
- The Electrical Conductivity (EC) and pH of the substrate are not taken into account.
- Substrate grown tomatoes and cucumbers are considered, which are both high-wire crops.
- Modern greenhouses are considered, with the following measurements available:
    - Solar radiation
    - Indoor humidity and temperature
    - Irrigation and drain
    - Slab/gutter scale weights

The reasons behind the items are discussed throughout the thesis.

Chapter 2

Modeling

To effectively control the water content of the substrate, the processes influencing its change are modelled. The term of water balance is introduced in detail in Section A-2-2. The water balance equation is the main process model which describes how the water content of the substrate changes over time.

$$y_{\mathrm{WC}}(k+1) \; = y_{\mathrm{WC}}(k) + Q_{\mathrm{IRR}}(k) - Q_{\mathrm{PWU}}(k) - Q_{\mathrm{DRN}}(k) \tag{2-1}$$

From (2-1) if the starting water content is known, the water content values in the future can be calculated using predicted values of irrigation, plant water uptake and drain. The irrigation amounts are the control inputs, so only the plant water uptake and drain needs to be modelled. In this chapter, the various modelling approaches are introduced, which were considered during the thesis project. During the data preparation, assumptions were introduced to make the modeling feasible or to simplify certain elements. The Assumptions A.1, A.2 and A.3, the need for them, and the reasoning behind their validity is discussed in Appendix A.

2-1 Evaluation of models

The evaluation of models can be straightforward, if only the error of the predictions/estimations is considered. [20] collected and organized watershed model evaluation methods. From the inspected measures, authors recommended the use of the Nash-Sutcliffe efficiency (NSE) [21], the Percent-bias (PBIAS) error index [22], and the Ratio of the Root Mean Square Error to the standard deviation of measured data (RSR) method. Using the notation, equations and descriptions of [20], a brief explanation is given on these three methods.

The NSE inspects the relation between the residual variance of simulated and measured data ("noise"), and the measured data variance ("information"). The calculation of the NSE is given by (2-2):

$$\text{NSE} = 1 - \left[\frac{\sum_{i=1}^{n} \left(Y_i^{obs} - Y_i^{sim} \right)^2}{\sum_{i=1}^{n} \left(Y_i^{obs} - Y^{mean} \right)^2} \right] \tag{2-2}$$

Where Y_i^{obs} is the i-th observation from a dataset with n instances. Y_i^{sim} is the i-th simulated value of the inspected variable(s), and Y^{mean} is the mean of the observed data. The value of NSE ranges from $-\infty$ to and inclusive 1, and values between 0 and 1 are generally acceptable levels of performance. Values under 0 mean, that the mean of the measurements provides better predictions than the simulated model. The calculation of the coefficient of determination (R^2) is almost identical to the calculation of NSE, the only difference is that R^2 is used to assess the performance on the fit of a statistical model, whereas the NSE evaluates how well a variable is predicted in simulation. In the following sections, NSE and R^2 is used interchangeably.

The PBIAS measures how often the simulated data is larger or smaller than the measured variables. Values other than 0 indicate model overestimation (PBIAS<0) or underestimation (PBIAS>0). (2-3) calculates PBIAS as a percentage value:

$$\text{PBIAS} = \left[\frac{\sum_{i=1}^{n} \left(Y_i^{obs} - Y_i^{sim} \right) \cdot (100)}{\sum_{i=1}^{n} \left(Y_i^{obs} \right)} \right] \tag{2-3}$$

with the same variables as defined below (2-2).

The RSR compares the Root Mean Square Error (RMSE) between simulated variables and measurements to the standard deviation of the measurements. The lower the RSR the more accurate the model simulation is. The calculation is given in (2-4).

$$\text{RSR} = \frac{\text{RMSE}}{\text{STDEV}_{obs}} = \frac{\left[\sqrt{\sum_{i=1}^{n} \left(Y_i^{obs} - Y_i^{sim} \right)^2} \right]}{\left[\sqrt{\sum_{i=1}^{n} \left(Y_i^{obs} - Y^{mean} \right)^2} \right]} \tag{2-4}$$

Other than the three recommended metrics, it is worthy to note Pearson's correlation coefficient r. r gives information on the linear correlation between simulated and measured data, with $r = 1$ being perfect linear correlation, $r = -1$ being perfect negative linear correlation, and $r = 0$ no linear correlation.

2-2 Plant water uptake modelling

The water uptake of the crop can be a function of many parameters. In relevant literature, various types of models are proposed to describe, how the amount of water taken up by plants is the function of climatic variables. [23] highlighted many different types of algorithms. The gold standard of how Evapotranspiration (ET) is calculated is described by the FAO Penman-Monteith equation [24]. Simplifications of the Penman-Monteith equation only depend on 1, 2 climatic parameters. [25] introduces a linear model where the amount of water taken up by the plant is only dependent on the solar radiation and a crop coefficient, as well as a constant nocturnal water uptake. A slightly more complex model is introduced in [26], where the plant water uptake is a bi-linear function of the solar radiation and the Vapor Pressure Deficit (VPD) of the crop. [27] extends the Penman-Monteith equation by taking into account the Leaf Area Index (LAI) of the crop, which results in a more accurate water uptake estimation for greenhouse grown crops. All of the mentioned models assume, that sufficient water is available at the root zone of the plant. This assumption is made for this thesis as well:

Assumption 2.1 (No drought-stress)**.** *The plants are not under drought stress during the generative growth phase.*

The reason why this assumption can be made comes from the growing practices of professional growers in horticulture: their strategies for irrigation involve measuring around 20-30% drain relative to irrigation, which keeps the water content of the substrate close to saturation during daytime, when the Plant Water Uptake (PWU) is the strongest. During nighttime the water content drops, but not enough to cause drought-stress.

Most of the ET and PWU models are fit for daily, or sometimes hourly predictions. Because the models are used for a Model Predictive Control (MPC) framework, and the derived PWU measurements (the process detailed in Section A-2-2) are created with a sampling time of 1 hour, the water uptake models are considered with the same sampling time. Because the water uptake of the crop was not measured directly, PWU was calculated using irrigation, drain and gutter scale measurement data. The models in this section are fitted to the calculated PWU values.

2-2-1 Linear model

First, the simplest model is considered: the linear regression model from [25]. (2-5) shows the original form of the model:

$$\lambda E = A_o K_c R + B_o \tag{2-5}$$

Where λ is the specific heat of water, R is the radiation, K_c is a coefficient specific to a crop, and A_0 and B_0 are regression coefficients. Because the model is only dependent on the radiation, and during night there is hardly any, B_0 is also called nocturnal ET. For the purpose of this research, $A_\mathrm{L} = \frac{A_0 \cdot K_c}{\lambda}$ is estimated together, and let $B_\mathrm{L} := \frac{B_0}{\lambda}$ to align the notation. So the modified linear model is:

$$E = A_\mathrm{L} R + B_\mathrm{L} \tag{2-6}$$

where E is the water taken up by the plant in l/m^2, R is the energy of the total radiation in J/m^2, A_L is the regression coefficient in l/J and B_L is the nocturnal ET in l/m^2. Because the models are fitted hourly, the above variables describe per hour values. The model fitting procedure is described in the following subsection together with the Bi-linear model.

2-2-2 Bi-Linear model

The main equation of the model from [26] can be seen in (2-7)

$$\lambda \cdot E = A \cdot R + B \cdot V \tag{2-7}$$

Where λ is the specific heat of water, E is the amount of water evaporating, A and B are linear coefficients, R is the solar radiation and V is the VPD. Because no data was available on either the VPD or the temperature of the leaves, VPD is not available to use for the identification. Because of this, it is substituted to Humidity Deficit (HD), which could serve as an alternative. The notation is changed slightly, with $A_{\mathrm{BL},1} = \frac{A}{\lambda}$, $A_{\mathrm{BL},2} = \frac{B}{\lambda}$, and by introducing an intercept coefficient B_{BL} for better accuracy. The Bi-linear model now reads:

$$E = A_{\mathrm{BL},1} \cdot R + A_{\mathrm{BL},2} \cdot H + B_{\mathrm{BL}} \tag{2-8}$$

where E is the water taken up by the plant in l/m^2, R is the energy of the total radiation in J/m^2, $A_{\mathrm{BL},1}$ is the first regression coefficient in l/J, H is the average humidity deficit in g/cm^3, $A_{\mathrm{BL},2}$ is the second regression coefficient in $\frac{cm^3 \cdot l}{g \cdot m^2}$ and B_{BL} is the intercept coefficient in l/m^2. Because the models are fitted hourly, the above variables describe per hour values.

To inspect, how well the two linear regression models describe the connection between PWU and the one or two climatic parameters, for each day in the dataset a model is fitted, which is then assessed for the NSE of the predictions, also known as the R^2. Figure 2-1 shows the daily achieved R^2 values (with the models fitted daily) as a function of the daily sum of radiation. The results clearly show, that on days with high radiation, the linear dependency between the variables is strong, and so a regression based model could describe the PWU well. On the other hand, on low radiation days the linear dependency is considerably lower. It can also be seen, that as expected the Bi-linear model has better fit if only considered on the training set.

Partial correlations	Today's radiation	Yesterday's radiation	Today's PWU
Today's radiation	1	0.13	0.84
Yesterday's radiation	0.13	1	0.13
Today's PWU	0.84	0.13	1

Table 2-1: Partial correlation between yesterday's radiation sum and today's PWU

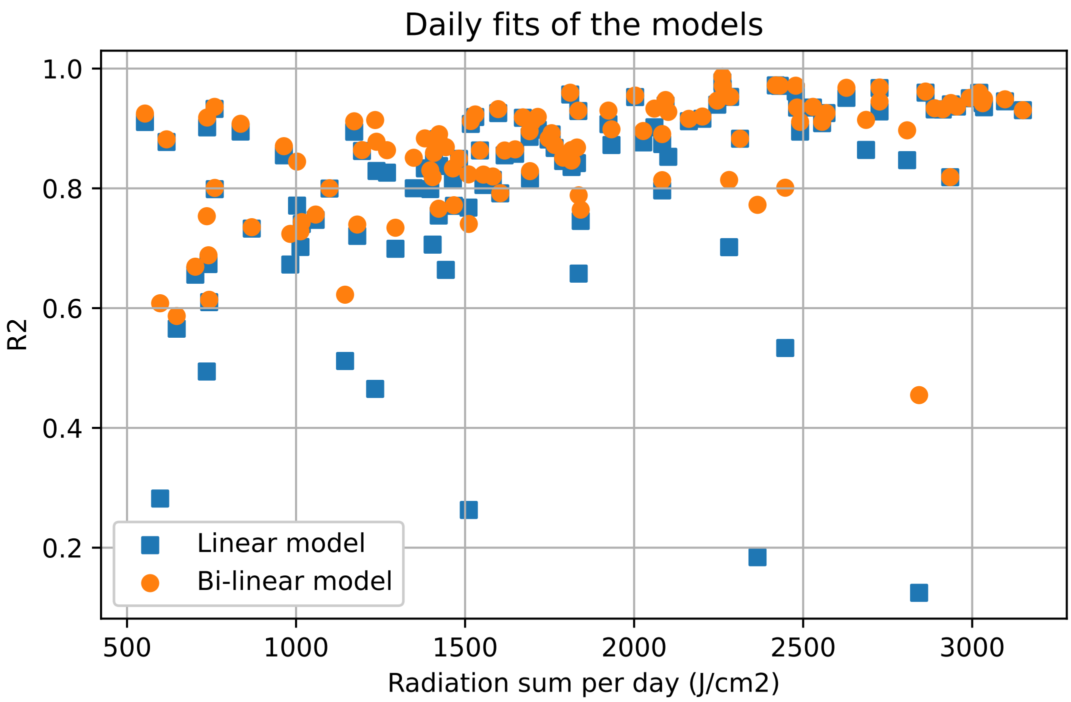

Figure 2-1: Water bucket model

A threshold is applied on the data, to filter out days with lower than 1000 J/cm^2 daily radiation. An additional filter is applied on the drain percentage: a drain percentage of more than 50% a day is considered as an anomaly, and so it is excluded from the model training. With these, around 16 % of the data was excluded. The remaining days are taken apart in a ratio of 70% training, 30% validation set. The days can be considered as independent, meaning, that the radiation of the previous day does not meaningfully influence the PWU of the actual day. This was verified by inspecting the partial correlation between the aforementioned variables. As Table 2-1 shows, the partial correlation is negligible between the mentioned variables.

With the days randomly separated into the training and validation sets, the coefficients of the linear and bi-linear models are estimated using regular least-squares approach. The fitted models are then tested on the validation set. Table 2-2 shows the results of the model fitting.

It can be seen, that the bi-linear model performs better in almost every area compared to the linear model. The improvements are most noticeable on the validation dataset, meaning, that the generalizability of the bi-linear model is higher. The high R^2 value of the Bi-linear

	Linear model	Bi-linear model
Linear coefficient(s)	$2.265 \cdot 10^{-7}$	$[1.482 \cdot 10^{-7}, 0.0255]$
Intercept coefficient	0.1052	0.0315
R2 on training set	0.83	0.84
R2 on validation set	0.79	0.86
PBIAS on training set	0%	0%
PBIAS on validation set	-2.138%	-0.484%
Normalized RMSE on training set	9%	9.1%
Normalized RMSE on validation set	10.7%	7.8%
RSR on training set	0.41	0.4
RSR on validation set	0.45	0.37

Table 2-2: Results of the PWU model fitting

model is above 0.8 on both the validation and training sets, and the normalized RMSE is under 10%, which is promising. Because of the simplicity of the model and the low amount of required inputs, this model is selected for the further work in this thesis. More complicated models require parameters (like convection, LAI, etc.) which were not available during the project, so their effectiveness was not tested.

2-3 Drain modelling

Another crucial process of the water balance equation is the drain. Because the controller is going to be used for prediction, drain has to be modelled in order to predict, how much drain is going to happen for certain combinations of irrigation and PWU.

Drain is the process of water leaving the growing substrate through channels in the bottom of the gutter. The accumulation of drain is measured and stored in the irrigation group of the climate computer, as it is seen in Table A-1. Physically, the drain is the excess water inside the substrate which was not absorbed, and so it is only dependent on the instantaneous water content of the substrate, $\nu(t)$, and the properties of the substrate itself. The process of water flowing through the substrate, which is not deeper than 0.5 meter, is most probably governed by a time-constant in the magnitude of minutes, if not seconds. There is a scarceness of drain models in the literature, which is probably due to the lack of predictive irrigation approaches which would use them, and the usual 5 minute sampling time in climate controller data, which does not allow the identification of a process with a faster time constant.

In the following subsections, various approaches are considered to model, how the drain measurements develop as a function of other measurements.

2-3-1 Water bucket model

Figure 2-2 shows all of the drain measurements from the data, with water deficit on the x axis. It can be seen, that above a Water Deficit (WD) of 0.4 l/m^2 practically no drain is measured. This knowledge can be used to simplify the model relations considerably with the following assumption:

Assumption 2.2 (No drain above a water deficit threshold)**.** *If WD is above a threshold, no drain is measured.*

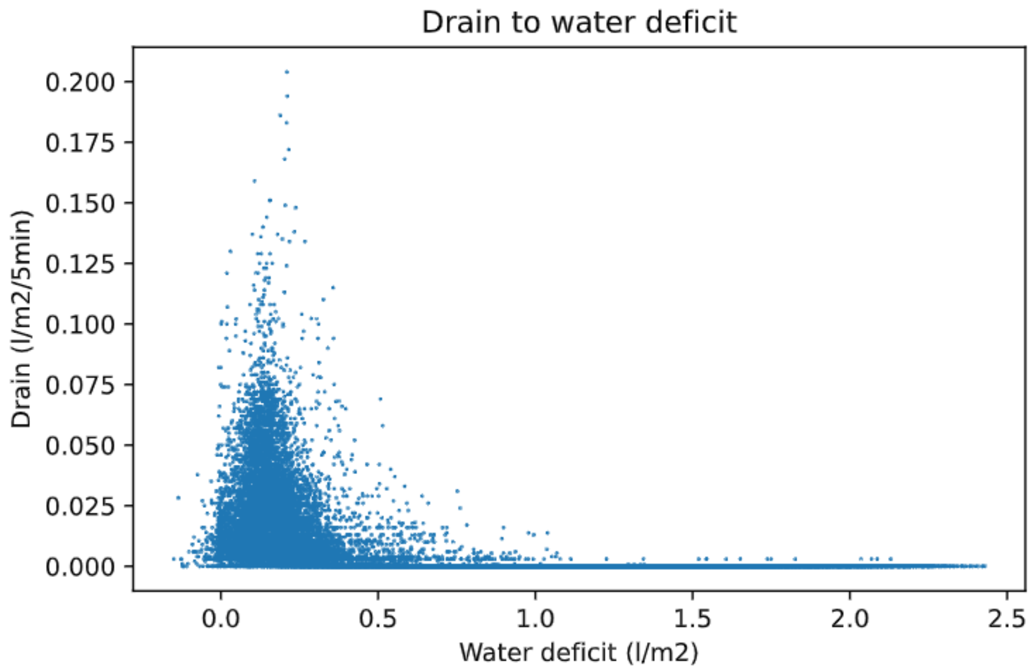

Figure 2-2: Drain - WD plot

The drain of the growing substrate can be modelled as a water bucket with a hole. As it can be seen in Figure 2-3, the hole of the bucket is in the upper part of the bucket. If the water level is below it, there is no drain happening. If the water level is in the level of the hole, the flow of the drain is a function of the water level. If the water level is above the upper part of the hole, we assume, that the drain is saturated.

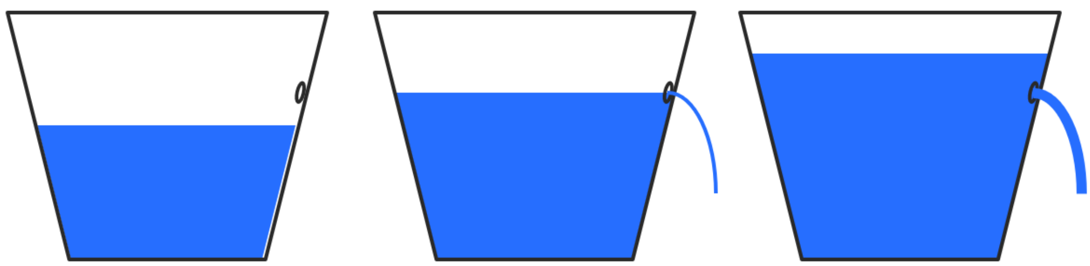

Figure 2-3: Water bucket model 1

Detailed in Section A-2-1, the WD is measured from the top of the bucket, which is denoted as maximum Volumetric Water Content (VWC). The meaning of WD is, that $\mu(t)$ liter of water per square-meter is missing until the VWC max level is reached. If the WD is above the flow start point, so $\mu(t) > y_{\mathrm{FSP}}$, no drain happens. If $\mu(t)$ is between the flow start point and the saturation point, so $y_{SP} < \mu(t) < y_{FSP}$, the drain flow is the function of current water deficit: $q(t) = f(\mu(t))$. And finally, if the water deficit is below the saturation point, $\mu(t) < y_{SP}$, we assume, that the maximum drain flow takes up a constant maximum value: $q(t) = q_{max}$. Figure 2-4 shows an illustration of the above explained concept.

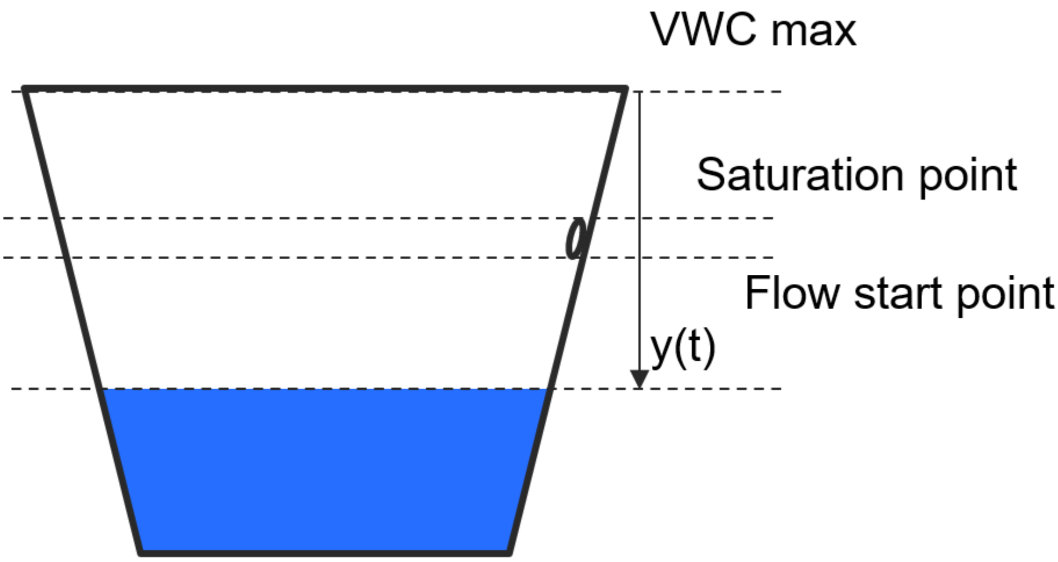

Figure 2-4: Water bucket model 2

The behaviour can be written up the following way:

$$q(t) = \begin{cases} 0 & \text{IF: } \mu(t) > y_{FSP} \\ f(\mu(t)) & \text{IF: } y_{SP} < \mu(t) < y_{FSP} \\ q_{max} & \text{IF: } \mu(t) < y_{SP} \end{cases} \tag{2-9}$$

If we assume a linear behaviour for the drain between the saturation and flow start points, so $f(\mu(t)) = a \cdot (\mu(t) - y_{FSP})$, the system turns into a piecewise affine (PWA) system:

$$q(t) = \begin{cases} 0 & \text{IF: } \mu(t) > y_{FSP} \\ a \cdot (\mu(t) - y_{FSP}) & \text{IF: } y_{SP} < \mu(t) < y_{FSP} \\ q_{max} & \text{IF: } \mu(t) < y_{SP} \end{cases} \tag{2-10}$$

The water deficit is influenced by not only the drain, but also the irrigation and the water taken up by the plant. For now, let us denote the drain flow by $q_{DRN}(t)$, the irrigation flow by $q_{IRR}(t)$ and the water flow which the plant takes up by $q_{PWU}(t)$. Because WD grows as the substrate gets dryer, in the water balance equation, the drain and plant water uptake flow have positive signs, while the irrigation has a negative sign.

$$\begin{aligned} \frac{d}{dt}\mu(t) &= q_{PWU}(t) + q_{DRN}(t) - q_{IRR}(t) \\ \frac{d}{dt}\mu(t) &= q_{PWU}(t) + q_{DRN}(\mu(t)) - q_{IRR}(t) \end{aligned} \tag{2-11}$$

The main time-constant of the system is most probably in the magnitude of minutes, if not seconds, so to correctly represent its dynamics, a sampling-time of 1-10 seconds would be needed. This is a problem, because the greenhouse data available is sampled every 5 minutes, $h_1 = 5\text{min}$. This means, that the measurements of the volumetric water content y_{VWC}, and therefore the water deficit measurements y_{WD} as well depict the following process:

$$\begin{aligned} y_{\text{WD}}(t_0 + h_1) &= y_{\text{WD}}(t_0) + \underbrace{\int_{t_0}^{t_0+h_1} q_{\text{PWU}}(t)dt}_{Q_{\text{PWU}}(t_0)} + \int_{t_0}^{t_0+h_1} q_{\text{DRN}}(\mu(t))dt - \\ &- \underbrace{\int_{t_0}^{t_0+h_1} q_{\text{IRR}}(t)dt}_{Q_{\text{IRR}}(t_0)} + \int_{t_0}^{t_0+h_1} \varepsilon(t)dt \end{aligned} \tag{2-12}$$

Where $\varepsilon(t)$ is an error affecting the measurements. Because we can not identify the dynamical model of drain with a data of this granularity, with the assumption that:

Assumption 2.3 (Water balance equation)**.** *The water balance equation holds*

we can try to identify an input-output behaviour:

$$\begin{aligned} Q_{\text{DRN}}(t_0) &= \int_{t_0}^{t_0+h_1} q_{\text{DRN}}(\mu(t))dt \\ Q_{\text{DRN}}(t_0) &= f\left(y_{\text{WD}}(t_0), \int_{t_0}^{t_0+h_1} q_{\text{PWU}}(t), \int_{t_0}^{t_0+h_1} q_{\text{IRR}}(t)\right) \\ Q_{\text{DRN}}(t_0) &= f\left(y_{\text{WD}}(t_0), Q_{\text{PWU}}(t_0), Q_{\text{IRR}}(t_0)\right) \end{aligned} \tag{2-13}$$

Because q_{DRN} is a function of only the instantaneous water deficit $\mu(t)$, and the water deficit can be estimated from the water deficit measurement at t_0 ($y_{\text{WD}}(t_0)$), the amount of irrigation which was applied between t_0 and $t_0 + h_1$ ($Q_{\text{IRR}}(t_0)$), and the water which was taken up by the plants in that interval ($Q_{\text{PWU}}(t_0)$). Following the previous equation, the following model can be written up:

$$\begin{aligned} &\text{Let:} \quad \tilde{y}_{\text{WD}}(t_0) = y_{\text{WD}}(t_0) - Q_{\text{IRR}}(t_0) + Q_{\text{PWU}}(t_0) \\ &\text{Then:} \\ &Q_{\text{DRN}}(t_0) = \begin{cases} 0 & \text{IF: } \tilde{y}_{\text{WD}}(t_0) > y_{\text{FSP}} \\ \alpha \cdot (\tilde{y}_{\text{WD}}(t_0) - y_{\text{FSP}}) & \text{IF: } y_{\text{SP}} < \tilde{y}_{\text{WD}}(t_0) < y_{\text{FSP}} \\ \beta & \text{IF: } \tilde{y}_{\text{WD}}(t_0) < y_{\text{SP}} \end{cases} \end{aligned} \tag{2-14}$$

This gives us three discrete time models for the water balance equation. Let us handle the water deficit as a state, and denote it with x. The system can then be written up by:

$$
\begin{aligned}
x(t+h_1) &= a_i x(t) + B_i \begin{bmatrix} Q_{\text{IRR}}(t) \\ Q_{\text{PWU}}(t) \\ 1 \end{bmatrix} \\
\begin{bmatrix} y(t) \\ Q_{\text{DRN}}(t) \end{bmatrix} &= C_i x(t) + D_i \begin{bmatrix} Q_{\text{IRR}}(t) \\ Q_{\text{PWU}}(t) \\ 1 \end{bmatrix} \quad \text{for} \quad i = 1, 2, 3
\end{aligned}
\tag{2-15}
$$

With:

$$
\begin{aligned}
&\text{IF:} \quad x(t) - Q_{\text{IRR}}(t) + Q_{\text{PWU}}(t) > y_{\text{FSP}} \quad \text{THEN:} \quad i = 1 \\
&x(t+h_1) = 1x(t) + \begin{bmatrix} -1 & 1 & 0 \end{bmatrix} \begin{bmatrix} Q_{\text{IRR}}(t) \\ Q_{\text{PWU}}(t) \\ 1 \end{bmatrix} \\
&\begin{bmatrix} y(t) \\ Q_{\text{DRN}}(t) \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} Q_{\text{IRR}}(t) \\ Q_{\text{PWU}}(t) \\ 1 \end{bmatrix} \\
&\text{IF:} \quad y_{\text{SP}} < x(t) - Q_{\text{IRR}}(t) + Q_{\text{PWU}}(t) < y_{\text{FSP}} \quad \text{THEN:} \quad i = 2 \\
&x(t+h_1) = (1+\alpha)x(t) + \begin{bmatrix} -(1+\alpha) & (1+\alpha) & -\alpha y_{\text{FSP}} \end{bmatrix} \begin{bmatrix} Q_{\text{IRR}}(t) \\ Q_{\text{PWU}}(t) \\ 1 \end{bmatrix} \\
&\begin{bmatrix} y(t) \\ Q_{\text{DRN}}(t) \end{bmatrix} = \begin{bmatrix} 1 \\ \alpha \end{bmatrix} x(t) + \begin{bmatrix} 0 & 0 & 0 \\ -\alpha & \alpha & -\alpha y_{\text{FSP}} \end{bmatrix} \begin{bmatrix} Q_{\text{IRR}}(t) \\ Q_{\text{PWU}}(t) \\ 1 \end{bmatrix} \\
&\text{IF:} \quad x(t) - Q_{\text{IRR}}(t) + Q_{\text{PWU}}(t) < y_{\text{SP}} \quad \text{THEN:} \quad i = 3 \\
&x(t+h_1) = 1x(t) + \begin{bmatrix} -1 & 1 & \beta \end{bmatrix} \begin{bmatrix} Q_{\text{IRR}}(t) \\ Q_{\text{PWU}}(t) \\ 1 \end{bmatrix} \\
&\begin{bmatrix} y(t) \\ Q_{\text{DRN}}(t) \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \beta \end{bmatrix} \begin{bmatrix} Q_{\text{IRR}}(t) \\ Q_{\text{PWU}}(t) \\ 1 \end{bmatrix}
\end{aligned}
\tag{2-16}
$$

Because the plant water uptake is calculated hourly, to get estimates for it every 5 minutes, we assume that:

Assumption 2.4 (Constant intersample PWU)**.** *In an hourly interval, the plant water uptake is constant.*

This means, that:

$$
Q_{\text{PWU}}(t + i \cdot h_1) = \frac{\bar{Q}_{\text{PWU}}(t)}{12} \quad \text{for} \quad i = 0, 1, ..., 11 \tag{2-17}
$$

where $\bar{Q}_{\text{PWU}}(t)$ is the estimated or measured plant water uptake for the upcoming hour.

The identifiable parameters in this Piecewise Affine (PWA) system are:

$$\theta = \begin{bmatrix} \alpha \\ \beta \\ y_{\text{FSP}} \\ y_{\text{SP}} \end{bmatrix} \tag{2-18}$$

To identify all of the parameters, first y_{FSP} has to be found. The value of this parameter can be found manually by searching for the first drain moment.

The problem with the above described model is, that for $i = 2$, the linear model is unstable, and this makes identifying the model in open-loop impossible. To remedy this, water content can be used instead of water deficit. The main problem with water content is, that only the maximal water content is known. There is no gutter scale measurement for a water content of 0%. Using now the gutter scale measurement $y_{\text{GS}}(t)$ variable with unit of l/m2:

$$y_{\text{GS}}(t_0 + h_1) \; = y_{\text{GS}}(t_0) - Q_{\text{PWU}}(t_0) - Q_{\text{DRN}}(t_0) + Q_{\text{IRR}}(t_0) \tag{2-19}$$

$$\begin{aligned} &\text{Let:} \quad \tilde{y}_{\text{GS}}(t_0) = y_{\text{GS}}(t_0) + Q_{\text{IRR}}(t_0) - Q_{\text{PWU}}(t_0) \\ &\text{Then:} \\ &Q_{\text{DRN}}(t_0) = \begin{cases} 0 & \text{IF: } \tilde{y}_{\text{GS}}(t_0) < y_{\text{FSP}} \\ \alpha \cdot (\tilde{y}_{\text{GS}}(t_0) - y_{\text{FSP}}) & \text{IF: } y_{\text{SP}} > \tilde{y}_{\text{GS}}(t_0) > y_{\text{FSP}} \\ \beta & \text{IF: } \tilde{y}_{\text{GS}}(t_0) > y_{\text{SP}} \end{cases} \end{aligned} \tag{2-20}$$

Now the PWA system with the gutter scale measurement $y(t)$ as state and an output, and y_{FSP} and y_{SP} readjusted to the gutter scale measurements:

$$
\begin{aligned}
&\text{IF:}\quad y(t)+Q_{\text{IRR}}(t)-Q_{\text{PWU}}(t)<y_{\text{FSP}}\quad\text{THEN:}\quad i=1\\
&y(t+h_1)=1y(t)+\begin{bmatrix}1 & -1 & 0\end{bmatrix}\begin{bmatrix}Q_{\text{IRR}}(t)\\ Q_{\text{PWU}}(t)\\ 1\end{bmatrix}\\
&\begin{bmatrix}y(t)\\ Q_{\text{DRN}}(t)\end{bmatrix}=\begin{bmatrix}1\\ 0\end{bmatrix}y(t)+\begin{bmatrix}0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}Q_{\text{IRR}}(t)\\ Q_{\text{PWU}}(t)\\ 1\end{bmatrix}\\
&\text{IF:}\quad y_{\text{SP}}>y(t)+Q_{\text{IRR}}(t)-Q_{\text{PWU}}(t)>y_{\text{FSP}}\quad\text{THEN:}\quad i=2\\
&y(t+h_1)=(1-\alpha)y(t)+\begin{bmatrix}(1-\alpha) & (\alpha-1) & +\alpha y_{\text{FSP}}\end{bmatrix}\begin{bmatrix}Q_{\text{IRR}}(t)\\ Q_{\text{PWU}}(t)\\ 1\end{bmatrix}\\
&\begin{bmatrix}y(t)\\ Q_{\text{DRN}}(t)\end{bmatrix}=\begin{bmatrix}1\\ \alpha\end{bmatrix}y(t)+\begin{bmatrix}0 & 0 & 0\\ \alpha & -\alpha & -\alpha y_{\text{FSP}}\end{bmatrix}\begin{bmatrix}Q_{\text{IRR}}(t)\\ Q_{\text{PWU}}(t)\\ 1\end{bmatrix}\\
&\text{IF:}\quad y(t)+Q_{\text{IRR}}(t)-Q_{\text{PWU}}(t)>y_{\text{SP}}\quad\text{THEN:}\quad i=3\\
&y(t+h_1)=1y(t)+\begin{bmatrix}1 & -1 & -\beta\end{bmatrix}\begin{bmatrix}Q_{\text{IRR}}(t)\\ Q_{\text{PWU}}(t)\\ 1\end{bmatrix}\\
&\begin{bmatrix}y(t)\\ Q_{\text{DRN}}(t)\end{bmatrix}=\begin{bmatrix}1\\ 0\end{bmatrix}y(t)+\begin{bmatrix}0 & 0 & 0\\ 0 & 0 & \beta\end{bmatrix}\begin{bmatrix}Q_{\text{IRR}}(t)\\ Q_{\text{PWU}}(t)\\ 1\end{bmatrix}
\end{aligned}
\tag{2-21}
$$

When inspecting the data, it was noticed, that there is a delay in the system. This can be corrected for by introducing a buffer state for the drain, x_1:

$$
\begin{aligned}
\begin{bmatrix}y(t+h_1)\\ x_1(t+h_1)\end{bmatrix}&=A_i\begin{bmatrix}y(t)\\ x_1(t)\end{bmatrix}+B_i\begin{bmatrix}Q_{\text{IRR}}(t)\\ Q_{\text{PWU}}(t)\\ 1\end{bmatrix}\\
\begin{bmatrix}y(t)\\ Q_{\text{DRN}}(t)\end{bmatrix}&=C_iy(t)+D_i\begin{bmatrix}Q_{\text{IRR}}(t)\\ Q_{\text{PWU}}(t)\\ 1\end{bmatrix}\quad\text{for}\quad i=1,2,3
\end{aligned}
\tag{2-22}
$$

And so with it the parametrized model:

$$
\begin{aligned}
&\text{IF:}\quad x(t) + Q_{\text{IRR}}(t) - Q_{\text{PWU}}(t) < y_{\text{FSP}} \quad \text{THEN:}\quad i = 1 \\
&\begin{bmatrix} y(t+h_1) \\ x_1(t+h_1) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y(t) \\ x_1(t) \end{bmatrix} + \begin{bmatrix} 1 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} Q_{\text{IRR}}(t) \\ Q_{\text{PWU}}(t) \\ 1 \end{bmatrix} \\
&\begin{bmatrix} y(t) \\ Q_{\text{DRN}}(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y(t) \\ x_1(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} Q_{\text{IRR}}(t) \\ Q_{\text{PWU}}(t) \\ 1 \end{bmatrix} \\
&\text{IF:}\quad y_{\text{SP}} > x(t) + Q_{\text{IRR}}(t) - Q_{\text{PWU}}(t) > y_{\text{FSP}} \quad \text{THEN:}\quad i = 2 \\
&\begin{bmatrix} y(t+h_1) \\ x_1(t+h_1) \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ \alpha & 0 \end{bmatrix} \begin{bmatrix} y(t) \\ x_1(t) \end{bmatrix} + \begin{bmatrix} 1 & -1 & 0 \\ \alpha & -\alpha & -\alpha y_{\text{FSP}} \end{bmatrix} \begin{bmatrix} Q_{\text{IRR}}(t) \\ Q_{\text{PWU}}(t) \\ 1 \end{bmatrix} \\
&\begin{bmatrix} y(t) \\ Q_{\text{DRN}}(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y(t) \\ x_1(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} Q_{\text{IRR}}(t) \\ Q_{\text{PWU}}(t) \\ 1 \end{bmatrix} \\
&\text{IF:}\quad x(t) + Q_{\text{IRR}}(t) - Q_{\text{PWU}}(t) > y_{\text{SP}} \quad \text{THEN:}\quad i = 3 \\
&\begin{bmatrix} y(t+h_1) \\ x_1(t+h_1) \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y(t) \\ x_1(t) \end{bmatrix} + \begin{bmatrix} 1 & -1 & 0 \\ 0 & 0 & \beta \end{bmatrix} \begin{bmatrix} Q_{\text{IRR}}(t) \\ Q_{\text{PWU}}(t) \\ 1 \end{bmatrix} \\
&\begin{bmatrix} y(t) \\ Q_{\text{DRN}}(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y(t) \\ x_1(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} Q_{\text{IRR}}(t) \\ Q_{\text{PWU}}(t) \\ 1 \end{bmatrix}
\end{aligned}
\tag{2-23}
$$

The developed model failed to yield good results. The predicted drain did not follow the actual drain well, and the predicted water content had scaling and drifting issues. Because of this a new approach was taken.

2-3-2 Data-driven ARX drain model

Because the "water bucket model" did not bring favorable results, a more general model identification was used next. Using the inputs and outputs already defined in the previous section, an Autoregressive with Exogenous Input (ARX) model was fitted on the available data. A Multiple Input Multiple Output (MIMO) ARX system has the following structure:

$$
\begin{aligned}
A_{11}(q)y_1(k) &= A_{12}y_2(k) + B_{11}(q)u_1(k) + B_{12}(q)u_2(k) \\
A_{22}(q)y_2(k) &= A_{21}y_1(k) + B_{21}(q)u_1(k) + B_{22}(q)u_2(k)
\end{aligned}
\tag{2-24}
$$

Where $A_{ij}(q)$ and $B_{ij}(q)$ are polynomials of the delay operator q. To ease the notation, the time index is denoted by $k \in \mathbb{Z}$, with $t = t_0 + k \cdot h_1$. The polynomials are given by:

$$
\begin{aligned}
A_{ii}(q) &= 1 + a_1^{ii} q^{-1} + \ldots + a_{n_{A_{ii}}}^{ii} q^{-n_{a_{ii}}} \\
A_{ij}(q) &= a_1^{ij} q^{-1} + \ldots + a_{n_{A_{ij}}}^{ij} q^{-n_{a_{ij}}} \quad \text{for} \quad i \neq j \\
B_{ij}(q) &= b_1^{ij} + b_2^{ij} q^{-1} + \ldots + b_{n_{B_{ij}}}^{ij} q^{-n_{b_{ij}}} \quad \text{for} \quad \forall i, j
\end{aligned}
\tag{2-25}
$$

Where the superscripts ^{ij} denote, which polynomials the coefficients belong to. To create a causal system, the following inequalities have to hold:

$$\text{for a fixed } i\text{:} \quad n_{A_{ii}} \geq n_{A_{ij}} \quad \text{AND} \quad n_{A_{ii}} \geq n_{B_{ij}} \quad \text{for} \quad \forall j \tag{2-26}$$

We reformulate (2-27) to include the input and output variables defined in the previous section:

$$\begin{aligned} A_{11}(q)y_{\mathrm{WD}}(k) &= A_{12}(q)Q_{\mathrm{DRN}}(k) + B_{11}(q)Q_{\mathrm{IRR}}(k) + B_{12}(q)Q_{\mathrm{PWU}}(k) \\ A_{22}(q)Q_{\mathrm{DRN}}(k) &= A_{21}(q)y_{\mathrm{WD}}(k) + B_{21}(q)Q_{\mathrm{IRR}}(k) + B_{22}(q)Q_{\mathrm{PWU}}(k) \end{aligned} \tag{2-27}$$

The parameters to be defined for the identification are: $n_{A_{ij}}$ and $n_{B_{ij}}$. Beside these, the identification data was cut to only include measurements below a certain WD level (similarly to the flow start point in the water bucket model, Assumption 2.2). This is needed, because above a WD of 0.4 l/m^2, no drain happens, and this nonlinearity can cause problems with the drain prediction. The number of parameters can be written up the following way:

$$N_a = \begin{bmatrix} n_{A_{11}} & n_{A_{12}} \\ n_{A_{21}} & n_{A_{22}} \end{bmatrix} \quad \text{AND} \quad N_b = \begin{bmatrix} n_{B_{11}} & n_{B_{12}} \\ n_{B_{21}} & n_{B_{22}} \end{bmatrix} \tag{2-28}$$

The system was identified with the use of the MATLAB function 'arx()', which can identify linear ARX models, given the coefficient matrices N_a and N_b. Because the dynamics of the system can change over time (plant growth, seasonal effects), the identification was made progressively, using a sliding window. This can be seen in Figure 2-5. Using the available data, the first 10 days are used first as identification data, and then the model is validated on the 11th day. After the validation happens, the window slides, starting on the 2nd day, and ending on the 11th day (including in the training the previous evaluation day). The reidentified model is the validated on the upcoming day, and so forth. This approach is used to imitate how the algorithm could be used in real world scenario.

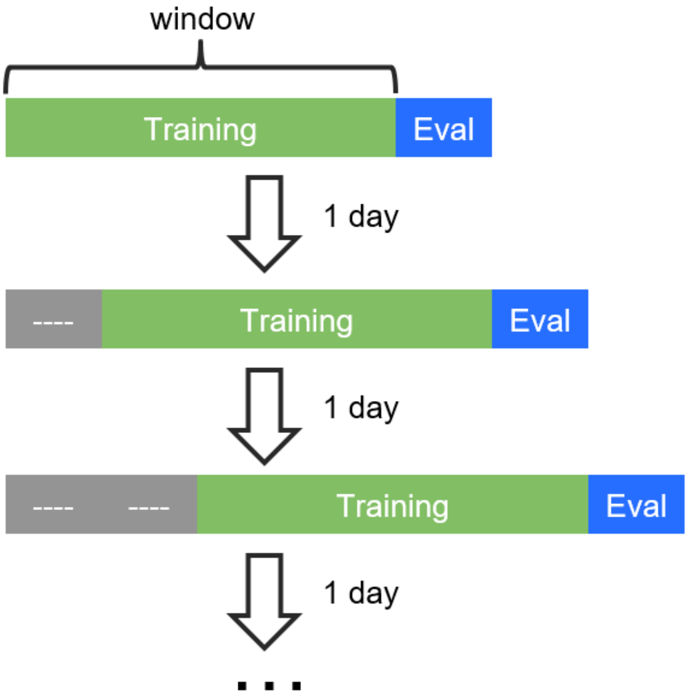

Figure 2-5: Progressive identification with sliding window

After a number of experiments, the following values brought the best fits, R^2, and total drain accuracy[1]:

$$N_a = \begin{bmatrix} 3 & 3 \\ 2 & 3 \end{bmatrix} \quad \text{AND} \quad N_b = \begin{bmatrix} 3 & 1 \\ 3 & 1 \end{bmatrix} \tag{2-29}$$

The data used was the part of the days, when the water deficit was below 0.4 l/m^2. Because of the fast dynamics of how the drain happens, only a steady-state response should be measurable between irrigation and drain. This, however, is not the case, because with only a first order polynomial for Q_{IRR}, the training and validation results were both inferior compared to a 3rd order polynomial. Because the Q_{PWU} input is a constant inside an hour, and only changes hourly, a first order polynomial was enough to describe its effect. Because of the delays between the drain sensor and the gutter scale (also mentioned in Section 2-3-1), the effect of drain on the water deficit was given a 3rd order polynomial. Anything below brought inferior results. This effect was not present in the influence of water deficit on drain. The number of autoregression coefficients $n_{a_{11}}$ and $n_{a_{22}}$ was set to 3 to keep the system causal.

With the above setup and progressive identification, the total drain accuracy was maintained above 70% for almost all days (from a set of 100 days), which is the most important factor in predicting the water deficit over the course of days in an hourly resolution.

[1]total drain accuracy is the ratio between the sum of the predicted drain increments and the sum of the measured drain increments

The identified ARX model can be transformed into a state-space representation, with the number of states being equal to $n_{A_{11}} + n_{A_{22}}$. The transformation was done using MATLAB's built in 'ss()' command. The function returns the SS system in an observable canonical form. Using the command, the obtained system is in the following form:

$$\begin{aligned} \boldsymbol{x}_{\mathrm{DRN}}(k+1) &= A_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}(k) + B\begin{bmatrix} Q_{\mathrm{IRR}}(k) \\ Q_{\mathrm{PWU}}(k) \end{bmatrix} \\ \begin{bmatrix} y_{\mathrm{WD}}(k) \\ Q_{\mathrm{DRN}}(k) \end{bmatrix} &= C\boldsymbol{x}_{\mathrm{DRN}}(k) + D\begin{bmatrix} Q_{\mathrm{IRR}}(k) \\ Q_{\mathrm{PWU}}(k) \end{bmatrix} \end{aligned} \tag{2-30}$$

Where $\boldsymbol{x} \in \mathbb{R}^n$, with $n = n_{A_{11}} + n_{A_{22}}$, and $A_{\mathrm{DRN}} \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times 2}$, $C \in \mathbb{R}^{2\times n}$ and $D \in \mathbb{R}^{2\times 2}$. D being non-zero shows, that some parts of the response of the system is stationary.

2-4 Water balance modeling

From the results of the last section, we can move on to formulate the prediction model for the water balance. Although the ARX model has an output for the water deficit, its accuracy is low, and the whole model is only usable below a certain water deficit level. From this follows, that it would be better to only use the drain as output in the ARX model, but this lead to compromised drain accuracy. The imposed structure with the cross dependency of drain and water deficit lead to an increased drain prediction accuracy, and so keeping the model was favorable. The water balance equation written on the water deficit is:

$$y_{\mathrm{WD}}(t+h_1) \; = y_{\mathrm{WD}}(t) + Q_{\mathrm{PWU}}(t) + Q_{\mathrm{DRN}}(t) - Q_{\mathrm{IRR}}(t) \tag{2-31}$$

From the above equation, $Q_{\mathrm{IRR}}(t)$ is the control input, $Q_{\mathrm{PWU}}(t)$ is acting as a measureable disturbance, and $Q_{\mathrm{DRN}}(t)$ is calculated using the model identified in the previous section. In line with this, we rename $Q_{\mathrm{IRR}}(t)$ as $u(t)$, $Q_{\mathrm{PWU}}(t)$ as $w(t)$ and Q_{DRN} as y_{DRN}. Now the water balance equation on the water deficit y_{WD} reads:

$$\begin{aligned} y_{\mathrm{WD}}(t+h_1) \; &= y_{\mathrm{WD}}(t) + w(t) + f\left(\boldsymbol{x}_{\mathrm{DRN}}, u(t), w(t)\right) - u(t) \\ y_{\mathrm{WD}}(t+h_1) \; &= y_{\mathrm{WD}}(t) + w(t) + \begin{bmatrix} 0 & 1 \end{bmatrix}\left(C\boldsymbol{x}_{\mathrm{DRN}}(k) + D\begin{bmatrix} u(t) \\ w(t) \end{bmatrix}\right) - u(t) \end{aligned} \tag{2-32}$$

Now, with the matrices B, C and D from (2-30):

$$C_{\mathrm{DRN}} = \begin{bmatrix} 0 & 1 \end{bmatrix} C \quad \text{and} \quad D = \begin{bmatrix} * & * \\ d_u & d_w \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} B_u & B_w \end{bmatrix} \tag{2-33}$$

We can write up the water balance equation as:

$$\begin{aligned} y_{\mathrm{WD}}(t+h_1) \; &= y_{\mathrm{WD}}(t) + w(t) + C_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}(t) + d_u u(t) + d_w w(t) - u(t) \\ y_{\mathrm{WD}}(t+h_1) \; &= y_{\mathrm{WD}}(t) + (1+d_w)w(t) + C_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}(t) + (d_u - 1)u(t) \end{aligned} \tag{2-34}$$

And so the new state-space system emerges:

$$\underbrace{\begin{bmatrix} y_{\mathrm{WD}}(t+h_1) \\ \boldsymbol{x}_{\mathrm{DRN}}(t+h_1) \end{bmatrix}}_{\boldsymbol{x}(t+h_1)} = \underbrace{\begin{bmatrix} 1 & C_{\mathrm{DRN}} \\ 0 & A_{\mathrm{DRN}} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} y_{\mathrm{WD}}(t) \\ \boldsymbol{x}_{\mathrm{DRN}}(t) \end{bmatrix}}_{\boldsymbol{x}(t)} + \underbrace{\begin{bmatrix} d_u - 1 \\ B_u \end{bmatrix}}_{B} u(t) + \underbrace{\begin{bmatrix} 1 + d_w \\ B_w \end{bmatrix}}_{G} w(t)$$
$$y_{\mathrm{WD}}(t) = \underbrace{\begin{bmatrix} 1 & 0 \end{bmatrix}}_{C} \begin{bmatrix} y_{\mathrm{WD}}(t) \\ \boldsymbol{x}_{\mathrm{DRN}}(t) \end{bmatrix} \tag{2-35}$$

And so

$$\begin{aligned} \boldsymbol{x}(t+h_1) &= A\boldsymbol{x}(t) + Bu(t) + Gw(t) \\ y_{\mathrm{WD}}(t) &= C\boldsymbol{x}(t) \end{aligned} \tag{2-36}$$

2-4-1 Hybrid water balance modeling

Because the water-balance model of the previous section is only valid under a specific water deficit threshold $\bar{y}_{\mathrm{WD}}$, for a hybrid water balance model, the system has to be transformed into a Mixed Logical Dynamical (MLD) form. Only looking at the identified drain model:

$$\begin{aligned} \boldsymbol{x}_{\mathrm{DRN}}(k+1) &= A_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}(k) + B_u u(k) + B_w w(k) \\ y_{\mathrm{DRN}}(k) &= C_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}(k) + d_u u(k) + d_w w(k) \end{aligned} \tag{2-37}$$

Where k denotes a sample every 5 minutes. The output of this system, $y_{\mathrm{DRN}}(k)$ with unit $\frac{l}{m^2 \cdot 5\mathrm{min}}$, depends on the water deficit, $y_{\mathrm{WD}}(k)$. If $y_{\mathrm{WD}}(k) < \bar{y}_{\mathrm{WD}}$, then the output is given by the above output equation, and if $y_{\mathrm{WD}}(k) \geq \bar{y}_{\mathrm{WD}}$, then the output is 0. This can also be represented (as shown in [28]) by an auxiliary binary variable, δ. If $y_{\mathrm{WD}}(k) < \bar{y}_{\mathrm{WD}}$, then $\delta = 1$, and if $y_{\mathrm{WD}}(k) \geq \bar{y}_{\mathrm{WD}}$, then $\delta = 0$.

$$\begin{aligned} \boldsymbol{x}_{\mathrm{DRN}}(k+1) &= A_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}(k) + B_u u(k) + B_w w(k) \\ y_{\mathrm{DRN}}(k) &= C_{\mathrm{DRN}}\delta(k)\boldsymbol{x}_{\mathrm{DRN}}(k) + d_u\delta(k)u(k) + d_w\delta(k)w(k) \end{aligned} \tag{2-38}$$

The condition in the previous paragraph can be written as:

$$\begin{aligned} \delta(k)\left(y^{\min} - \bar{y}_{\mathrm{WD}}\right) &\leq y_{\mathrm{WD}}(k) - \bar{y}_{\mathrm{WD}} \\ \delta(k)\left(y^{\max} - \bar{y}_{\mathrm{WD}} + \varepsilon\right) &\leq -y_{\mathrm{WD}}(k) + y^{\max} \end{aligned} \tag{2-39}$$

with $y^{\min}$ being the smallest, and $y^{\max}$ being the largest possible value $y_{\mathrm{WD}}(k)$ can take, and ε being the computational precision (a number really close to 0). Because now the state equation is nonlinear, new real valued auxiliary variables have to be introduced: $z_1(k) = \delta(k)C_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}(k)$, $z_2(k) = \delta(k)u(k)$ and $z_3(k) = \delta(k)w(k)$, with $z_1, z_2, z_3 \in \mathbb{R}$. Finally, let $\boldsymbol{z} = [z_1, z_2, z_3]^T$. Constraints are introduced to keep the values of $\boldsymbol{z}$ connected to the variables:

$$
\begin{aligned}
z_1(k) &\leq C_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}^{\max}\delta(k)\\
z_1(k) &\geq C_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}^{\min}\delta(k)\\
z_1(k) &\leq C_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}(k) - C_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}^{\min}(1-\delta(k))\\
z_1(k) &\geq C_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}(k) - C_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}^{\max}(1-\delta(k))\\
z_2(k) &\leq u^{\max}\delta(k)\\
z_2(k) &\geq u^{\min}\delta(k)\\
z_2(k) &\leq u(k) - u^{\min}(1-\delta(k))\\
z_2(k) &\geq u(k) - u^{\max}(1-\delta(k))\\
z_3(k) &\leq w^{\max}\delta(k)\\
z_3(k) &\geq w^{\min}\delta(k)\\
z_3(k) &\leq w(k) - w^{\min}(1-\delta(k))\\
z_3(k) &\geq w(k) - w^{\max}(1-\delta(k))
\end{aligned}
\tag{2-40}
$$

With $u^{\min}$ and $w^{\min}$ being the smallest, $u^{\max}$ and $w^{\max}$ being the largest irrigation and PWU input respectively. Now the hybrid system can be written as:

$$
\begin{aligned}
\boldsymbol{x}_{\mathrm{DRN}}(k+1) &= A_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}(k) + B_u u(k) + B_w w(k)\\
y_{\mathrm{DRN}}(k) &= \underbrace{\begin{bmatrix}1 & d_u & d_w\end{bmatrix}}_{C_z}\boldsymbol{z}(k)
\end{aligned}
\tag{2-41}
$$

We can combine the hybrid drain model with the water balance model, with also observing the drain occuring (this is important by the MPC formulation):

$$
\begin{aligned}
\underbrace{\begin{bmatrix}y_{\mathrm{WD}}(k+1)\\ \boldsymbol{x}_{\mathrm{DRN}}(k+1)\end{bmatrix}}_{\boldsymbol{x}(k+1)} &= \underbrace{\begin{bmatrix}1 & 0\\ 0 & A_{\mathrm{DRN}}\end{bmatrix}}_{A}\underbrace{\begin{bmatrix}y_{\mathrm{WD}}(k)\\ \boldsymbol{x}_{\mathrm{DRN}}(k)\end{bmatrix}}_{\boldsymbol{x}(k)} + \underbrace{\begin{bmatrix}-1\\ B_u\end{bmatrix}}_{B_1}u(k) + \underbrace{\begin{bmatrix}C_z\\ 0\end{bmatrix}}_{B_3}\boldsymbol{z}(k) + \underbrace{\begin{bmatrix}1\\ B_w\end{bmatrix}}_{G}w(k)\\
\begin{bmatrix}y_{\mathrm{WD}}(k)\\ y_{\mathrm{DRN}}(k)\end{bmatrix} &= \underbrace{\begin{bmatrix}1 & 0\\ 0 & 0\end{bmatrix}}_{C}\begin{bmatrix}y_{\mathrm{WD}}(k)\\ \boldsymbol{x}_{\mathrm{DRN}}(k)\end{bmatrix} + \underbrace{\begin{bmatrix}0\\ C_z\end{bmatrix}}_{D_3}\boldsymbol{z}(k)
\end{aligned}
\tag{2-42}
$$

And so the system with concise notation is:

$$
\begin{aligned}
\boldsymbol{x}(k+1) &= A\boldsymbol{x}(k) + B_1 u(k) + B_3\boldsymbol{z}(k) + Gw(k)\\
\begin{bmatrix}y_{\mathrm{WD}}(k)\\ y_{\mathrm{DRN}}(k)\end{bmatrix} &= C\boldsymbol{x}(k) + D_3\boldsymbol{z}(k)
\end{aligned}
\tag{2-43}
$$

Reorganizing the constraints into the form:

$$
\begin{aligned}
&\underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & -C_{\text{DRN}} \\ 0 & C_{\text{DRN}} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -1 & 0 \\ 1 & 0 \end{bmatrix}}_{E_1} \underbrace{\begin{bmatrix} y_{\text{WD}}(k) \\ \boldsymbol{x}_{\text{DRN}}(k) \end{bmatrix}}_{\boldsymbol{x}(k)} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{E_2} u(k) + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \\ 1 \\ 0 \\ 0 \end{bmatrix}}_{E_3} w(k) + \\
&+ \underbrace{\begin{bmatrix} -C_{\text{DRN}}\boldsymbol{x}_{\text{DRN}}^{\max} \\ C_{\text{DRN}}\boldsymbol{x}_{\text{DRN}}^{\min} \\ -C_{\text{DRN}}\boldsymbol{x}_{\text{DRN}}^{\min} \\ C_{\text{DRN}}\boldsymbol{x}_{\text{DRN}}^{\max} \\ -u^{\max} \\ u^{\min} \\ -u^{\min} \\ u^{\max} \\ -w^{\max} \\ w^{\min} \\ -w^{\min} \\ w^{\max} \\ y^{\min} - \bar{y}_{\text{WD}} \\ y^{\max} - \bar{y}_{\text{WD}} + \varepsilon \end{bmatrix}}_{\boldsymbol{E}_4} \delta(k) + \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ -1 & 0 & 0 \\ 1 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{E_5} \begin{bmatrix} z_1(k) \\ z_2(k) \\ z_3(k) \end{bmatrix} \leq \underbrace{\begin{bmatrix} 0 \\ 0 \\ -C_{\text{DRN}}\boldsymbol{x}_{\text{DRN}}^{\min} \\ C_{\text{DRN}}\boldsymbol{x}_{\text{DRN}}^{\max} \\ 0 \\ 0 \\ -u^{\min} \\ u^{\max} \\ 0 \\ 0 \\ -w^{\min} \\ w^{\max} \\ -\bar{y}_{\text{WD}} \\ y^{\max} \end{bmatrix}}_{\boldsymbol{g}_6} \\
&E_1\boldsymbol{x}(k) + E_2u(k) + E_3w(k) + E_4\delta(k) + E_5\boldsymbol{z}(k) \leq \boldsymbol{g}_6
\end{aligned} \tag{2-44}
$$

And so the complete hybrid system is:

$$
\begin{aligned}
\boldsymbol{x}(k+1) &= A\boldsymbol{x}(k) + B_1u(k) + B_3\boldsymbol{z}(k) + Gw(k) \\
\begin{bmatrix} y_{\text{WD}}(k) \\ \boldsymbol{x}_{\text{DRN}}(k) \end{bmatrix} &= C\boldsymbol{x}(k) + D_3\boldsymbol{z}(k) \\
\text{s. t.} \quad & E_1\boldsymbol{x}(k) + E_2u(k) + E_3w(k) + E_4\delta(k) + E_5\boldsymbol{z}(k) \leq \boldsymbol{g}_6
\end{aligned} \tag{2-45}
$$

Where the system has 1 binary variable, and 14 linear constraints.

2-4-2 PWA model

The predictions with the created hybrid model had issues. As can be seen in Figure 2-6, the prediction closely follows the actual measurements until a water deficit of 0 is reached. Here, the model goes into a negative regime, which would not be possible physically.

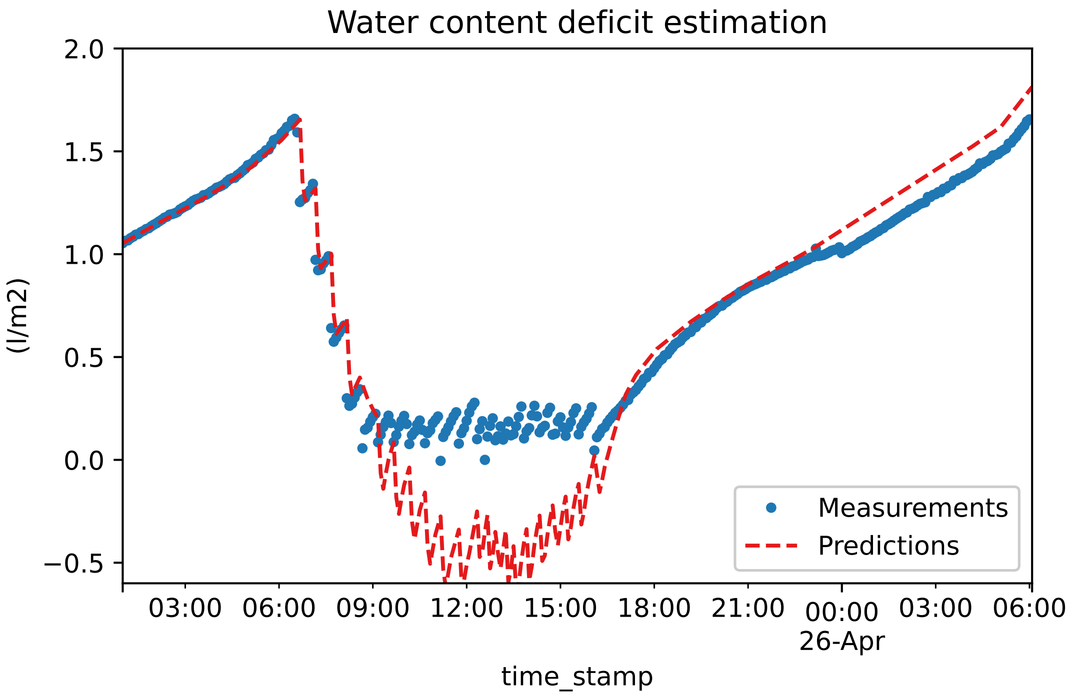

Figure 2-6: Prediction with the hybrid model 1

To counteract this anomaly, it is worthwile to look at the predicted drain compared to the actual drain. Figure 2-7 shows, how the drain prediction changes compared to the actual measured drain. It can be seen, that the predicted drain is less than the actual drain, when the water deficit measurements are close to 0.

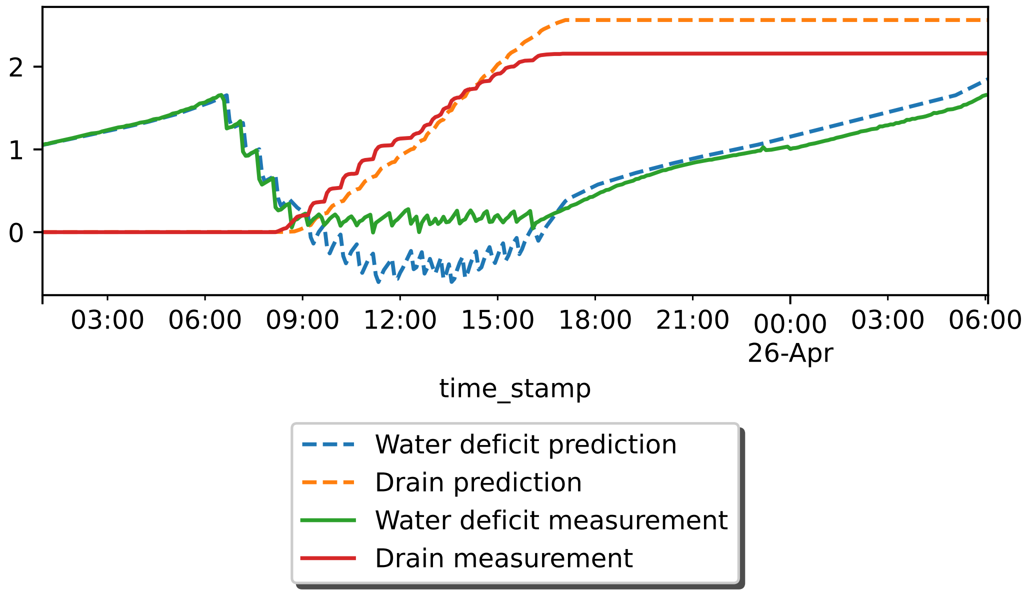

Figure 2-7: Prediction with the hybrid model 2

To enforce the non-negativity of the water deficit curve and still maintain the water balance equation, the only possibility is to lead away the excess irrigation water as drain. Similarly to the water bucket model, three linear models can describe the system in this case. The first model is active above a water deficit threshold, $\bar{y}_{\mathrm{WD}}$. Here, just as in the previous hybrid model, no drain happens, and the change in water deficit is the difference between plant water uptake and irrigation. Additionally, the influence of irrigation and plant water uptake is not considered on the model of the drain:

$$\begin{aligned}
\underbrace{\begin{bmatrix} y_{\mathrm{WD}}(k+1) \\ \boldsymbol{x}_{\mathrm{DRN}}(k+1) \end{bmatrix}}_{\boldsymbol{x}(k+1)} &= \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & A_{\mathrm{DRN}} \end{bmatrix}}_{A_1} \underbrace{\begin{bmatrix} y_{\mathrm{WD}}(k) \\ \boldsymbol{x}_{\mathrm{DRN}}(k) \end{bmatrix}}_{\boldsymbol{x}(k)} + \underbrace{\begin{bmatrix} -1 \\ 0 \end{bmatrix}}_{B_1} u(k) + \underbrace{\begin{bmatrix} 1 \\ 0 \end{bmatrix}}_{G_1} w(k) \\
\begin{bmatrix} y_{\mathrm{WD}}(k) \\ y_{\mathrm{DRN}}(k) \end{bmatrix} &= \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}}_{C_1} \begin{bmatrix} y_{\mathrm{WD}}(k) \\ \boldsymbol{x}_{\mathrm{DRN}}(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 0 \end{bmatrix}}_{D_1} u(k) + \underbrace{\begin{bmatrix} 0 \\ 0 \end{bmatrix}}_{W_1} w(k)
\end{aligned} \tag{2-46}$$

The second model is active, when the water deficit is below $\bar{y}_{\mathrm{WD}}$, but the following value, $y_{\mathrm{WD}}(k+1)$ is above zero. Then, the identified drain model has its effect on the water deficit as well:

$$\begin{aligned}
\underbrace{\begin{bmatrix} y_{\mathrm{WD}}(k+1) \\ \boldsymbol{x}_{\mathrm{DRN}}(k+1) \end{bmatrix}}_{\boldsymbol{x}(k+1)} &= \underbrace{\begin{bmatrix} 1 & C_{\mathrm{DRN}} \\ 0 & A_{\mathrm{DRN}} \end{bmatrix}}_{A_2} \underbrace{\begin{bmatrix} y_{\mathrm{WD}}(k) \\ \boldsymbol{x}_{\mathrm{DRN}}(k) \end{bmatrix}}_{\boldsymbol{x}(k)} + \underbrace{\begin{bmatrix} d_u - 1 \\ B_u \end{bmatrix}}_{B_2} u(k) + \underbrace{\begin{bmatrix} d_w + 1 \\ B_w \end{bmatrix}}_{G_2} w(k) \\
\begin{bmatrix} y_{\mathrm{WD}}(k) \\ y_{\mathrm{DRN}}(k) \end{bmatrix} &= \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & C_{\mathrm{DRN}} \end{bmatrix}}_{C_2} \begin{bmatrix} y_{\mathrm{WD}}(k) \\ \boldsymbol{x}_{\mathrm{DRN}}(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ d_u \end{bmatrix}}_{D_2} u(k) + \underbrace{\begin{bmatrix} 0 \\ d_w \end{bmatrix}}_{W_2} w(k)
\end{aligned} \tag{2-47}$$

And finally, when $y_{\mathrm{WD}}(k+1)$ is predicted to be below 0, all the excess water has to go out through drain. This means, that because $y_{\mathrm{WD}}(k+1) = y_{\mathrm{WD}}(k) - u(k) + w(k) + Q_{\mathrm{DRN}}(k)$ and $y_{\mathrm{WD}}(k+1) := 0$, $Q_{\mathrm{DRN}}(k) = u(k) - w(k) - y_{\mathrm{WD}}(k)$. So the 3rd and final system is:

$$\begin{aligned}
\underbrace{\begin{bmatrix} y_{\mathrm{WD}}(k+1) \\ \boldsymbol{x}_{\mathrm{DRN}}(k+1) \end{bmatrix}}_{\boldsymbol{x}(k+1)} &= \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & A_{\mathrm{DRN}} \end{bmatrix}}_{A_3} \underbrace{\begin{bmatrix} y_{\mathrm{WD}}(k) \\ \boldsymbol{x}_{\mathrm{DRN}}(k) \end{bmatrix}}_{\boldsymbol{x}(k)} + \underbrace{\begin{bmatrix} 0 \\ B_u \end{bmatrix}}_{B_3} u(k) + \underbrace{\begin{bmatrix} 0 \\ B_w \end{bmatrix}}_{G_3} w(k) \\
\begin{bmatrix} y_{\mathrm{WD}}(k) \\ y_{\mathrm{DRN}}(k) \end{bmatrix} &= \underbrace{\begin{bmatrix} 1 & 0 \\ -1 & 0 \end{bmatrix}}_{C_3} \begin{bmatrix} y_{\mathrm{WD}}(k) \\ \boldsymbol{x}_{\mathrm{DRN}}(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}_{D_3} u(k) + \underbrace{\begin{bmatrix} 0 \\ -1 \end{bmatrix}}_{W_3} w(k)
\end{aligned} \tag{2-48}$$

Let us write the predicted water deficit according to the drain model as: $\hat{f}(\boldsymbol{x}(k), u(k), w(k)) = \begin{bmatrix} 1 & C_{\mathrm{DRN}} \end{bmatrix} \boldsymbol{x}(k) + (d_u - 1)u(k) + (d_w + 1)w(k)$, and so the complete PWA representation of the system is:

$$\begin{aligned}
&\boldsymbol{x}(k+1) = A_i \boldsymbol{x}(k) + B_i u(k) + G_i w(k) \\
&\begin{bmatrix} y_{\mathrm{WD}}(k) \\ y_{\mathrm{DRN}}(k) \end{bmatrix} = C_i \boldsymbol{x}(k) + D_i u(k) + W_i w(k) \\
&i = \begin{cases} 1 & \text{IF: } y_{\mathrm{WD}}(k) > \bar{y}_{\mathrm{WD}} \text{ AND } \hat{f}(\boldsymbol{x}(k), u(k), w(k)) \geq 0 \\ 2 & \text{IF: } y_{\mathrm{WD}}(k) <= \bar{y}_{\mathrm{WD}} \text{ AND } \hat{f}(\boldsymbol{x}(k), u(k), w(k)) \geq 0 \\ 3 & \text{IF: } \hat{f}(\boldsymbol{x}(k), u(k), w(k)) < 0 \end{cases}
\end{aligned} \tag{2-49}$$

2-4-3 MLD formulation of the PWA model

The PWA model can be transformed into an Mixed Logical Dynamical (MLD) formulation, with the use of the logical rules (modes) introduced in (2-49). Looking now at the original MLD model with $\delta_1 = \delta$:

$$\begin{aligned}
\boldsymbol{x}_{\mathrm{DRN}}(k+1) &= A_{\mathrm{DRN}} \boldsymbol{x}_{\mathrm{DRN}}(k) + B_u u(k) + B_w w(k) \\
y_{\mathrm{DRN}}(k) &= C_{\mathrm{DRN}} \delta_1(k) \boldsymbol{x}_{\mathrm{DRN}}(k) + d_u \delta_1(k) u(k) + d_w \delta_1(k) w(k)
\end{aligned} \tag{2-50}$$

The two additions to the model are, (I) that when $\hat{f}(\boldsymbol{x}(k), u(k), w(k)) < 0$ the drain is equal to $Q_{\mathrm{DRN}}(k) = -y_{\mathrm{WD}}(k) + u(k) - w(k)$, and (II) when $y_{\mathrm{WD}}(k) > \bar{y}_{\mathrm{WD}}$, the input $u(k)$ and the disturbance $w(k)$ are not having an effect on the model of the drain. This is modelled the following way:

$$\begin{aligned}\boldsymbol{x}_{\mathrm{DRN}}(k+1) &= A_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}(k) + B_u\delta_1(k)u(k) + B_w\delta_1(k)w(k)\\ y_{\mathrm{DRN}}(k) &= C_{\mathrm{DRN}}\delta_2(k)\delta_1(k)\boldsymbol{x}_{\mathrm{DRN}}(k) + d_u\delta_2(k)\delta_1(k)u(k) + d_w\delta_2(k)\delta_1(k)w(k)+\\ &\quad + (1-\delta_2(k))u(k) - (1-\delta_2(k))y_{\mathrm{WD}}(k) - (1-\delta_2(k))w(k)\end{aligned} \tag{2-51}$$

With the constraints:

$$\begin{aligned}\delta_1(k)\left(y^{\min} - \bar{y}_{\mathrm{WD}}\right) &\leq y_{\mathrm{WD}}(k) - \bar{y}_{\mathrm{WD}}\\ \delta_1(k)\left(y^{\max} - \bar{y}_{\mathrm{WD}} + \varepsilon\right) &\leq -y_{\mathrm{WD}}(k) + y^{\max}\\ -y^{\min}\delta_2(k) &\leq \begin{bmatrix}1 & C_{\mathrm{DRN}}\end{bmatrix}\boldsymbol{x}(k) + (d_u-1)u(k) + (d_w+1)w(k) - y^{\min}\\ -\delta_2(k)\left(y^{\max}+\varepsilon\right) &\leq -\left(\begin{bmatrix}1 & C_{\mathrm{DRN}}\end{bmatrix}\boldsymbol{x}(k) + (d_u-1)u(k) + (d_w+1)w(k)\right) - \varepsilon\end{aligned} \tag{2-52}$$

Using the method described in [28], a new binary variable δ_3 is introduced, with $\delta_3(k) = \delta_1(k)\delta_2(k)$. With it (2-51) now reads:

$$\begin{aligned}\boldsymbol{x}_{\mathrm{DRN}}(k+1) &= A_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}(k) + B_u\delta_1(k)u(k) + B_w\delta_1(k)w(k)\\ y_{\mathrm{DRN}}(k) &= C_{\mathrm{DRN}}\delta_3(k)\boldsymbol{x}_{\mathrm{DRN}}(k) + d_u\delta_3(k)u(k) + d_w\delta_3(k)w(k)+\\ &\quad + (1-\delta_2(k))u(k) - (1-\delta_2(k))y_{\mathrm{WD}}(k) - (1-\delta_2(k))w(k)\end{aligned} \tag{2-53}$$

and 3 constraints are introduced additionally to the already defined ones in (2-52):

$$\begin{aligned}-\delta_1(k) + \delta_3(k) &\leq 0\\ -\delta_2(k) + \delta_3(k) &\leq 0\\ \delta_1(k) + \delta_2(k) - \delta_3(k) &\leq 1\end{aligned} \tag{2-54}$$

Similarly to Section 2-4-1, auxiliary real variables are introduced:

$$\boldsymbol{z}(k) = \begin{bmatrix} z_1(k)\\ z_2(k)\\ z_3(k)\\ z_4(k)\\ z_5(k)\\ z_6(k)\\ z_7(k)\\ z_8(k)\end{bmatrix} := \begin{bmatrix} \delta_1(k)u(k)\\ \delta_1(k)w(k)\\ \delta_2(k)u(k)\\ \delta_2(k)y_{\mathrm{WD}}(k)\\ \delta_2(k)w(k)\\ \delta_3(k)C_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}(k)\\ \delta_3(k)u(k)\\ \delta_3(k)w(k)\end{bmatrix} \tag{2-55}$$

And so (2-53) gets the following form:

$$\begin{aligned}\boldsymbol{x}_{\mathrm{DRN}}(k+1) &= A_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}(k) + B_u z_1(k) + B_w z_2(k)\\ y_{\mathrm{DRN}}(k) &= z_6(k) + d_u z_7(k) + d_w z_8(k)+\\ &\quad + u(k) - y_{\mathrm{WD}}(k) - w(k) - z_3(k) + z_4(k) + z_5(k)\end{aligned} \tag{2-56}$$

and the following constraints are introduced to keep the real valued auxiliary variables connected to the auxiliary binary variables and the states and inputs:

$$
\begin{aligned}
&\underbrace{\begin{bmatrix} z_1(k)\\ z_2(k)\\ z_3(k)\\ z_4(k)\\ z_5(k)\\ z_6(k)\\ z_7(k)\\ z_8(k)\end{bmatrix}}_{\boldsymbol{z}(k)} \leq \underbrace{\begin{bmatrix} u^{\max} & 0 & 0\\ w^{\max} & 0 & 0\\ 0 & u^{\max} & 0\\ 0 & y_{\mathrm{WD}}^{\max} & 0\\ 0 & w^{\max} & 0\\ 0 & 0 & C_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}^{\max}\\ 0 & 0 & u^{\max}\\ 0 & 0 & w^{\max}\end{bmatrix}}_{E_{4,1}} \underbrace{\begin{bmatrix}\delta_1(k)\\ \delta_2(k)\\ \delta_3(k)\end{bmatrix}}_{\boldsymbol{\delta}(k)}\\
&\boldsymbol{z}(k) \geq \underbrace{\begin{bmatrix} u^{\min} & 0 & 0\\ w^{\min} & 0 & 0\\ 0 & u^{\min} & 0\\ 0 & y_{\mathrm{WD}}^{\min} & 0\\ 0 & w^{\min} & 0\\ 0 & 0 & C_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}^{\min}\\ 0 & 0 & u^{\min}\\ 0 & 0 & w^{\min}\end{bmatrix}}_{E_{4,2}} \boldsymbol{\delta}(k)\\
&\boldsymbol{z}(k) \leq \underbrace{\begin{bmatrix} 0 & 0\\ 0 & 0\\ 0 & 0\\ 1 & 0\\ 0 & 0\\ 0 & C_{\mathrm{DRN}}\\ 0 & 0\\ 0 & 0\end{bmatrix}}_{E_{1,1}} \begin{bmatrix} y_{\mathrm{WD}}(k)\\ \boldsymbol{x}_{\mathrm{DRN}}(k)\end{bmatrix} + \underbrace{\begin{bmatrix}1\\0\\1\\0\\0\\0\\1\\0\end{bmatrix}}_{E_{2,1}} u(k) + \underbrace{\begin{bmatrix}0\\1\\0\\0\\1\\0\\0\\1\end{bmatrix}}_{E_{3,1}} w(k) - E_{4,2} \underbrace{\begin{bmatrix}1\\1\\1\end{bmatrix}}_{\boldsymbol{1}} + E_{4,2}\boldsymbol{\delta}(k))\\
&\boldsymbol{z}(k) \geq E_{1,1} \begin{bmatrix} y_{\mathrm{WD}}(k)\\ \boldsymbol{x}_{\mathrm{DRN}}(k)\end{bmatrix} + E_{2,1}u(k) + E_{3,1}w(k) - E_{4,1}\boldsymbol{1} + E_{4,1}\boldsymbol{\delta}(k))
\end{aligned}
\tag{2-57}
$$

The MLD model is restructured:

$$
\begin{aligned}
\boldsymbol{x}_{\mathrm{DRN}}(k+1) &= A_{\mathrm{DRN}}\boldsymbol{x}_{\mathrm{DRN}}(k) + \underbrace{\begin{bmatrix} B_u & B_w & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}}_{B_z} \boldsymbol{z}(k)\\
y_{\mathrm{DRN}}(k) &= u(k) - y_{\mathrm{WD}}(k) - w(k) + \underbrace{\begin{bmatrix} 0 & 0 & -1 & 1 & 1 & 1 & d_u & d_w\end{bmatrix}}_{C_z} \boldsymbol{z}(k)
\end{aligned}
\tag{2-58}
$$

Now that the MLD drain model is fully defined, it is combined with the water balance equation:

$$\begin{aligned}
\underbrace{\begin{bmatrix} y_{\mathrm{WD}}(k+1) \\ \boldsymbol{x}_{\mathrm{DRN}}(k+1) \end{bmatrix}}_{\boldsymbol{x}(k+1)} &= \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & A_{\mathrm{DRN}} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} y_{\mathrm{WD}}(k) \\ \boldsymbol{x}_{\mathrm{DRN}}(k) \end{bmatrix}}_{\boldsymbol{x}(k)} + \underbrace{\begin{bmatrix} 0 \\ 0 \end{bmatrix}}_{B_1} u(k) + \underbrace{\begin{bmatrix} C_z \\ B_z \end{bmatrix}}_{B_3} \boldsymbol{z}(k) + \underbrace{\begin{bmatrix} 0 \\ 0 \end{bmatrix}}_{G} w(k) \\
\begin{bmatrix} y_{\mathrm{WD}}(k) \\ y_{\mathrm{DRN}}(k) \end{bmatrix} &= \underbrace{\begin{bmatrix} 1 & 0 \\ -1 & 0 \end{bmatrix}}_{C} \begin{bmatrix} y_{\mathrm{WD}}(k) \\ \boldsymbol{x}_{\mathrm{DRN}}(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}_{D_1} u(k) + \underbrace{\begin{bmatrix} 0 \\ C_z \end{bmatrix}}_{D_3} \boldsymbol{z}(k) + \underbrace{\begin{bmatrix} 0 \\ -1 \end{bmatrix}}_{W} w(k)
\end{aligned} \tag{2-59}$$

The constraints from (2-52) are further transformed:

$$\begin{gathered}
E_{4,3} = \begin{bmatrix} y^{\min} - \bar{y}_{\mathrm{WD}} & 0 & 0 \\ y^{\max} - \bar{y}_{\mathrm{WD}} + \varepsilon & 0 & 0 \\ 0 & -y^{\min} & 0 \\ 0 & -\left(y^{\max} + \varepsilon\right) & 0 \end{bmatrix}, \quad \boldsymbol{g}_{6,1} = \begin{bmatrix} -\bar{y}_{\mathrm{WD}} \\ y^{\max} \\ -y^{\min} \\ -\varepsilon \end{bmatrix} \\
E_{1,2} = \begin{bmatrix} -\begin{bmatrix} 1 & 0 \end{bmatrix} \\ \begin{bmatrix} 1 & 0 \end{bmatrix} \\ -\begin{bmatrix} 1 & C_{\mathrm{DRN}} \end{bmatrix} \\ \begin{bmatrix} 1 & C_{\mathrm{DRN}} \end{bmatrix} \end{bmatrix}, \quad E_{2,2} = \begin{bmatrix} 0 \\ 0 \\ -\left(d_u - 1\right) \\ \left(d_u - 1\right) \end{bmatrix}, \quad E_{3,2} = \begin{bmatrix} 0 \\ 0 \\ -\left(d_w - 1\right) \\ \left(d_w - 1\right) \end{bmatrix}
\end{gathered} \tag{2-60}$$

As well as (2-54):

$$E_{4,4} = \begin{bmatrix} -1 & 0 & 1 \\ 0 & -1 & 1 \\ 1 & 1 & -1 \end{bmatrix}, \quad \boldsymbol{g}_{6,2} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{2-61}$$

The constraining inequalities are also transformed into a linear algebraic form:

$$\underbrace{\begin{bmatrix} 0 \\ 0 \\ -E_{1,1} \\ E_{1,1} \\ E_{1,2} \\ 0 \end{bmatrix}}_{E_1} \boldsymbol{x}(k) + \underbrace{\begin{bmatrix} 0 \\ 0 \\ -E_{2,1} \\ E_{2,1} \\ E_{2,2} \\ 0 \end{bmatrix}}_{E_2} u(k) + \underbrace{\begin{bmatrix} 0 \\ 0 \\ -E_{3,1} \\ E_{3,1} \\ E_{3,2} \\ 0 \end{bmatrix}}_{E_3} w(k) + \underbrace{\begin{bmatrix} -E_{4,1} \\ E_{4,2} \\ -E_{4,2} \\ E_{4,1} \\ E_{4,3} \\ E_{4,4} \end{bmatrix}}_{\boldsymbol{E}_4} \boldsymbol{\delta}(k) + \underbrace{\begin{bmatrix} I \\ -I \\ I \\ -I \\ 0 \\ 0 \end{bmatrix}}_{E_5} \boldsymbol{z}(k) \leq \underbrace{\begin{bmatrix} 0 \\ 0 \\ -E_{4,2}\mathbf{1} \\ E_{4,1}\mathbf{1} \\ \boldsymbol{g}_{6,1} \\ \boldsymbol{g}_{6,2} \end{bmatrix}}_{\boldsymbol{g}_6} \tag{2-62}$$

And so the final system is constructed:

$$\begin{aligned}
\boldsymbol{x}(k+1) &= A\boldsymbol{x}(k) + B_1 u(k) + B_3 \boldsymbol{z}(k) + G w(k) \\
\begin{bmatrix} y_{\mathrm{WD}}(k) \\ y_{\mathrm{DRN}}(k) \end{bmatrix} &= C\boldsymbol{x}(k) + D_1 u(k) + D_3 \boldsymbol{z}(k) + W w(k) \\
\text{s. t.} \quad & E_1 \boldsymbol{x}(k) + E_2 u(k) + E_3 w(k) + E_4 \delta(k) + E_5 \boldsymbol{z}(k) \leq \boldsymbol{g}_6
\end{aligned} \tag{2-63}$$

The system has 3 auxiliary binary variables and $4 \cdot 8 + 4 + 3 = 39$ constraints.

2-5 Summary

The previous sections described the modelling decisions which were made during the project. In Section 2-2 a linear and a bi-linear PWU model was identified, from which the bi-linear model was chosen for further work. Section 2-3 showed that the 5 minute sampling time of the data makes the identification of a physics based model troublesome, and so a linear dynamical ARX model was identified to model the drain below a water deficit threshold, which was then transformed into a State-Space (SS) model. In Section 2-4 the water balance equation was recreated, which was then modified to incorporate the zero-drain assumption (Assumption 2.2) in Section 2-4-1, and the additional non-zero water deficit rule in Section 2-4-2. Section 2-4-3 showed the transformation of the PWA model into an MLD form. The developed MLD formulations are suitable to use for optimal control tasks, because their evolution can be described through linear equations with the help of the auxiliary variables, while the inequality constraints ensure, that the models behave according to the defined rules.

Chapter 3

MPC

The main idea of this thesis project is to use a predictive control approach to influence the Water Deficit (WD) curve of the growing substrate. In this chapter, specific problem formulations are introduced which use the models developed in Chapter 2. The ideas and notation of the Model Predictive Control (MPC) framework used in this chapter are taken from [29].

3-1 Linear MPC

The first type of MPC considered is a regular tracking MPC for a linear system. It will be used to benchmark the performance of the Mixed Logical Dynamical (MLD) MPCs, which are going to be presented later on in this chapter. The linear model considered for this section has the same structure as described in Section 2-4, but it is fitted on the whole dataset, not just below a WD of 0.4 l/m^2. Because of this, the model is less accurate, but the control algorithm is less resource intensive to compute.

The linear system is given by:

$$\begin{aligned} \underbrace{\begin{bmatrix} y_{\mathrm{WD}}(k+1) \\ \boldsymbol{x}_{\mathrm{DRN}}(k+1) \end{bmatrix}}_{\boldsymbol{x}(k+1)} &= \underbrace{\begin{bmatrix} 1 & C_{\mathrm{DRN}} \\ 0 & A_{\mathrm{DRN}} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} y_{\mathrm{WD}}(k) \\ \boldsymbol{x}_{\mathrm{DRN}}(k) \end{bmatrix}}_{\boldsymbol{x}(k)} + \underbrace{\begin{bmatrix} d_u - 1 \\ B_u \end{bmatrix}}_{B} u(k) + \underbrace{\begin{bmatrix} 1 + d_w \\ B_w \end{bmatrix}}_{G} w(k) \\ \underbrace{\begin{bmatrix} y_{\mathrm{WD}}(k) \\ y_{\mathrm{DRN}}(k) \end{bmatrix}}_{\boldsymbol{y}(k)} &= \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & C_{\mathrm{DRN}} \end{bmatrix}}_{C} \begin{bmatrix} y_{\mathrm{WD}}(k) \\ \boldsymbol{x}_{\mathrm{DRN}}(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ d_u \end{bmatrix}}_{D} u(k) + \underbrace{\begin{bmatrix} 0 \\ d_w \end{bmatrix}}_{W} w(k) \end{aligned} \tag{3-1}$$

The optimal control problem is defined by the following cost function:

$$V_N(\boldsymbol{y}_N, \boldsymbol{u}_N) = \frac{1}{2} \sum_{k=0}^{N-1} q \left(y_{\mathrm{WD}}(k) - y_{\mathrm{ref}}(k) \right)^2 + r u^2(k) \tag{3-2}$$

Where N is the control horizon, $y_{\text{ref}}(k)$ is the reference trajectory for the WD, q is the stage cost for the tracking, and r is the cost for water usage. It should be noted, that a regular tracking MPC problem would have $u(k) - u_{\text{ref}}(k)$ in the stage cost, instead of just $u(k)$. For this problem however, the cost on the input (irrigation) is used to penalize the water usage, and so the task of the controller is to find a balance between tracking and resource usage. The cost function in (3-2) is rewritten into a linear algebraic equation:

$$V_N(\boldsymbol{y}_N, \boldsymbol{u}_N) = \frac{1}{2}\left(\boldsymbol{y}_N - \boldsymbol{y}_{\text{ref},N}\right)^T Q \left(\boldsymbol{y}_N - \boldsymbol{y}_{\text{ref},N}\right) + \frac{1}{2}\boldsymbol{u}_N^T R \boldsymbol{u}_N \tag{3-3}$$

with:

$$\begin{gathered} Q = \begin{bmatrix} \begin{bmatrix} q & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \cdots & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} q & 0 \\ 0 & 0 \end{bmatrix} & \cdots & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \vdots & \vdots & \ddots & \vdots \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \cdots & \begin{bmatrix} q & 0 \\ 0 & 0 \end{bmatrix} \end{bmatrix}, \quad R = \begin{bmatrix} r & 0 & \dots & 0 \\ 0 & r & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & r \end{bmatrix} \\ \boldsymbol{y}_N = \begin{bmatrix} \boldsymbol{y}(0) \\ \boldsymbol{y}(1) \\ \vdots \\ \boldsymbol{y}(N-1) \end{bmatrix} \quad \boldsymbol{y}_{\text{ref},N} = \begin{bmatrix} \begin{bmatrix} y_{\text{ref}}(0) \\ 0 \end{bmatrix} \\ \begin{bmatrix} y_{\text{ref}}(1) \\ 0 \end{bmatrix} \\ \vdots \\ \begin{bmatrix} y_{\text{ref}}(N-1) \\ 0 \end{bmatrix} \end{bmatrix} \end{gathered} \tag{3-4}$$

If $q > 0$, then Q is positive semi-definite, and if $r > 0$, then R is positive definite. Because $V_N(\boldsymbol{y}_N, \boldsymbol{u}_N)$ is a function of the series of outputs, it must be transformed so, that the output disappears from the cost, and only the decision variable $\boldsymbol{u}_N$, the initial state $\boldsymbol{x}_0$ and the disturbance vector $\boldsymbol{w}_N$ remains in the cost. This can be done using the following formulation:

$$
\begin{aligned}
\boldsymbol{y}_N &= P\boldsymbol{p}_N + S\boldsymbol{u}_N \\
\begin{bmatrix} \boldsymbol{y}(0) \\ \boldsymbol{y}(1) \\ \vdots \\ \boldsymbol{y}(N-1) \end{bmatrix} &= \underbrace{\begin{bmatrix} P_x & P_w \end{bmatrix}}_{P} \underbrace{\begin{bmatrix} \boldsymbol{x}_0 \\ \boldsymbol{w}_N \end{bmatrix}}_{\boldsymbol{p}_N} + S\boldsymbol{u}_N \\
\begin{bmatrix} \boldsymbol{y}(0) \\ \boldsymbol{y}(1) \\ \vdots \\ \boldsymbol{y}(N-1) \end{bmatrix} &= \underbrace{\begin{bmatrix} C \\ CA \\ \vdots \\ CA^{N-1} \end{bmatrix}}_{P_x} \boldsymbol{x}_0 + \underbrace{\begin{bmatrix} D & 0 & \dots & 0 \\ CB & D & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{N-2}B & CA^{N-3}B & \dots & D \end{bmatrix}}_{S} \begin{bmatrix} u(0) \\ u(1) \\ \vdots \\ u(N-1) \end{bmatrix} \\
&+ \underbrace{\begin{bmatrix} W & 0 & \dots & 0 \\ CG & W & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{N-2}G & CA^{N-3}G & \dots & W \end{bmatrix}}_{P_w} \begin{bmatrix} w(0) \\ w(1) \\ \vdots \\ w(N-1) \end{bmatrix}
\end{aligned}
\tag{3-5}
$$

With the use of the prediction matrices S and P, the cost function from (3-3) is reformulated:

$$
\begin{aligned}
V_N(\boldsymbol{u}_N, \boldsymbol{y}_N) &= \frac{1}{2}\left(\boldsymbol{y}_N - \boldsymbol{y}_{\text{ref},N}\right)^T Q \left(\boldsymbol{y}_N - \boldsymbol{y}_{\text{ref},N}\right) + \frac{1}{2}\boldsymbol{u}_N^T R \boldsymbol{u}_N \\
V_N(\boldsymbol{u}_N, \boldsymbol{p}_N) &= \frac{1}{2}\left(P\boldsymbol{p}_N + S\boldsymbol{u}_N - \boldsymbol{y}_{\text{ref},N}\right)^T Q \left(P\boldsymbol{p}_N + S\boldsymbol{u}_N - \boldsymbol{y}_{\text{ref},N}\right) + \frac{1}{2}\boldsymbol{u}_N^T R \boldsymbol{u}_N \\
V_N(\boldsymbol{u}_N, \boldsymbol{p}_N) &= \frac{1}{2}\left(\boldsymbol{u}_N^T S^T Q S \boldsymbol{u}_N + \boldsymbol{u}_N^T R \boldsymbol{u}_N\right) + \frac{1}{2}\left(\boldsymbol{p}_N^T P^T Q P \boldsymbol{p}_N + \boldsymbol{y}_{\text{ref},N}^T Q \boldsymbol{y}_{\text{ref},N}\right) + \\
&+ \left(\boldsymbol{p}_N^T P^T Q S - \boldsymbol{y}_{\text{ref},N}^T Q S\right)\boldsymbol{u}_N - \boldsymbol{p}_N^T P^T Q \boldsymbol{y}_{\text{ref},N} \\
V_N(\boldsymbol{u}_N, \boldsymbol{p}_N) &= \frac{1}{2}\boldsymbol{u}_N^T \underbrace{\left(S^T Q S + R\right)}_{H} \boldsymbol{u}_N + \underbrace{\left(\boldsymbol{p}_N^T P^T Q S - \boldsymbol{y}_{\text{ref},N}^T Q S\right)}_{\boldsymbol{h}^T} \boldsymbol{u}_N + \\
&+ \underbrace{\frac{1}{2}\left(\boldsymbol{p}_N^T P^T Q P \boldsymbol{p}_N + \boldsymbol{y}_{\text{ref},N}^T Q \boldsymbol{y}_{\text{ref},N} - 2\boldsymbol{p}_N^T P^T Q \boldsymbol{y}_{\text{ref},N}\right)}_{c} \\
V_N(\boldsymbol{u}_N, \boldsymbol{p}_N) &= \frac{1}{2}\boldsymbol{u}_N^T H \boldsymbol{u}_N + \boldsymbol{h}^T \boldsymbol{u}_N + c
\end{aligned}
\tag{3-6}
$$

The resulting optimal control problem ($\min_{u_N} V_N$) is convex if H is positive-semi definite.

3-1-1 Constraints

One of the big advantages of MPC compared to other linear control approaches is that constraints can be imposed on the inputs, states and outputs. First, the input constraints are considered.

The amount of irrigation which can be applied can not be negative. This means, that $\underline{u} = 0$. We can also impose an upper limit on irrigation, which is denoted by $\bar{u}$. The formulation of these constraints is:

$$\begin{aligned} A_u \boldsymbol{u}_N &\leq \boldsymbol{b}_u \\ \begin{bmatrix} -I \\ I \end{bmatrix} \boldsymbol{u}_N &\leq \begin{bmatrix} -\underline{u}\mathbf{1} \\ \bar{u}\mathbf{1} \end{bmatrix} \end{aligned} \tag{3-7}$$

Where $\mathbf{1} = [1, 1, \dots, 1]^T$ of appropriate size. Constraints can also be imposed on the water deficit and drain, $\boldsymbol{y}(k)$. One constraint is non-negativity $\underline{\boldsymbol{y}} = [0, 0]^T$. A maximum constraint can also be imposed to avoid drying out the substrate or flushing it out too much, $\bar{\boldsymbol{y}}$. Because $\boldsymbol{y}_N$ can be predicted using the already defined matrices, the constraint can be written as:

$$\begin{aligned} \tilde{A}_y \boldsymbol{y}_N &\leq \boldsymbol{b}_y \\ \begin{bmatrix} -I \\ I \end{bmatrix} \boldsymbol{y}_N &\leq \begin{bmatrix} -(\mathbf{1} \otimes I)\underline{\boldsymbol{y}} \\ (\mathbf{1} \otimes I)\bar{\boldsymbol{y}} \end{bmatrix} \\ \tilde{A}_y \left(P \boldsymbol{p}_N + S \boldsymbol{u}_N \right) &\leq \boldsymbol{b}_y \\ \underbrace{\tilde{A}_y S}_{A_y} \boldsymbol{u}_N &\leq \boldsymbol{b}_y - \underbrace{\tilde{A}_y P}_{F_y} \boldsymbol{p}_N \\ A_y \boldsymbol{u}_N &\leq \boldsymbol{b}_y - F_y \boldsymbol{p}_N \end{aligned} \tag{3-8}$$

Where $\otimes$ denotes the Kronecker product of two matrices. Besides the minimum and maximum constraints, the total amount of irrigation and drain can also be restricted. Defining $\bar{u}_{\text{TOT}}$ and $\bar{y}_{\text{TOT, DRN}}$ as the total irrigation and drain respectively in l/m^2 for the prediction horizon:

$$\mathbf{1}^T \boldsymbol{u}_N \leq \bar{u}_{\text{TOT}} \tag{3-9}$$

and

$$\begin{aligned} (\mathbf{1}^T \otimes \begin{bmatrix} 0 & 1 \end{bmatrix}) \boldsymbol{y}_N &\leq \bar{y}_{\text{TOT, DRN}} \\ (\mathbf{1}^T \otimes \begin{bmatrix} 0 & 1 \end{bmatrix}) \left(P \boldsymbol{p}_N + S \boldsymbol{u}_N \right) &\leq \bar{y}_{\text{TOT, DRN}} \\ \underbrace{(\mathbf{1}^T \otimes \begin{bmatrix} 0 & 1 \end{bmatrix}) S}_{A_{\text{TOT},y}} \boldsymbol{u}_N &\leq \bar{y}_{\text{TOT, DRN}} - \underbrace{(\mathbf{1}^T \otimes \begin{bmatrix} 0 & 1 \end{bmatrix}) P}_{F_{\text{TOT},y}} \boldsymbol{p}_N \end{aligned} \tag{3-10}$$

The optimal control problem is extended with the constraints:

$$\begin{aligned} &\min_{\boldsymbol{u}_N} V_N(\boldsymbol{u}_N, \boldsymbol{p}_N) = \frac{1}{2} \boldsymbol{u}_N^T H \boldsymbol{u}_N + \boldsymbol{h}^T \boldsymbol{u}_N + c \\ &\text{s. t.} \quad \underbrace{\begin{bmatrix} A_u \\ A_y \\ \mathbf{1}^T \\ A_{\text{TOT},y} \end{bmatrix}}_{A_{\text{ieq}}} \boldsymbol{u}_N \leq \underbrace{\begin{bmatrix} \boldsymbol{b}_u \\ \boldsymbol{b}_y \\ \bar{u}_{\text{TOT}} \\ \bar{y}_{\text{TOT, DRN}} \end{bmatrix}}_{\boldsymbol{b}_{\text{ieq}}} - \underbrace{\begin{bmatrix} 0 \\ F_y \\ 0 \\ F_{\text{TOT}, y} \end{bmatrix}}_{F_{\text{ieq}}} \boldsymbol{p}_N \end{aligned} \tag{3-11}$$

Because the constraints are linear, the problem stays convex, and so convex optimization can be used to solve it without the need of a multi-start approach.

3-1-2 Move-blocking

Considering, that $h_1 = 5\text{min}$, A control horizon of 48 hours is $N = 60/5 \cdot 48 = 576$ steps, and so 576 free optimization variables (irrigation inputs). Because the accuracy of the predictions drops going further into the future, reducing the granularity of predictions in the future could reduce the computational complexity. 5 minute granularity can be maintained until 2-3 hours ahead, and then an hourly amount of irrigation should be calculated which would be applied in a fixed pattern. Let us now denote the total prediction horizon in hours by: H_t, e.g. $H_\text{t} = 48$. Let $h_1 = 5\text{min}$ and $h_2 = 60\text{min}$. Let the hours of fine granularity ($h = h_1$) be denoted by H_f. Now we can calculate the number of predictions by: $N_\text{t} = H_\text{f} \cdot h_2/h_1 + H_\text{t} - H_\text{f} = (h_2/h_1 - 1)H_\text{f} + H_\text{t}$. The prediction problem has to be restructured for the different sampling times. Using the same notation as before:

$$\boldsymbol{y}_N = P\boldsymbol{p}_N + S\boldsymbol{u}_N \tag{3-12}$$

But now with $N_\text{f} = H_\text{f} \cdot h_2/h_1$, the input vector is:

$$\tilde{\boldsymbol{u}}_N = \begin{bmatrix} u(0) \\ u(1) \\ \vdots \\ u(N_\text{f} - 1) \\ \tilde{u}(N_\text{f}) \\ \tilde{u}(N_\text{f} + 1) \\ \vdots \\ \tilde{u}(N_\text{t} - 1) \end{bmatrix} \tag{3-13}$$

Where $u(k)$ is the amount of irrigation for the upcoming h_1 minutes if $k < N_\text{f}$, and $\tilde{u}(k)$ is the amount of irrigation for the upcoming h_2 minutes if $k \geq N_\text{f}$. When $k \geq N_\text{f}$, the irrigation input $\tilde{u}(k)$ is per hour, and so a matrix is used to distribute the irrigation in a fixed pattern, e.g. every 10 minutes:

$$\begin{aligned} \boldsymbol{u}_{k \cdot h_2, (k+1) \cdot h_2} &= T\tilde{u}(k) \\ \begin{bmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k + h_2 - 1) \end{bmatrix} &= \underbrace{\begin{bmatrix} 2\frac{h_1}{h_2} \\ 0 \\ 2\frac{h_1}{h_2} \\ \vdots \\ 2\frac{h_1}{h_2} \\ 0 \end{bmatrix}}_{T} \tilde{u}(k) \end{aligned} \tag{3-14}$$

Now for the whole horizon:

$$
\begin{aligned}
\boldsymbol{u}_N &= \bar{T}\tilde{\boldsymbol{u}}_N \\
\begin{bmatrix} \boldsymbol{u}_{0,N_\mathrm{f}} \\ \boldsymbol{u}_{N_\mathrm{f},N_\mathrm{f}+h_2} \\ \vdots \\ \boldsymbol{u}_{N_\mathrm{t}-h_2,N_\mathrm{t}} \end{bmatrix} &= \underbrace{\begin{bmatrix} I & 0 & \dots & 0 \\ 0 & T & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & T \end{bmatrix}}_{\bar{T}} \begin{bmatrix} \boldsymbol{u}_{0,N_\mathrm{f}} \\ \tilde{u}(N_\mathrm{f}) \\ \vdots \\ \tilde{u}(N_\mathrm{t}-1) \end{bmatrix}
\end{aligned} \tag{3-15}
$$

And so the only change to the predictions is in the S matrix:

$$
\boldsymbol{y}_N = P\boldsymbol{p}_N + S\bar{T}\tilde{\boldsymbol{u}}_N \tag{3-16}
$$

And by the constraints the A_ieq matrix:

$$
A_\mathrm{ieq}\bar{T}\boldsymbol{u}_N \leq \boldsymbol{b}_\mathrm{ieq} - F_\mathrm{ieq}\boldsymbol{p}_N \tag{3-17}
$$

With the introduced changes, the decision variables of the problem can be reduced significantly. E.g. with $H_\mathrm{t} = 48$ hours and $H_\mathrm{f} = 3$ hours, $N_\mathrm{t} = (60/5-1)\cdot 3 + 48 = 81$ samples and free decision variables, compared to the original 576 free decision variables.

3-2 Hybrid MPC

To use the model from Section 2-4-1, the problem formulation from (3-3) is used, which is a combined economic and tracking problem. The MLD model from (2-45) has 5 different variables: $\boldsymbol{x}$ is the real-valued state vector of the system, u is the irrigation input, w is the Plant Water Uptake (PWU). The last two variables are auxiliary, meaning that they have been introduced because of the MLD framework. $\boldsymbol{\delta}$ is an auxiliary binary variable vector and $\boldsymbol{z}$ is a real-valued auxiliary vector. For prediction purposes, it is assumed, that the initial state of the system $\boldsymbol{x}_0$ is known, as well as the predicted disturbance vector $\boldsymbol{w}_N$. The irrigation input and auxiliary variables are however calculated through optimization, where they are the decision variables. Because the MLD water balance system is deterministic, $\boldsymbol{y}_N$ can be written as a function of the optimization variables $\boldsymbol{v}_N = \left[\boldsymbol{u}_N^T, \boldsymbol{\delta}_N^T, \boldsymbol{z}_N^T\right]^T$ and the fixed parameters: initial state $\boldsymbol{x}_0$ and series of disturbances $\boldsymbol{w}_N$.

$$
\begin{aligned}
\boldsymbol{y}_N &= P\boldsymbol{p}_N + S\boldsymbol{v}_N \\
\begin{bmatrix} \boldsymbol{y}(0) \\ \boldsymbol{y}(1) \\ \vdots \\ \boldsymbol{y}(N-1) \end{bmatrix} &= \underbrace{\begin{bmatrix} P_x & P_w \end{bmatrix}}_{P} \underbrace{\begin{bmatrix} \boldsymbol{x}_0 \\ \boldsymbol{w}_N \end{bmatrix}}_{\boldsymbol{p}_N} + \underbrace{\begin{bmatrix} S_u & 0 & S_z \end{bmatrix}}_{S} \underbrace{\begin{bmatrix} \boldsymbol{u}_N \\ \boldsymbol{\delta}_N \\ \boldsymbol{z}_N \end{bmatrix}}_{\boldsymbol{v}_N} \\
\begin{bmatrix} \boldsymbol{y}(0) \\ \boldsymbol{y}(1) \\ \vdots \\ \boldsymbol{y}(N-1) \end{bmatrix} &= \underbrace{\begin{bmatrix} C \\ CA \\ \vdots \\ CA^{N-1} \end{bmatrix}}_{P_x} \boldsymbol{x}_0 + \underbrace{\begin{bmatrix} D_1 & 0 & \dots & 0 \\ CB_1 & D_1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{N-2}B_1 & CA^{N-3}B_1 & \dots & D_1 \end{bmatrix}}_{S_u} \begin{bmatrix} u(0) \\ u(1) \\ \vdots \\ u(N-1) \end{bmatrix} + \\
&+ \underbrace{\begin{bmatrix} D_3 & 0 & \dots & 0 \\ CB_3 & D_3 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{N-2}B_3 & CA^{N-3}B_3 & \dots & D_3 \end{bmatrix}}_{S_z} \begin{bmatrix} \boldsymbol{z}(0) \\ \boldsymbol{z}(1) \\ \vdots \\ \boldsymbol{z}(N-1) \end{bmatrix} + \\
&+ \underbrace{\begin{bmatrix} W & 0 & \dots & 0 \\ CG & W & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{N-2}G & CA^{N-3}G & \dots & W \end{bmatrix}}_{P_w} \begin{bmatrix} w(0) \\ w(1) \\ \vdots \\ w(N-1) \end{bmatrix}
\end{aligned}
\tag{3-18}
$$

The cost function has the same form as in (3-11), with $\boldsymbol{u}_N$ replaced by $\boldsymbol{v}_N$, and with a modified $\bar{R}$ matrix:

$$
\begin{aligned}
V_N(\boldsymbol{v}_N, \boldsymbol{p}_N) &= \frac{1}{2}\boldsymbol{v}_N^T \underbrace{\left(S^T Q S + \bar{R}\right)}_{H} \boldsymbol{v}_N + \underbrace{\left(\boldsymbol{p}_N^T P^T Q S - \boldsymbol{y}_{\text{ref},N}^T Q S\right)}_{\boldsymbol{h}^T} \boldsymbol{v}_N + \\
&+ \underbrace{\frac{1}{2}\left(\boldsymbol{p}_N^T P^T Q P \boldsymbol{p}_N + \boldsymbol{y}_{\text{ref},N}^T Q \boldsymbol{y}_{\text{ref},N} - 2\boldsymbol{p}_N^T P^T Q \boldsymbol{y}_{\text{ref},N}\right)}_{c} \\
V_N(\boldsymbol{v}_N, \boldsymbol{p}_N) &= \frac{1}{2}\boldsymbol{v}_N^T H \boldsymbol{v}_N + \boldsymbol{h}^T \boldsymbol{v}_N + c
\end{aligned}
\tag{3-19}
$$

with

$$
\bar{R} = \begin{bmatrix} R & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}
\tag{3-20}
$$

The constraints defining the auxiliary variables are extended for the whole horizon N:

$$
\begin{aligned}
&\tilde{E}_1\boldsymbol{x}_N + \tilde{E}_2\boldsymbol{u}_N + \tilde{E}_3\boldsymbol{w}_N + \tilde{E}_4\boldsymbol{\delta}_N + \tilde{E}_5\boldsymbol{z}_N \le \tilde{\boldsymbol{g}}_6 \\
&\text{s. t.} \quad \tilde{E}_i = \begin{bmatrix} E_i & 0 & \dots & 0 \\ 0 & E_i & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & E_i \end{bmatrix} \quad \text{for} \quad i = 1, \dots, 5 \quad \text{and} \quad \tilde{\boldsymbol{g}}_6 = \begin{bmatrix} \boldsymbol{g}_6 \\ \boldsymbol{g}_6 \\ \vdots \\ \boldsymbol{g}_6 \end{bmatrix}
\end{aligned} \tag{3-21}
$$

As $\boldsymbol{y}_N$, $\boldsymbol{x}_N$ can also be defined as a linear combination of the other variables:

$$
\begin{aligned}
&\boldsymbol{x}_N = \tilde{P}\boldsymbol{p}_N + \tilde{S}\boldsymbol{v}_N \\
&\begin{bmatrix} \boldsymbol{x}(0) \\ \boldsymbol{x}(1) \\ \vdots \\ \boldsymbol{x}(N-1) \end{bmatrix} = \underbrace{\begin{bmatrix} \tilde{P}_x & \tilde{P}_w \end{bmatrix}}_{\tilde{P}} \underbrace{\begin{bmatrix} \boldsymbol{x}_0 \\ \boldsymbol{w}_N \end{bmatrix}}_{\boldsymbol{p}_N} + \underbrace{\begin{bmatrix} \tilde{S}_u & 0 & \tilde{S}_z \end{bmatrix}}_{\tilde{S}} \underbrace{\begin{bmatrix} \boldsymbol{u}_N \\ \boldsymbol{\delta}_N \\ \boldsymbol{z}_N \end{bmatrix}}_{\boldsymbol{v}_N} \\
&\boldsymbol{x}_N = \underbrace{\begin{bmatrix} I \\ A \\ \vdots \\ A^{N-1} \end{bmatrix}}_{\tilde{P}_x} \boldsymbol{x}_0 + \underbrace{\begin{bmatrix} 0 & 0 & \dots & 0 \\ B_1 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-2}B_1 & A^{N-3}B_1 & \dots & 0 \end{bmatrix}}_{\tilde{S}_u} \boldsymbol{u}_N + \\
&\quad + \underbrace{\begin{bmatrix} 0 & 0 & \dots & 0 \\ B_3 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-2}B_3 & A^{N-3}B_3 & \dots & 0 \end{bmatrix}}_{\tilde{S}_z} \boldsymbol{z}_N + \underbrace{\begin{bmatrix} 0 & 0 & \dots & 0 \\ G & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-2}G & A^{N-3}G & \dots & 0 \end{bmatrix}}_{\tilde{P}_w} \boldsymbol{w}_N \\
&\boldsymbol{x}_N = \tilde{P}_x\boldsymbol{x}_0 + \tilde{S}_u\boldsymbol{u}_N + \tilde{S}_z\boldsymbol{z}_N + \tilde{P}_w\boldsymbol{w}_N
\end{aligned} \tag{3-22}
$$

Rewriting the inequalities for $\boldsymbol{v}_N$ and $\boldsymbol{p}_N$:

$$
\begin{aligned}
&\tilde{E}_1\boldsymbol{x}_N + \tilde{E}_2\boldsymbol{u}_N + \tilde{E}_3\boldsymbol{w}_N + \tilde{E}_4\boldsymbol{\delta}_N + \tilde{E}_5\boldsymbol{z}_N \le \tilde{\boldsymbol{g}}_6 \\
&\tilde{E}_1\left(\tilde{P}_x\boldsymbol{x}_0 + \tilde{S}_u\boldsymbol{u}_N + \tilde{S}_z\boldsymbol{z}_N + \tilde{P}_w\boldsymbol{w}_N\right) + \tilde{E}_2\boldsymbol{u}_N + \tilde{E}_3\boldsymbol{w}_N + \tilde{E}_4\boldsymbol{\delta}_N + \tilde{E}_5\boldsymbol{z}_N \le \tilde{\boldsymbol{g}}_6 \\
&\tilde{E}_1\tilde{P}_x\boldsymbol{x}_0 + \left(\tilde{E}_1\tilde{S}_u + \tilde{E}_2\right)\boldsymbol{u}_N + \left(\tilde{E}_1\tilde{P}_w + \tilde{E}_3\right)\boldsymbol{w}_N + \tilde{E}_4\boldsymbol{\delta}_N + \left(\tilde{E}_1\tilde{S}_z + \tilde{E}_5\right)\boldsymbol{z}_N \le \tilde{\boldsymbol{g}}_6 \\
&\underbrace{\begin{bmatrix} \tilde{E}_1\tilde{P}_x & \tilde{E}_1\tilde{P}_w + \tilde{E}_3 \end{bmatrix}}_{F_2} \begin{bmatrix} \boldsymbol{x}_0 \\ \boldsymbol{w}_N \end{bmatrix} + \underbrace{\begin{bmatrix} \tilde{E}_1\tilde{S}_u + \tilde{E}_2 & \tilde{E}_4 & \tilde{E}_1\tilde{S}_z + \tilde{E}_5 \end{bmatrix}}_{F_1} \begin{bmatrix} \boldsymbol{u}_N \\ \boldsymbol{\delta}_N \\ \boldsymbol{z}_N \end{bmatrix} \le \tilde{\boldsymbol{g}}_6 \\
&F_1\boldsymbol{v}_N \le \tilde{\boldsymbol{g}}_6 - F_2\boldsymbol{p}_N
\end{aligned} \tag{3-23}
$$

And with this the optimal control problem is assembled:

$$\begin{aligned} &\min_{\boldsymbol{v}_N} V_N(\boldsymbol{v}_N, \boldsymbol{p}_N) = \frac{1}{2}\boldsymbol{v}_N^T H \boldsymbol{v}_N + \boldsymbol{h}^T \boldsymbol{v}_N + c \\ &\text{s. t.} \quad F_1 \boldsymbol{v}_N \leq \tilde{\boldsymbol{g}}_6 - F_2 \boldsymbol{p}_N \end{aligned} \tag{3-24}$$

Although move-blocking can only simplify the irrigation inputs, not the auxiliary variables, it can be worthwhile to introduce it for the hybrid model as well. The predictions change by:

$$\begin{gathered} \boldsymbol{y}_N = P\boldsymbol{p}_N + S\boldsymbol{v}_N \\ \begin{bmatrix} y_{\mathrm{WD}}(0) \\ y_{\mathrm{WD}}(1) \\ \vdots \\ y_{\mathrm{WD}}(N-1) \end{bmatrix} = \underbrace{\begin{bmatrix} P_x & P_w \end{bmatrix}}_{P} \underbrace{\begin{bmatrix} \boldsymbol{x}_0 \\ \boldsymbol{w}_N \end{bmatrix}}_{\boldsymbol{p}_N} + \underbrace{\begin{bmatrix} S_u \bar{T} & 0 & S_z \end{bmatrix}}_{S} \underbrace{\begin{bmatrix} \tilde{\boldsymbol{u}}_N \\ \boldsymbol{\delta}_N \\ \boldsymbol{z}_N \end{bmatrix}}_{\boldsymbol{v}_N} \end{gathered} \tag{3-25}$$

And by the constraints the $\tilde{S}_u$ matrix:

$$\begin{gathered} \boldsymbol{x}_N = \tilde{P}\boldsymbol{p}_N + \tilde{S}\boldsymbol{v}_N \\ \begin{bmatrix} \boldsymbol{x}(0) \\ \boldsymbol{x}(1) \\ \vdots \\ \boldsymbol{x}(N-1) \end{bmatrix} = \underbrace{\begin{bmatrix} \tilde{P}_x & \tilde{P}_w \end{bmatrix}}_{\tilde{P}} \underbrace{\begin{bmatrix} \boldsymbol{x}_0 \\ \boldsymbol{w}_N \end{bmatrix}}_{\boldsymbol{p}_N} + \underbrace{\begin{bmatrix} \tilde{S}_u \bar{T} & 0 & \tilde{S}_z \end{bmatrix}}_{\tilde{S}} \underbrace{\begin{bmatrix} \tilde{\boldsymbol{u}}_N \\ \boldsymbol{\delta}_N \\ \boldsymbol{z}_N \end{bmatrix}}_{\boldsymbol{v}_N} \end{gathered} \tag{3-26}$$

and the F_1 matrix:

$$\begin{aligned} &\underbrace{\begin{bmatrix} \tilde{E}_1 \tilde{P}_x & \tilde{E}_1 \tilde{P}_w + \tilde{E}_3 \end{bmatrix}}_{F_2} \begin{bmatrix} \boldsymbol{x}_0 \\ \boldsymbol{w}_N \end{bmatrix} + \underbrace{\begin{bmatrix} \left(\tilde{E}_1 \tilde{S}_u + \tilde{E}_2\right) \bar{T} & \tilde{E}_4 & \tilde{E}_1 \tilde{S}_z + \tilde{E}_5 \end{bmatrix}}_{F_1} \begin{bmatrix} \tilde{\boldsymbol{u}}_N \\ \boldsymbol{\delta}_N \\ \boldsymbol{z}_N \end{bmatrix} \leq \tilde{\boldsymbol{g}}_6 \\ &F_1 \boldsymbol{v}_N \leq \tilde{\boldsymbol{g}}_6 - F_2 \boldsymbol{p}_N \end{aligned} \tag{3-27}$$

3-2-1 Constraints

Because the constraints in Section 3-1-1 are only imposed on the inputs and outputs of the system, the change in the formulation is not substantial:

$$\begin{gathered} A_u \boldsymbol{v}_N \leq \boldsymbol{b}_u \\ \begin{bmatrix} -I & 0 & 0 \\ I & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_N \\ \boldsymbol{\delta}_N \\ \boldsymbol{z}_N \end{bmatrix} \leq \begin{bmatrix} -\underline{u}\mathbf{1} \\ \bar{u}\mathbf{1} \end{bmatrix} \end{gathered} \tag{3-28}$$

$$
\begin{aligned}
\tilde{A}_y \boldsymbol{y}_N &\leq \boldsymbol{b}_y \\
\begin{bmatrix} -I \\ I \end{bmatrix} \boldsymbol{y}_N &\leq \begin{bmatrix} -(\mathbf{1} \otimes I)\underline{\boldsymbol{y}} \\ (\mathbf{1} \otimes I)\bar{\boldsymbol{y}} \end{bmatrix} \\
\tilde{A}_y \left(P\boldsymbol{p}_N + S\boldsymbol{v}_N \right) &\leq \boldsymbol{b}_y \\
\underbrace{\tilde{A}_y S}_{A_y} \boldsymbol{v}_N &\leq \boldsymbol{b}_y - \underbrace{\tilde{A}_y P}_{F_y} \boldsymbol{p}_N \\
A_y \boldsymbol{v}_N &\leq \boldsymbol{b}_y - F_y \boldsymbol{p}_N
\end{aligned}
\tag{3-29}
$$

The constraints on the total amounts are:

$$
\begin{aligned}
A_{\text{TOT},u} \boldsymbol{v}_N &\leq \bar{u}_{\text{TOT}} \\
\underbrace{\begin{bmatrix} \mathbf{1}^T & \mathbf{0}^T & \mathbf{0}^T \end{bmatrix}}_{A_{\text{TOT},u}} \begin{bmatrix} \boldsymbol{u}_N \\ \boldsymbol{\delta}_N \\ \boldsymbol{z}_N \end{bmatrix} &\leq \bar{u}_{\text{TOT}}
\end{aligned}
\tag{3-30}
$$

and

$$
\begin{aligned}
(\mathbf{1}^T \otimes \begin{bmatrix} 0 & 1 \end{bmatrix}) \boldsymbol{y}_N &\leq \bar{y}_{\text{TOT, DRN}} \\
(\mathbf{1}^T \otimes \begin{bmatrix} 0 & 1 \end{bmatrix}) \left(P\boldsymbol{p}_N + S\boldsymbol{v}_N \right) &\leq \bar{y}_{\text{TOT, DRN}} \\
\underbrace{(\mathbf{1}^T \otimes \begin{bmatrix} 0 & 1 \end{bmatrix}) S}_{A_{\text{TOT},y}} \boldsymbol{v}_N &\leq \bar{y}_{\text{TOT, DRN}} - \underbrace{(\mathbf{1}^T \otimes \begin{bmatrix} 0 & 1 \end{bmatrix}) P}_{F_{\text{TOT},y}} \boldsymbol{p}_N
\end{aligned}
\tag{3-31}
$$

And so the constraints are assembled:

$$
\underbrace{\begin{bmatrix} A_u \\ A_y \\ A_{\text{TOT},u} \\ A_{\text{TOT},y} \end{bmatrix}}_{A_{\text{ieq}}} \boldsymbol{u}_N \leq \underbrace{\begin{bmatrix} \boldsymbol{b}_u \\ \boldsymbol{b}_y \\ \bar{u}_{\text{TOT}} \\ \bar{y}_{\text{TOT, DRN}} \end{bmatrix}}_{\boldsymbol{b}_{\text{ieq}}} - \underbrace{\begin{bmatrix} 0 \\ F_y \\ 0 \\ F_{\text{TOT}}, y \end{bmatrix}}_{F_{\text{ieq}}} \boldsymbol{p}_N
\tag{3-32}
$$

And the optimal control problem is extended with the constraints:

$$
\begin{aligned}
&\min_{\boldsymbol{v}_N} V_N(\boldsymbol{v}_N, \boldsymbol{p}_N) = \frac{1}{2} \boldsymbol{v}_N^T H \boldsymbol{v}_N + \boldsymbol{h}^T \boldsymbol{v}_N + c \\
&\text{s. t.} \quad \begin{bmatrix} F_1 \\ A_{\text{ieq}} \end{bmatrix} \boldsymbol{v}_N \leq \begin{bmatrix} \boldsymbol{g}_6 \\ \boldsymbol{b}_{\text{ieq}} \end{bmatrix} - \begin{bmatrix} F_2 \\ F_{\text{ieq}} \end{bmatrix} \boldsymbol{p}_N
\end{aligned}
\tag{3-33}
$$

The optimal control problem in (3-33) is an Mixed Integer Quadratic Programming (MIQP) optimization problem. Because the constraints are all linear, if H is positive semi-definite, the relaxed problem is convex, which is exploited by GUROBI[30] to provide a bound on the optimal solution, thus guaranteeing optimality. The formulation introduced in this section is used for both the original and the modified MLD model.

3-2-2 Linear cost function

The quadratic cost function which was introduced in (3-33) puts a big computational load on the solver. For long prediction (and control) horizons, meaning 24-48 hours, the complexity of the systems grow substantially. Although this complexity is reduced by the reduced granularity of the irrigation inputs after H_f hours, the number of auxiliary variables $\boldsymbol{\delta}_N$ and $\boldsymbol{z}_N$ stays the same. This means, that even for the simpler hybrid model with a 48 hour prediction with a sampling time of h_1= 5 min, there is 48*60/5=576 binary and 3*48*60/5=1728 real valued auxiliary variables. Considering that the MIQP is NP complete [28], which means that its complexity grows exponentially with the size of the problem, a branch-and-bound algorithm would in worst case have to solve $2^{576} \approx 10^{173}$ regular Quadratic Programming (QP) problems to arrive at the optimum. This is unfeasible. One way to circumvent this problem is a linear cost function. Instead of:

$$V_N(\boldsymbol{y}_N, \boldsymbol{u}_N) = \frac{1}{2}\sum_{k=0}^{N-1} q\left(y_\mathrm{WD}(k) - y_\mathrm{ref}(k)\right)^2 + ru^2(k) \tag{3-34}$$

the absolute value of the differences is minimised:

$$V_N(\boldsymbol{y}_N, \boldsymbol{u}_N) = \sum_{k=0}^{N-1} q \cdot |y_\mathrm{WD}(k) - y_\mathrm{ref}(k)| + r \cdot |u(k)| \tag{3-35}$$

Which is then transformed into:

$$V_N(\boldsymbol{y}_N, \boldsymbol{u}_N) = ||Q\left(\boldsymbol{y}_N - \boldsymbol{y}_{\mathrm{ref},N}\right)||_1 + \cdot ||R\boldsymbol{u}_N||_1 \tag{3-36}$$

And with $\boldsymbol{y}_N = P\boldsymbol{p}_N + S\boldsymbol{v}_N$:

$$\begin{aligned} \min_{\boldsymbol{v}_N} V_N(\boldsymbol{v}_N, \boldsymbol{p}_N) &= ||Q\left(P\boldsymbol{p}_N + S\boldsymbol{v}_N - \boldsymbol{y}_{\mathrm{ref},N}\right)||_1 + \left|\left|\bar{R}\boldsymbol{v}_N\right|\right|_1 \\ \text{s. t.} \quad \begin{bmatrix} F_1 \\ A_\mathrm{ieq} \end{bmatrix} \boldsymbol{v}_N &\leq \begin{bmatrix} \boldsymbol{g}_6 \\ \boldsymbol{b}_\mathrm{ieq} \end{bmatrix} - \begin{bmatrix} F_2 \\ F_\mathrm{ieq} \end{bmatrix} \boldsymbol{p}_N \end{aligned} \tag{3-37}$$

The 1-norm of the cost function can be transformed into a new optimization problem with additional constraints. The resulting problem is Mixed Integer Linear Programming (MILP), meaning, that both the cost function and the constraints are linear. This way, the computational complexity of the algorithm is reduced.

3-2-3 Reducing complexity with heuristics

The complexity of the optimization problem can further be reduced by actively using heuristics during the assembly of the optimization problem. The two biggest drawbacks of an MLD formulation are the high number of binary variables and the amount of constraints. The number of constraints can hardly be reduced because the MLD formulation is based on connecting auxiliary real and binary variables with inequality constraints. The number of binary variables per timestep is also fixed, and so it can only be reduced by making the control horizon shorter. On the other hand, the binary variables can be fixed to certain values each timestep using equality constraints, which then the solver can substitute in and presolve, making the problem less complex. Figure 2-6 is a general shape of WD curve which can be seen in most greenhouses: Dutch growers tend to prefer maximal water content during the day to ensure, that all the gutters inside a compartment have sufficient water supply, while during the night they aim for a certain drydown to let the root zone of the plants get enough oxygen.

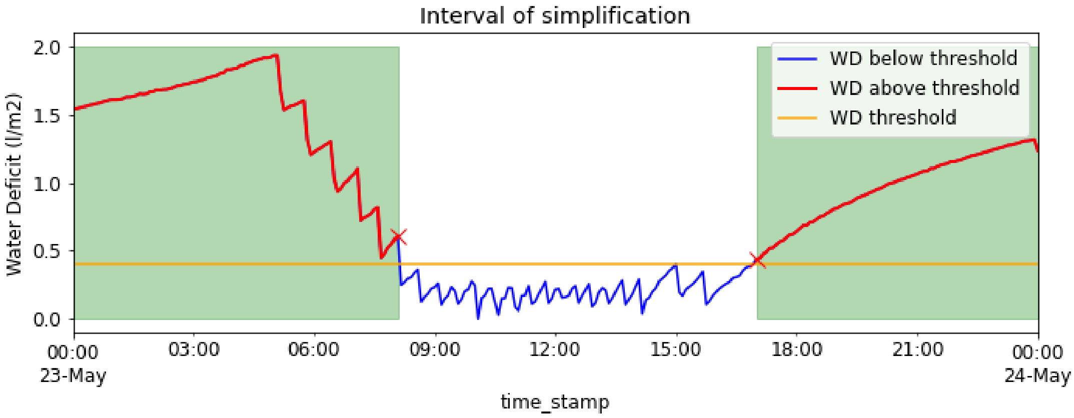

Figure 3-1: Interval of simplification

The cost function in (3-37) is based on the deviation from the reference trajectory $\boldsymbol{y}_{\mathrm{ref},N}$, and if the reference trajectory has a similar shape as Figure 2-6, it is likely that the provided solution is going to follow that trajectory accurately, given that there are no restricting constraints, such as maximal irrigation amount or maximal WD. In Figure 3-1 the two ends of the interval are marked, where the WD is below a threshold of 0.4 l/m^2. Outside this interval, it is possible to use only the simple "drainless" model, which means that $\delta = 0$ in the model of Section 2-4-1, and that $\delta_1 = 0$, $\delta_2 = 1$, and consequently $\delta_3 = 0$ in the model of Section 2-4-3. The interval where the simplified model can be used is denoted by the green opaque area. The constraints are formulated in the following way:

$$
\begin{aligned}
A_{\mathrm{H}}\boldsymbol{v}_N &= \mathbf{1} \otimes 0 \\
\underbrace{\begin{bmatrix} 0 & A_{\mathrm{H},\delta} & 0 \end{bmatrix}}_{A_{\mathrm{H}}} \begin{bmatrix} \boldsymbol{u}_N \\ \boldsymbol{\delta}_N \\ \boldsymbol{z}_N \end{bmatrix} &= \mathbf{1} \otimes 0
\end{aligned} \tag{3-38}
$$

or:

$$
\begin{aligned}
A_{\text{H}}\boldsymbol{v}_N &= \mathbf{1} \otimes \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \\
\underbrace{\begin{bmatrix} 0 & A_{\text{H},\delta} & 0 \end{bmatrix}}_{A_{\text{H}}} \begin{bmatrix} \boldsymbol{u}_N \\ \boldsymbol{\delta}_N \\ \boldsymbol{z}_N \end{bmatrix} &= \mathbf{1} \otimes \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}
\end{aligned} \tag{3-39}
$$

Where $A_{\text{H},\delta}$ selects only the δ values which are outside the described interval. The matrix depends on the time of calculation, the length of the control horizon as well as the length and relative position of the described interval. To avoid faulty calculations, the assumption of the WD being above the threshold can be enforced using inequality constraints on the output $\boldsymbol{y}_N$:

$$
\begin{aligned}
\tilde{A}_{\text{H},y}\boldsymbol{y}_N &\leq -\mathbf{1} \otimes \bar{y}_{\text{WD}} \\
\tilde{A}_{\text{H},y}\left(P\boldsymbol{p}_N + S\boldsymbol{v}_N\right) &\leq -\mathbf{1} \otimes \bar{y}_{\text{WD}} \\
\underbrace{\tilde{A}_{\text{H},y}S}_{A_{\text{H},y}}\,\boldsymbol{v}_N &\leq -\mathbf{1} \otimes \bar{y}_{\text{WD}} - \underbrace{\tilde{A}_{\text{H},y}P}_{F_{\text{H},y}}\,\boldsymbol{p}_N \\
A_{\text{H},y}\boldsymbol{v}_N &\leq -\mathbf{1} \otimes \bar{y}_{\text{WD}} - F_{\text{H},y}\boldsymbol{p}_N
\end{aligned} \tag{3-40}
$$

Where $\tilde{A}_{\text{H},y}$ (similarly to $A_{\text{H},\delta}$) selects the WD values outside the interval. To counter possible degradation of solutions and to avoid conflicts with the constraints, the interval can be extended by 1 hour on both ends, providing flexibility to the optimisation algorithm. Figure 3-2 shows with opaque green area the possible values WD can take during the simplified optimisation.

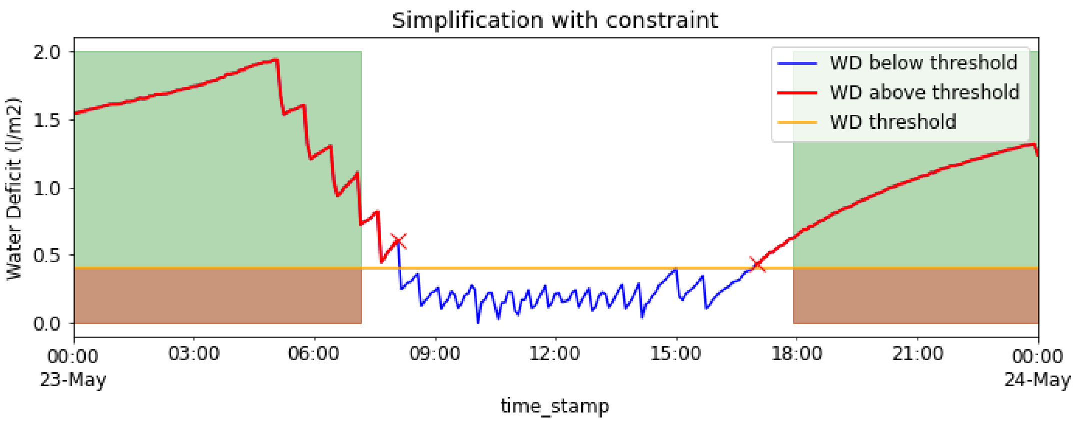

Figure 3-2: Interval of simplification with constraints

3-3 Summary

The chapter used the formulated models from Chapter 2 to create the optimal control problems needed for the use of Model Predictive Control (MPC). Section 3-1 detailed, how a regular linear MPC was created for the irrigation problem, with cost and constraint definitions, as well as move-blocking. Section 3-2 showed, how the optimal control problem can be reformulated to accommodate the use of MLD models. A linear cost function was presented to reduce the complexity of the mixed integer problem, creating a MILP problem, which can be efficiently solved with modern optimization software. Further simplification was done using heuristics which reduced the number of auxiliary binary variables.

Chapter 4

Validation

The previous two chapters introduced both the modeling and the control design of the irrigation system. Although the accuracy of the individual models (Plant Water Uptake (PWU) and drain) was discussed in Chapter 2, the combined performance of the water balance model is still a question, which is answered in Section 4-1. In Section 4-2 the controller is assessed in terms of feasibility, run-time, and with comparing the created irrigation decisions to the grower's strategies.

4-1 Model validation

Chapter 2 introduced 4 water balance models, of which 2 describe the same system but in different ways. The first model was introduced in Section 2-4-1 in (2-36), which is a linear State-Space (SS) model. The second model from (2-45) is a Mixed Logical Dynamical (MLD) model, which is only identified below a certain Water Deficit (WD) threshold, and above that the drain is set to 0. The third and fourth models describe the same system: the second model extended with the condition, that the WD can not sink below 0. The model validation handles the Piecewise Affine (PWA) representation of the system from (2-49).

For the evaluation of the models a number of tests are conducted. First, the models are trained on a part of the dataset and are then evaluated on the other part, without retraining. Then the models are trained and evaluated in a progressive manner (as discussed in Figure 2-5), which simulates how the controller could be used in real-life. To test the accuracy of the models, the same historic irrigation inputs are used as in the data, and the WD, PWU and drain predictions are compared with the measurements. Both approaches are evaluated on different greenhouse datasets. These can be found in the appendix B. In general, the following metrics are used by the evaluation:

- R^2 of the predictions
- Normalized Root Mean Square Error (RMSE) of the predictions

- Accuracy of the daily accumulates
- Moving average of RMSE

4-1-1 Static identification

The first dataset for the evaluation is the one which was used during the project to create the models and the controller. It is detailed at the beginning of Chapter A. Figure 4-1 shows a 48 hour prediction of WD on the validation data. From the 3 models, the PWA model is the most accurate, followed by the first type of hybrid model. The linear model is the least accurate. The bottom subplot shows the moving average of the normalized RMSE with a sliding window of 20 samples. It can be seen that with the PWA model, the moving average of the RMSE stays below 20% for the whole 48 hours.

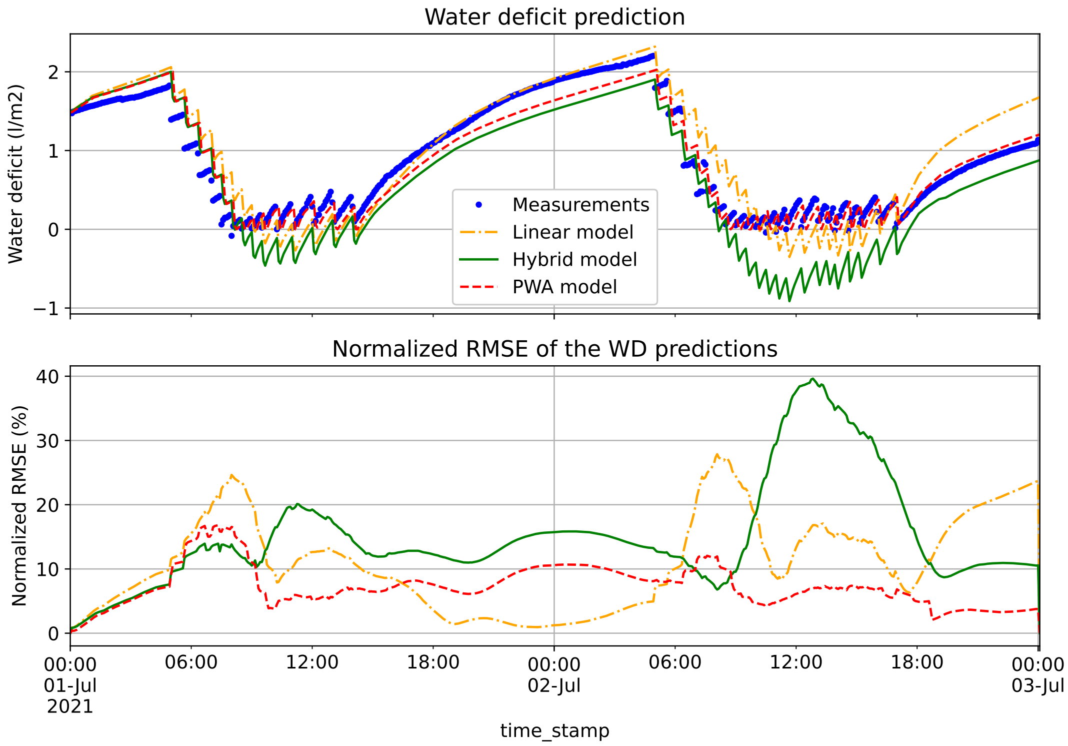

Figure 4-1: Example prediction on the first dataset - WD

Figure 4-2 shows the measured and predicted cumulative values of drain and PWU. There is only one PWU prediction because it is a static regressional model, which provides the disturbance input $w(k)$ to the different dynamical models. It can be seen, that although the PWA model predicts the WD the most accurately, the drain is best predicted by the hybrid model. It can also be seen, that the PWU model underestimates the actual PWU. To understand, how the PWA model predicts the most accurately, it is enough to look at its logical structure: if the water deficit would go below 0, the excess water from the irrigation leaves through drain. Because the PWU is underestimated, the irrigation would push the

WD to a negative regime, which in turn results in more drain. The mismatch in PWU is added to the drain and so the water output is balanced out, and the WD prediction stays accurate. This behaviour can keep the system in line if the PWU is underestimated, but not if it is overestimated.

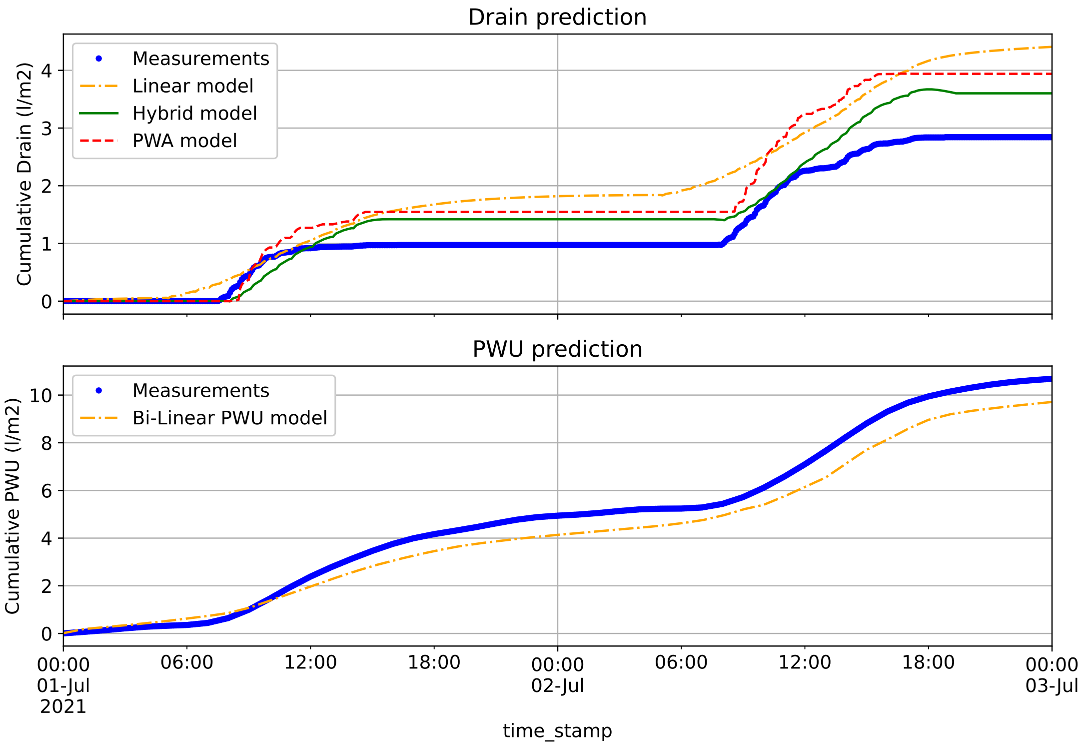

Figure 4-2: Example prediction on the first dataset - PWU and drain

The models were trained on 70% of the data. The validation was done on the remaining 30%. The 2 days accuracy: RMSE, R^2 and accuracy of the cumulative values in the end can be seen in Figure 4-3. The R^2 of the predictions stays high with the PWA model, but a slight drop can be noticed after around 15 days from the end of the identification set (beginning of the validation set). The RMSE of the predictions is also low with the PWA model. The end-of-prediction cumulative values of the drain differ the least with the hybrid model. It is interesting to notice, that the cumulative errors of the PWU are almost like the drain errors of the PWA model mirrored on the x axis, and that the PWU model is constantly underestimating the actual water uptake of the plants. This supports the argument from the previous paragraph.

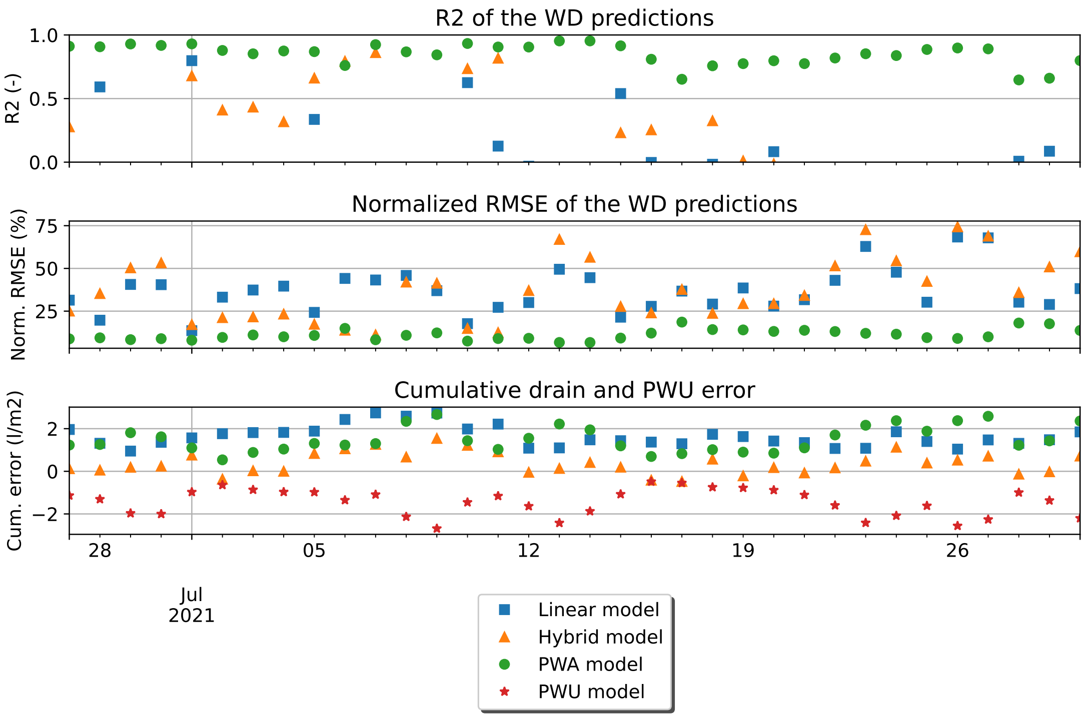

Figure 4-3: Accuracy with the static identification approach

The per sample errors can also be inspected. This can give a general idea on how the errors are distributed, when are the models the least and the most accurate. Figure 4-4 shows the mean of the WD errors with a band of $\pm\sigma$ (standard deviation) around it. As shown in Figure 4-3, the PWA model has both the lowest mean of error and the lowest standard deviation of errors. It is interesting, that compared to the linear and hybrid models, the PWA model does not suffer from a growing σ band nor a drift of mean error. This is explained by the extra nonlinearity (maximal drain outflow if the WD would go to zero) that was added to the model. The PWA model provides unbiased predictions, except for the early hours in the morning, when the first irrigation shots happen. During these early hours the model overestimates the WD. The other two models provide unreliable predictions with a big standard deviation, and with an additional drift by the hybrid model.

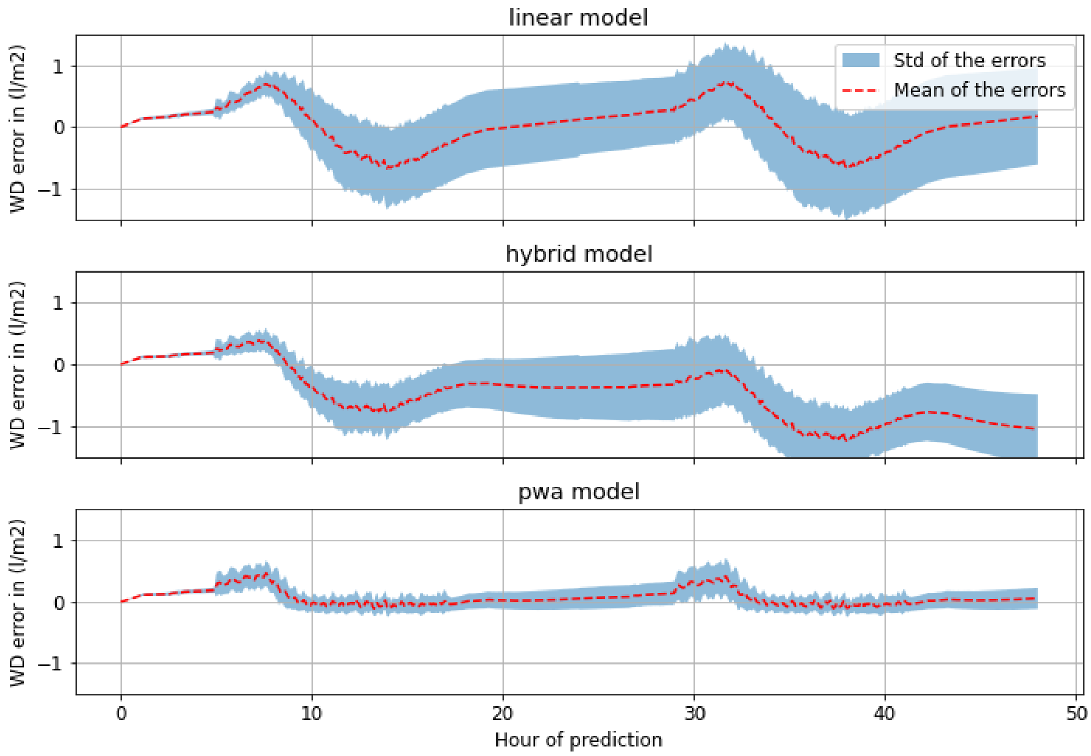

Figure 4-4: Error distribution of the predictions with the static identification approach

From the presented results it is shown, that from the three models the PWA model is by far the most accurate. Its introduced nonlinearity to output the excess irrigation water via drain makes it robust against drift in the WD predictions.

4-1-2 Progressive identification

In Section 2-3-2 it was mentioned, that the dynamical model of the drain prediction was identified through the use of a progressive (sliding window) approach, which is illustrated in Figure 2-5. To test the approach, the first 10 days were used to identify the models. Predictions are calculated using the identified models, and the data of the next 2 days are used to validate the predictions. After the validation, the identification data is extended with an additional day and the earliest day is removed, so the number of days in the training dataset stays the same. This is referred to as sliding window identification. For the validation the same days were used as with the static identification, but with the progressive approach. This means, that the identification of the models started 10 days before the validation set (this means a considerably lower data quantity than with the static identification). On the other hand the algorithm had the advantage, that it could constantly reidentify the models using the days before the actual validation data of 2 days.

Figure 4-5 shows the accuracy of the progressive identification approach. As it can be seen, the accuracy of all of the algorithms is considerably lower than with the static identification. There are days where even the PWA model's R^2 drops below 0.

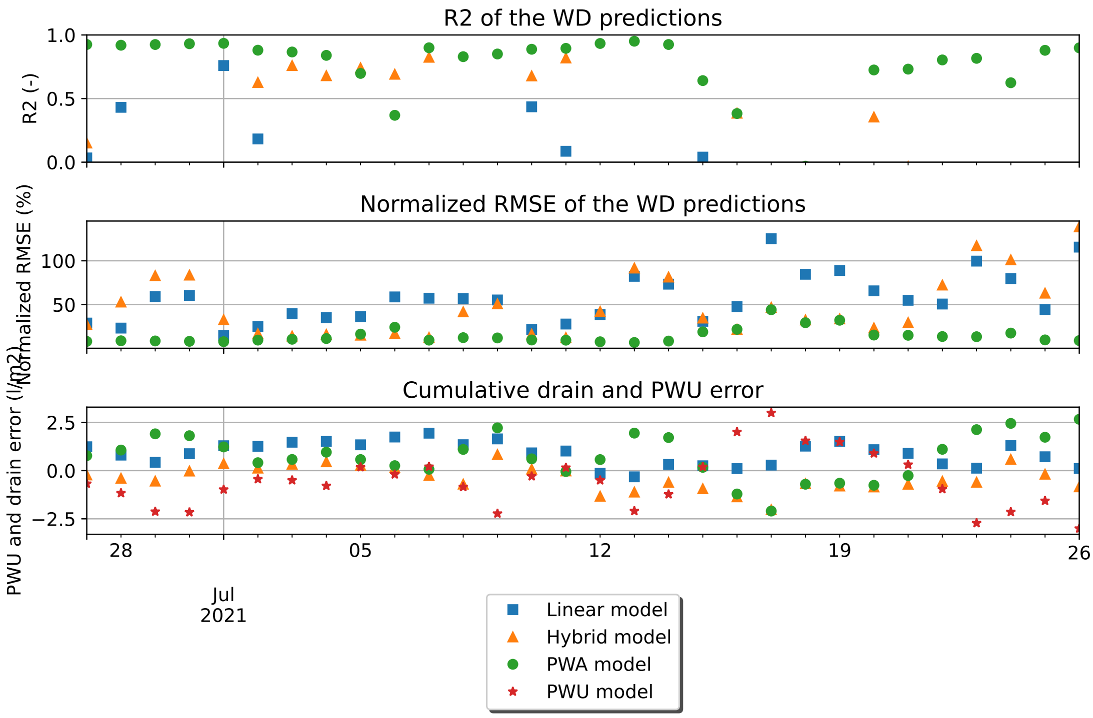

Figure 4-5: Accuracy with the progressive identification approach

Figure 4-6 shows, how the WD prediction errors are distributed throughout the prediction horizon. The progressive identification shows more bias and bigger standard deviation than the static identification. The reason behind these results could be the fact, that the progressive identification is susceptible to local model mismatches.

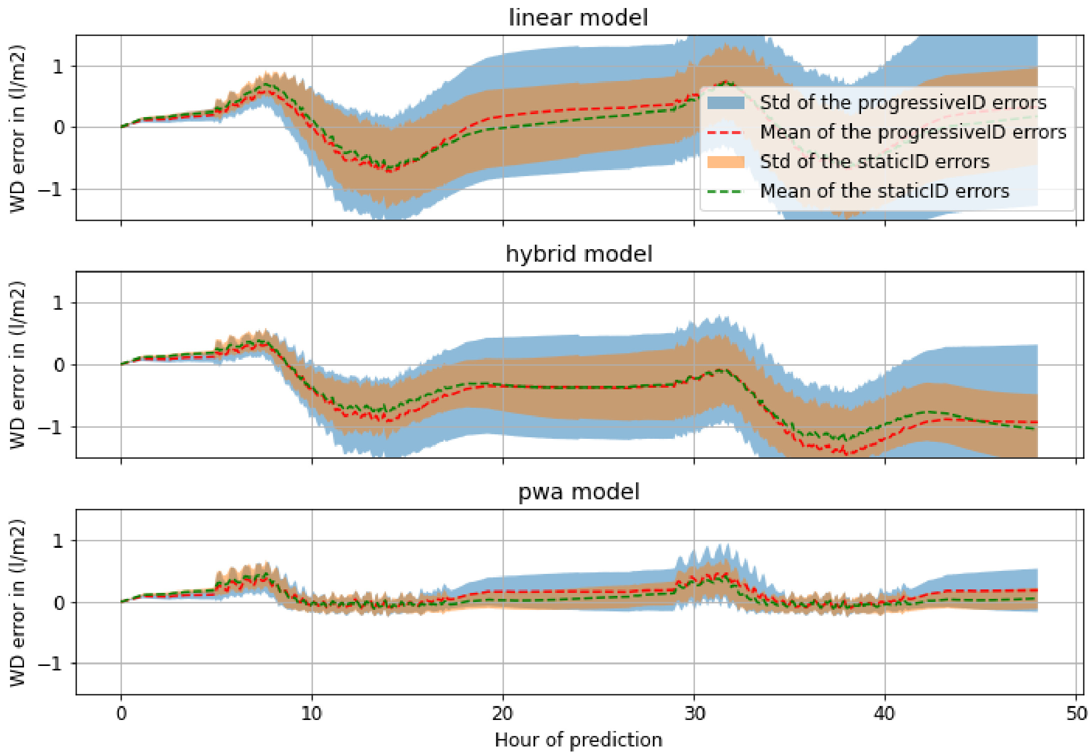

Figure 4-6: Combined error distribution of the predictions

The appendix contains the same validation figures for other greenhouse datasets. The generalizability of the approach seems to be high: compared to the original set there were slightly better and also slightly worse performing models. It is interesting to note, that although the static identification proved better with the original dataset, the validation on the "Greenhouse 2" dataset showed, that the progressive identification provided better overall accuracy with a lower amount of drift.

4-1-3 Prediction using forecast parameters

The previous sections showed how the models perform under ideal circumstances, when both the radiation and the humidity deficit are known beforehand. In real-life application, only estimates (forecasts) are available on these parameters. The radiation predictions are usually supplied by local meteo stations, to which the effect of scheduled lighting strategy can be added. Humidity Deficit (HD) on the other hand is an indoor climate parameter, which is constantly influenced by the processes inside the greenhouse. Certain climate control algorithms have the ability to predict, how the humidity deficit inside the greenhouse is going change, and therefore forecasts on HD can be obtained. During the thesis project I had access to the data and results of one of these algorithms.

In this subsection the accuracy of the forecasts is inspected using a dataset from a Dutch tomato growing greenhouse between 01-04-2022 and 22-06-2022, which is just above 80 days. First, the accuracy of the forecasts is inspected, then PWU predictions made using the forecast variables are compared to PWU predictions made with the actual variables. The performance of predicting the WD trajectory is also compared.

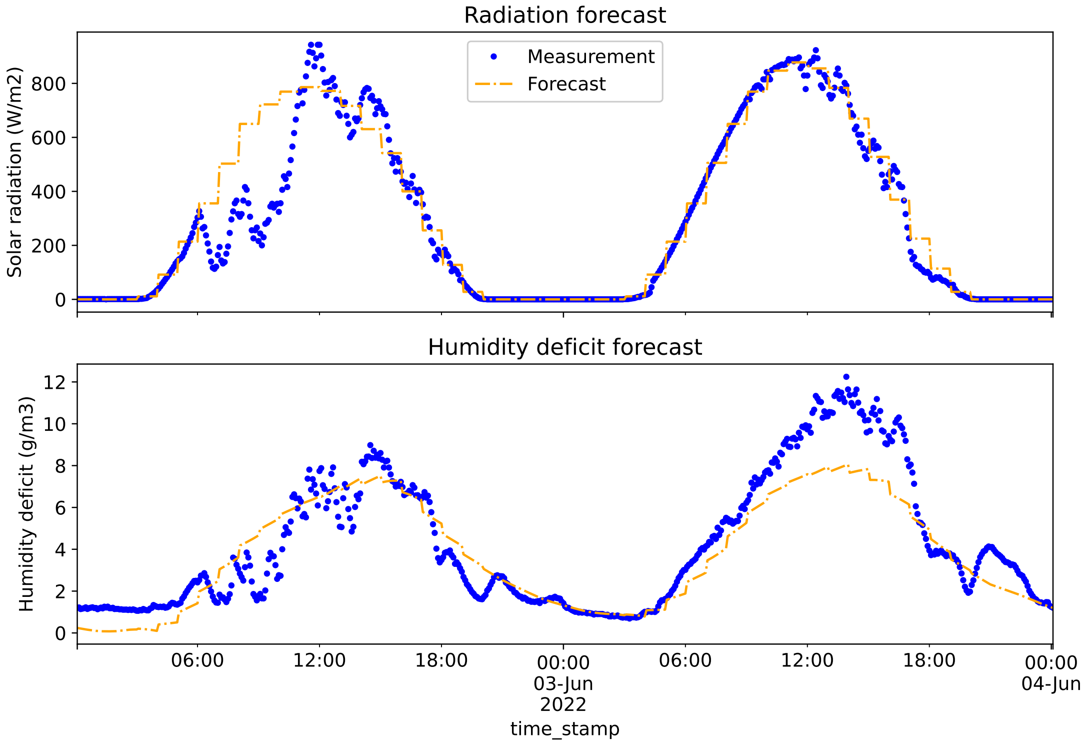

Figure 4-7: 2 day forecast of radiation and indoor HD

Figure 4-7 shows a forecast of 2 days where the first day is inaccurate and the second day is accurate in terms of radiation. The inaccuracy during the first day is possibly caused by cloud movements. The forecasts used by the greenhouse did not take into account the cloud movements, and so independent on how far in the future the forecasts are, the radiation predictions have the same shape as can be seen in the figure.

Figure 4-8 shows the average and standard deviation of errors every 5 minute of a 48 hour forecast taken at midnight, for every day in the validation dataset. It can be seen, that both of the climatic parameters are forecast with a maximum of around 30% standard deviation. With radiation forecasts, there is a small positive bias around noon, which means, that the forecasts overestimate the radiation. This can happen because the forecasts do not take into account cloud movements, which reduce the amount of actual solar radiation landing on the greenhouse. The HD forecasts do not have a structured bias.

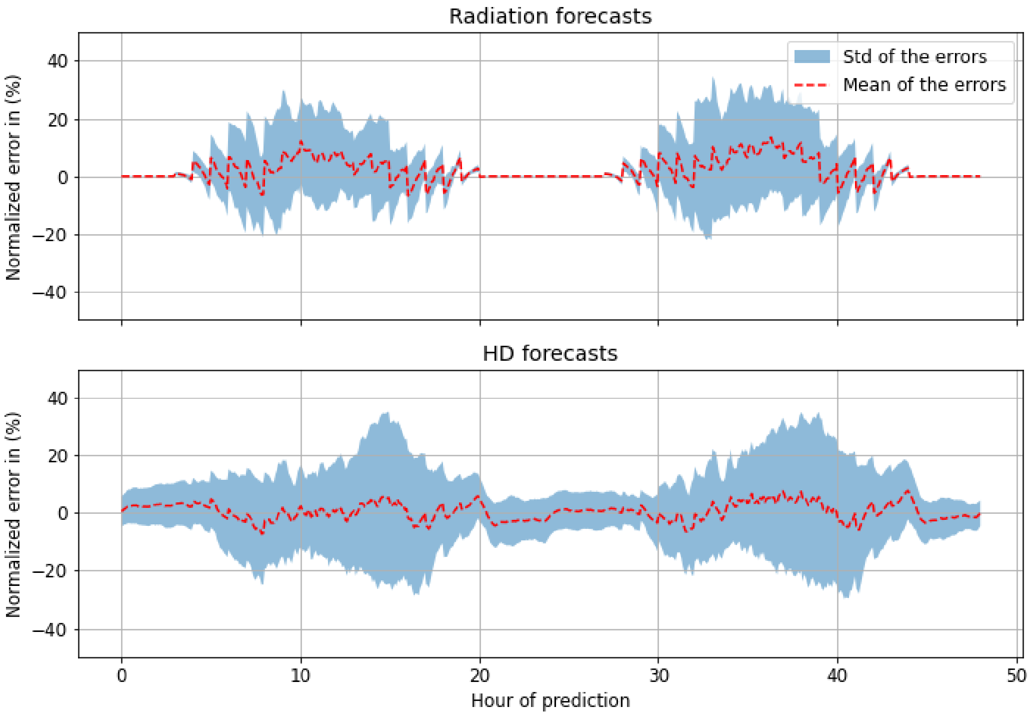

Figure 4-8: Mean and standard deviation of the error of climate forecasts

Figure 4-9 shows predictions made on a day with low forecast errors. It can be seen, that the models produce quite similar results. Both predictions precede the actual PWU, but the daily cumulative amounts are close to eachother.

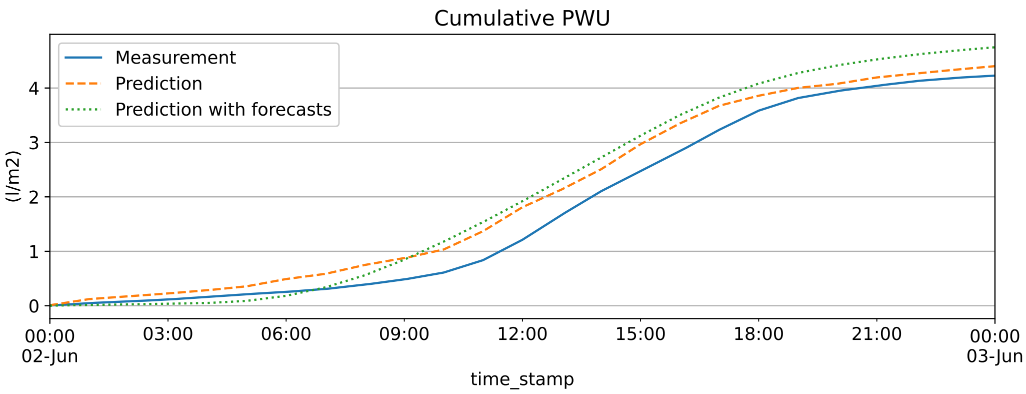

Figure 4-9: PWU predictions on an accurate forecast day

Figure 4-10 shows the error plot of the models with and without forecast climate data. Making predictions at midnight for 48 hours ahead shows, that the PWU model has a clear positive bias. Comparing the two models reveals, that both the mean and the standard deviation of errors is on par with each other, implying that the use of forecast climate data did not hinder the accuracy of the predictions, at least on this dataset. Figure 4-11 shows, that the errors during a 48 hour WD prediction do not change significantly with forecast data.

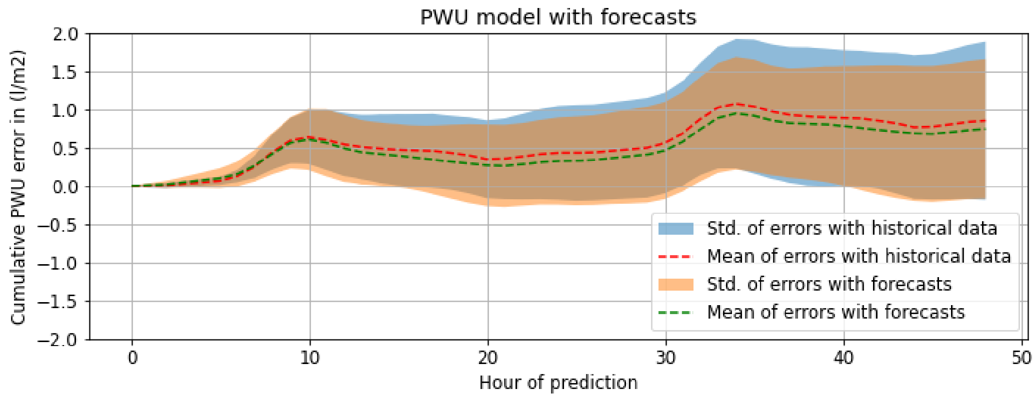

Figure 4-10: Mean and standard deviation of the error of PWU predictions

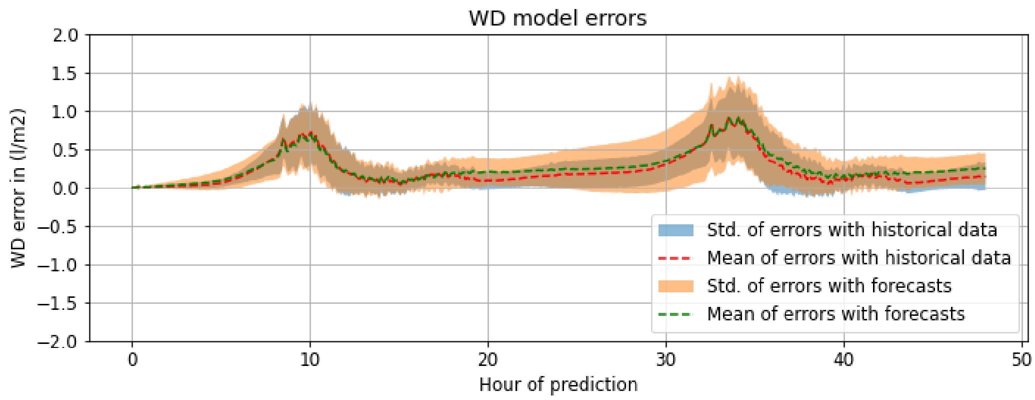

Figure 4-11: Mean and standard deviation of the error of WD predictions

4-2 Controller validation

Because during the project there was no possibility to test the controller in a real greenhouse, and because accurate greenhouse irrigation simulation models were not available, the assessment of the controllers was done in a different way. The controllers were tested in an open-loop on the available data. The following measures were considered:

- Input recreation accuracy on historical data
- Computation time of the optimal control problem

For the tests in this section the GUROBI[30] optimizer was used with an academic licence. The optimization algorithm was run on a laptop PC with 8GB of RAM and an Intel Core i5-7300HQ CPU with 4 physical cores.

4-2-1 Input recreation accuracy

Because closed-loop validation was not possible during the project, the inputs the controllers calculate are compared to the original historical inputs. For these experiments, only non-negativity constraints were imposed on the irrigation shots and the WD output, and the reference WD trajectory is set to the historical WD values which were realized during the day of the measurement. Cost (q) on the deviation of the WD from the predefined trajectory was set to 1, as well as the cost on irrigation (r). For the PWU estimation, the historical indoor HD and solar radiation data were used. The experiment was done on a day in the validation dataset where the model fits were good. The control horizon was 8 hours, from 5am until 1pm. The models were identified using the static identification approach presented in Section 4-1-1

Figure 4-12 show the WD output which the Model Predictive Control (MPC) algorithms calculated. It can be seen, that the trajectories match the reference perfectly. This comes from the fact, that each controller found an optimal input sequence regarding its own model. As can be seen in Figure 4-13, although each controller realized the predefined trajectory, the inputs are substantially different.

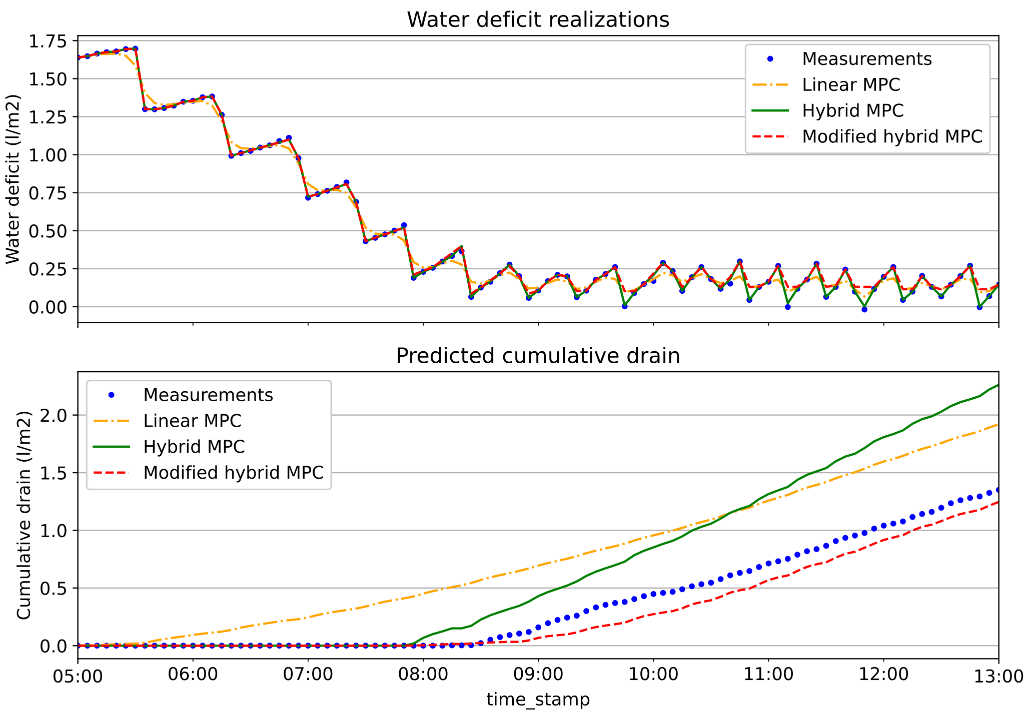

Figure 4-12: Realized WD trajectories with the MPC algorithms

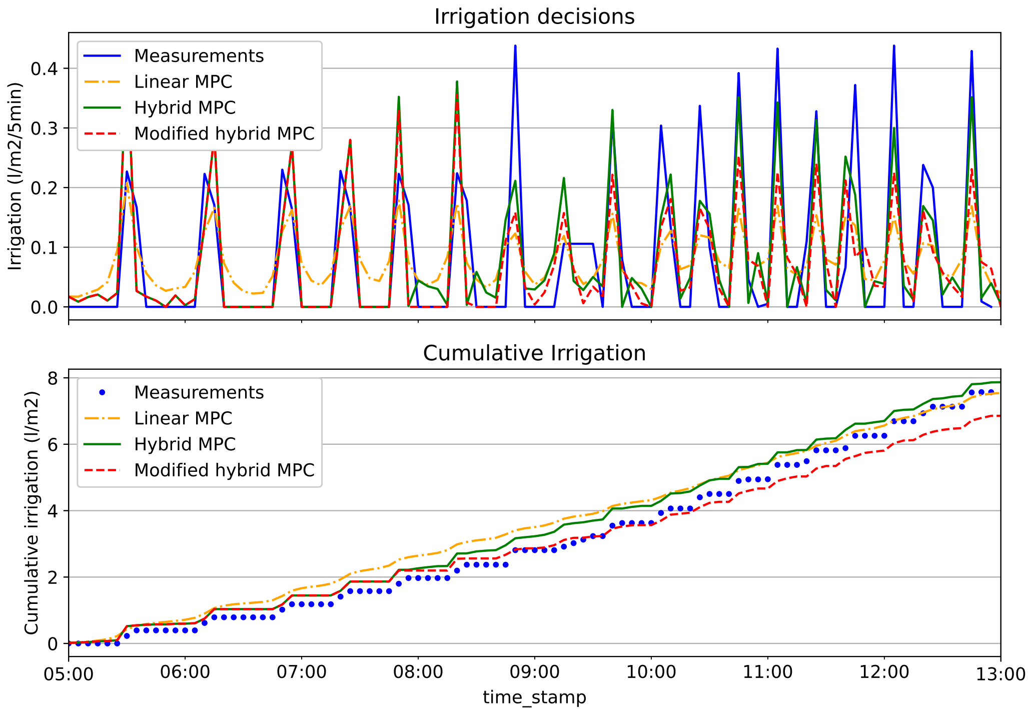

Figure 4-13: Irrigation inputs created by the MPC algorithms

The calculated inputs do not match the historical irrigation decisions completely, but the cumulative values are really close. Table 4-1 shows the statistical measures of the fit between the historical and calculated values.

Input comparison	Linear MPC	Hybrid MPC	Modified hybrid MPC
Normalized RMSE of the irrigation shots	22%	17%	20%
R^2 of the irrigation shots	0.43	0.63	0.52
Normalized RMSE of the cumulative irrigation	6.5%	4.7%	4.5%
R^2 of the cumulative irrigation	0.95	0.98	0.98
Normalized end-of-day difference of cumulative values	-2%	2%	-11%

Table 4-1: Accuracy of recreated irrigation schedules

From the algorithms, the hybrid MPC performed the best, achieving the highest fits and, except for the RMSE of cumulative irrigation, the lowest errors. It is interesting to note, that all the three algorithms recreated the irrigation decisions quite accurately, independent on the performance of their respective models. From the algorithms only the modified hybrid MPC was able to accurately predict the drain trajectory.

The input recreation was also inspected using forecast radiation and HD variables (similarly as in Section 4-1-3). As mentioned in Section 4-1-3, the dataset used for forecast validation is different than what was used in the previous tests in this section, so it is only meaningful to compare the results of the forecast variable based prediction to the historical variable based one. Figure 4-14 shows the recreated inputs for a day when the forecasts were accurate, while Figure 4-15 shows the results for a day with less accurate forecasts. Considering that the irrigation decisions are made in the morning for the whole day, both figures depict an acceptable irrigation schedule. In Figure 4-14 only a minor difference can be seen between the irrigation decisions made with and without forecasts. Figure 4-15 shows bigger difference between the irrigation decisions, and also between the predictions and the historical irrigation decisions, and the error of the end-of-the-day cumulative irrigation is around 10% for the regular predictions and around 16% for the predictions made with forecasts.

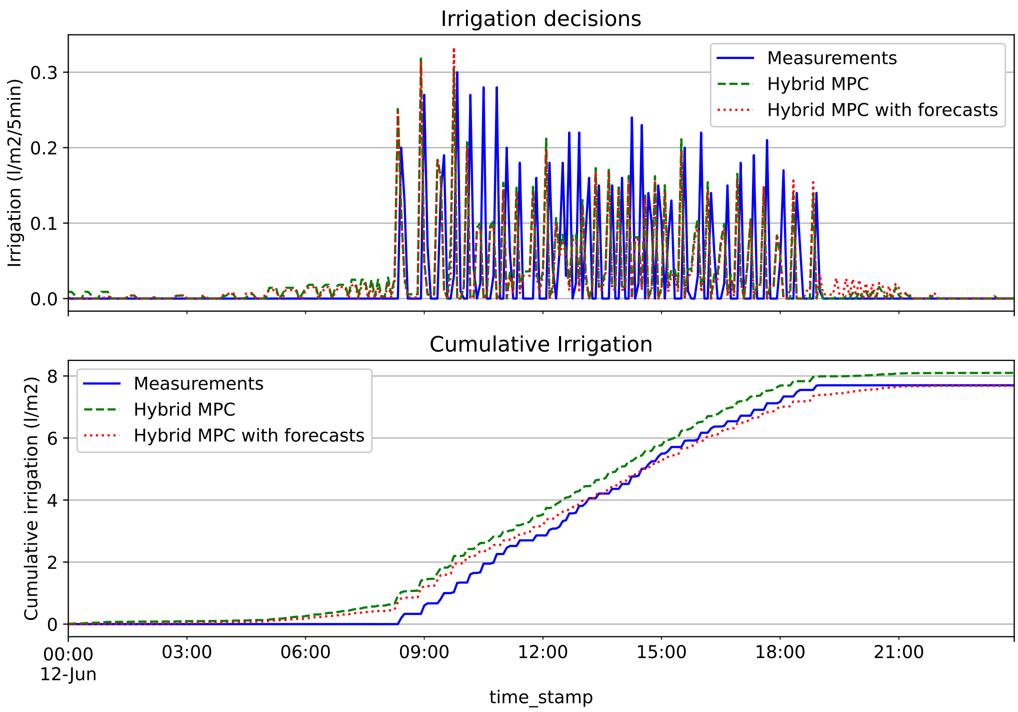

Figure 4-14: Irrigation inputs created by the hybrid MPC algorithm for a day with accurate forecasts

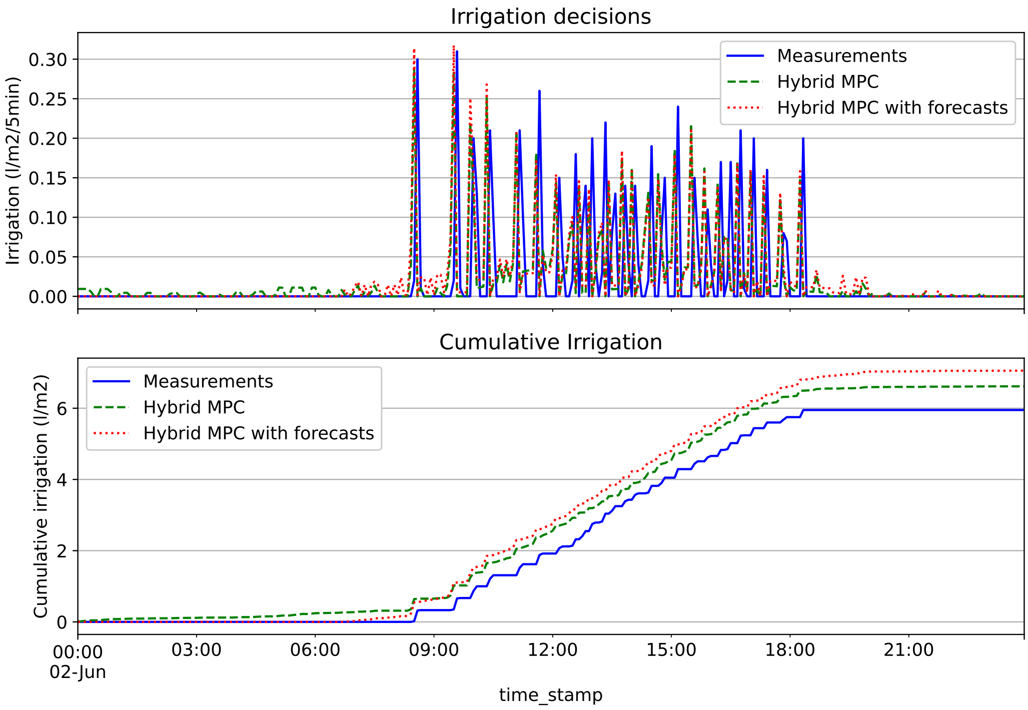

Figure 4-15: Irrigation inputs created by the hybrid MPC algorithm for a day with inaccurate forecasts

4-2-2 Computation time

One of the biggest drawbacks of Mixed Integer Linear Programming (MILP) and Mixed Integer Quadratic Programming (MIQP) is the time it takes to compute solutions. Table 4-2 shows how the computation time of the algorithms change as the control horizon of the system grows. For these experiments, only non-negativity constraints of the irrigation and WD were imposed on the optimization problem. The same costs were used as in the previous section. The starting time of the MPC was 6 am, which means, that with every horizon length the interval under the WD threshold was included.

Computation time in (s)	Ht = 2 hour	Ht = 5 hour	Ht = 12 hour	Ht = 24 hour
Linear MPC with QP	0.024	0.033	0.084	0.256
Hybrid MPC with MIQP	0.264	-	-	-
Modified hybrid MPC with MIQP	0.330	314.8	>1000	>1000
Hybrid MPC with MIQP and heuristics	0.117	-	-	-
Modified hybrid MPC with MIQP and heuristics	0.281	209.8	>1000	>1000
Hybrid MPC with MILP	0.141	1.97	190.3	>1000
Modified hybrid MPC with MILP	0.326	30.41	>1000	>1000
Hybrid MPC with MILP and heuristics	0.079	1.66	169.3	376.1
Modified hybrid MPC with MILP and heuristics	0.128	12.51	>1000	>1000

Table 4-2: Computation time of optimal solutions

The results show, that the most accurate modified hybrid model suffers the most from high computation time. The linear MPC provides optimal results the quickest. A quadratic cost function put an extra load on the mixed integer optimization algorithm, but its early convergence is higher than the MILP problems. For optimal solutions, the linear cost function is recommended, but for suboptimal solutions the use of the quadratic cost function could yield acceptable results faster. The calculation of the MIQP failed during the validation with the hybrid MPC algorithm for a control horizon equal to or longer than 5 hours because of convexity issues. Using a linear cost function, heuristics for simplification, an optimal irrigation schedule can be created for 24 hours ahead in less than 7 minutes.

4-3 Summary

The validation results show, that from the developed models, the modified hybrid MLD or PWA models (which are the same) could be used effectively for prediction purposes. Although 48 hour predictions could sometimes be accurate, the validation on other greenhouse data showed, that 24 hour predictions are still accurate, 48 hour ones not that much. The shorter horizon length is also favorable for the hybrid MPC controllers, because of the computational time.

The controller validation showed big similarities between historical and calculated inputs, which is promising for the real world application of the algorithm. Using forecast parameters does not hinder model accuracy substantially, and the created irrigation decisions are close to the ones made using historical climate variables. Because of the high computational times and the availability of weather forecasts (which is hourly), the controller could be used most effectively in a receding horizon, where every 1 hour new irrigation decisions are created. Although the modified hybrid model was the most accurate in terms of prediction, the regular hybrid model provided very accurate reproductions of the historical irrigation decisions. Paired with the heuristics the run-time of this algorithm can be significantly reduced, therefore even 24 hour long irrigation schedules can be created with it.

Chapter 5

Conclusion

In the previous chapters, the process of modeling and identification, control design, and the evaluation of the models and control algorithms was presented. The following few paragraphs summarize the findings of this research.

The modelling of the Plant Water Uptake (PWU) relied on a simple bi-linear regression model structure from the literature. On the other hand, different unique approaches were created to model the dynamics of drain in the system. From the developed models, the Piecewise Affine (PWA) model with 3 modes proved to be the most accurate, which was also able to counteract the occasional inaccuracy of PWU predictions. The validation results showed, that the model was able to predict quite accurately for at least 24 hours ahead the evolution of the Water Deficit (WD) curve, given climatic parameters and fixed irrigation decisions. The use of forecast radiation and Humidity Deficit (HD) data did not hinder the accuracy of the models substantially, which is a promising result towards the real-world application of the control algorithm. It should be noted, that both the training and validation data had a narrow band of variations, which comes from the fact, that professional growers prefer having consistent irrigation decisions throughout the growing season. Although the water balance equation keeps the individual models in check, the low variation of the data may cause the developed models to perform poorly in situations, where the WD trajectory differs considerably from the training dataset.

The inspected control algorithms offer great flexibility in their application. Through the adjustment of costs and constraints, the behaviour of the Model Predictive Control (MPC) can be changed substantially, which gives the developed controller the opportunity to satisfy different goals, or make a trade-off between multiple targets. The models define the basis of what kind of optimization problems can be written up. The validation results showed, that although the simple linear model resulted in a regular quadratic program, which is the fastest to solve, the linear model fails to describe the water balance of the system accurately. The use of the modified Mixed Logical Dynamical (MLD) model lead to either a Mixed Integer Quadratic Programming (MIQP) problem or a Mixed Integer Linear Programming (MILP) problem, which were both by orders of magnitude slower to solve, especially for long horizons. Move-blocking was effectively used to reduce the complexity of the regular linear MPC problem, but it failed to reduce the complexity of the MLD MPC approaches. To successfully reduce the complexity of the MLD optimal control problems, heuristics were applied to simplify the models during high WD hours. Although the modified hybrid model was the most accurate in terms of prediction, the regular hybrid model provided very accurate reproductions of the historical irrigation decisions. Paired with the heuristics the run-time of this algorithm can be significantly reduced, therefore even 24 hour long irrigation schedules can be created with it.

5-1 Recommendation for future research

The developed controller showed, that MPC could be viable in creating an autonomous irrigation controller, the behaviour of which is easily adjusted to the preference of greenhouse growers. Although the thesis project showed how the created framework could be used, numerous options for development are present which could further enhance the performance of the algorithm, or decrease its computational complexity. In the following paragraphs, recommendations are given on future research possibilities, as well as on how to apply the control algorithms on real greenhouses.

The main weakness of every model based control approach is the model itself. To achieve better performance, measurements of Vapor Pressure Deficit (VPD) could be used instead of HD, if available. This minor change could contribute to better accuracy for long term predictions. Advanced PWU models, such as [27], [31] and [24] could also be used to substitute the bi-linear approach, given that there are additional measurements available on e.g. convection, Leaf Area Index (LAI) etc. The use of more complicated and nonlinear models would not hinder the computational performance of the system, because the PWU predictions are independent from the water balance as long as the plants are not under drought-stress.

As validation showed, the choking point of MLD MPC is the number of constraints and binary variables the optimization algorithm has to handle. Koopman operators can depict nonlinear systems such as the PWA water balance model into a high order linear system. The created linear system could be used for linear MPC, so that only a regular Quadratic Programming (QP) problem has to be solved, which would greatly improve the computational speed and prediction horizon of the system. Additionally, move-blocking could be used to further improve the computational speed. [32] summarizes the application of Koopman operators in a really concise manner, and [33] already showed, how Koopman operator theory can be used for MPC.

Nonlinear MPC could also be used to solve the optimal control problem with the PWA model. Although there are clear drawbacks of non-convex nonlinear optimization, with good initial guesses on irrigation, and with effective solvers, nonlinear MPC could be viable for suboptimal MPC, which could handle the trade-off between computation time and control performance.

The developed algorithms provide an output of irrigation amounts in l/m^2 for every 5 minutes in the control horizon. This can be translated to an irrigation schedule, which could then be automatically supplied to climate computers. During every hour, a new irrigation schedule is made using actual measurement data, and this is then uploaded in the climate computer, overwriting the previous set of decisions. In case of network problems, or computational issues, the previously uploaded irrigation schedules would define the irrigation strategy, or a fallback could be initiated, during which the decision making would return to the standard radiation influence based irrigation.

During an interview with a professional Canadian tomato grower, the possibility of including Electrical Conductivity (EC) of the drain or substrate inside the models was brought forward, as well as the inclusion of daily radiation-irrigation ratio as a controllable parameter. A big disadvantage of the developed algorithm was also discussed: small irrigation shots are not realistic to use because they result in uneven watering through the length of the gutter and the pressurized drippers. Using an extra auxiliary binary variable, a minimal irrigation shot amount can be set up, which would correct this behaviour.

Appendix A

Data preparation

The data used in the project comes from a greenhouse experiment in the Netherlands, where long truss tomatoes were grown. The greenhouse was equipped with a meteorologic station, a climate measurement box and additionally an irrigation actuation and measurement group. In the greenhouse, two slab weighing scales were present, both of which collected data during the whole experiment. The experiment lasted a whole growing season, but the data was heavily corrupted at many places, so only a part of it was used in this project: from 11/04/2021 until 01/08/2021. The data has a sampling time of h_1=5 minutes, and $n = 32257$ available samples. Table A-1 shows the available and for the project relevant measurements from the data. The column '% of N.a.' gives the percentage of "not-a-number" (faulty) measurements in the data.

Table A-1: List of available parameters from the greenhouse

Parameter name	Unit	% of N.a.	Details
Radiation sum	J/cm^2	2.3	Sum of outside radiation and indoor lights
Outdoor absolute humidity	g/cm^3	6.1	From weather station
Global radiation	W/m^2	8	From weather station
Total radiation	W/m^2	16.7	From weather station
Indoor absolute humidity	g/cm^3	24.6	From climate computer
Indoor temperature	$°C$	4.4	From climate computer
Indoor relative humidity	%	4.4	From climate computer
Indoor humidity deficit	g/cm^3	4.4	From climate computer
Drain percentage	%	4.3	From irrigation group
Cumulative drain	l/m^2	3.6	From irrigation group

Table A-1: List of available parameters from the greenhouse

Parameter name	Unit	% of N.a.	Details
Cumulative irrigation	l/m^2	3.4	From irrigation group
Cumulative absorption	l/m^3	5.7	From irrigation group
Irrigation EC	dS/m	3.6	From irrigation group
Irrigation pH	–	3.4	From irrigation group
Slab weight 1	kg/m^2	0	From irrigation group
Slab weight 2	kg/m^2	0	From irrigation group

A-1 Data cleaning

As with all real-world problems, the available data of a system is noisy, some parts are missing, and sometimes it is inaccurate. In this section a short summary is given on the data cleaning steps which were taken to increase the usability of the available data.

Some measurements, e.g. the radiation sum (J/cm^2) and the cumulative irrigation (l/m^2) contained spikes in the data. These were removed by inspecting their numerical derivatives, and in places where the derivatives showed a successive positive-negative, or negative-positive jump above a pre-defined threshold, the measurements were removed. An example can be seen in Figure A-1.

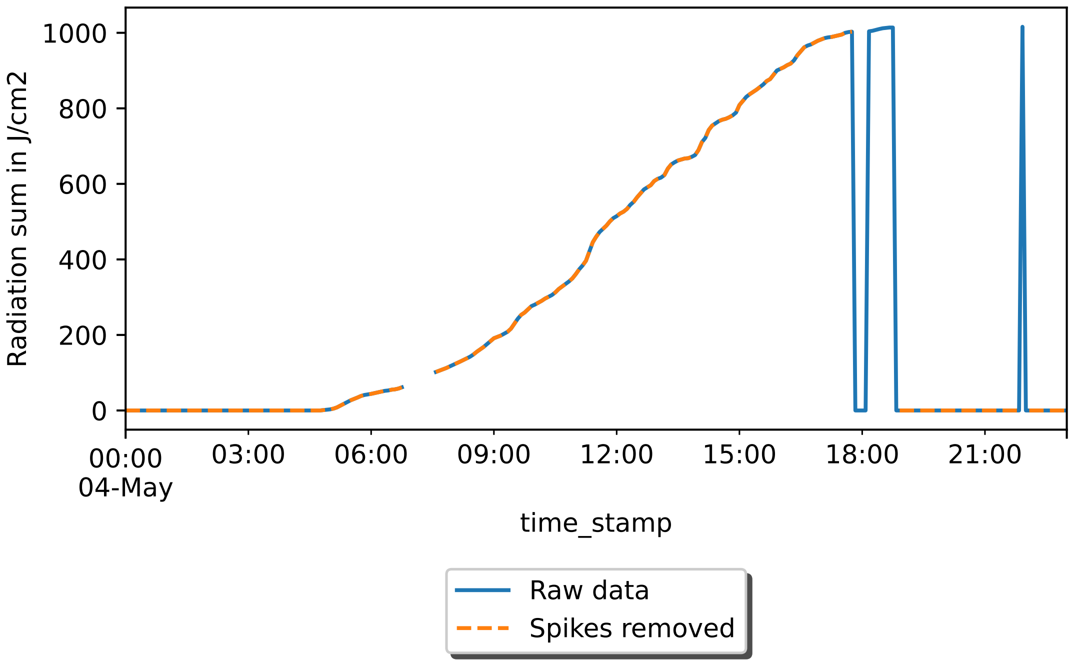

Figure A-1: Removing spikes from the data

After the removal of spikes, the missing datapoints were reconstructed using interpolation. For cumulative irrigation and drain measurements a hybrid interpolation approach was used: during daytime, linear interpolation was used, and during nighttime the gaps were filled using backfill. Other measurements were interpolated using regular linear interpolation.

The cumulative measurements from Table A-1 were all originally reset to 0 every morning. Looking at the numerical derivatives this caused big negative valued jumps which had to be cleared. To remedy this, the accumulated measurements were reconstructed by inspecting their numerical difference. Because of the measurements cumulative nature, the difference only went into a negative regime when the value of the measurement was reset to 0. Using this knowledge, negative jumps below a certain threshold value were cleared from the difference and set to 0, and then the cumulative values were reconstructed by creating the cumulative sum of the difference series.

The missing points in the data were filled using linear interpolation. New measurements were derived using the existing cumulative data. Taking the numerical differences, increments were created of irrigation, drain and radiation sum.

A-2 Derived measurements

Table A-1 Contains the list of the most important variables, which were available in the greenhouse dataset. For the project, new measurements were derived from the existing ones. This section goes over how these new measurements were created.

A-2-1 Water deficit

The gutter scale measurements of the dataset had many different problems. One of them was, that they were not normalized to m^2 values, as any other measurements in the greenhouse. The other problem was, that the interval the measurements were in changed over the course of weeks, so to the same water content two different gutter scale measurements could be matched depending on the day. For the first problem, the following problem is written up:

$$y_{\mathrm{VWC}}(k) = \frac{y_{\mathrm{GS}}(k) - y_{\mathrm{PW}}(k) - c_{\mathrm{SW}}}{r_{\mathrm{GS}} \cdot \rho} \tag{A-1}$$

Where $y_{\mathrm{VWC}}(k)$ is the volumetric water content of the substrate in l/m^2 at time k, $y_{\mathrm{GS}}(k)$ is the gutter scale measurement in kg at time k, $y_{\mathrm{PW}}(k)$ is the weight of the plant taken by the substrate (and not the wires) in kg at time k, and c_{SW} is the constant weight of the dry substrate in kg. ρ is the density of the irrigation water in kg/l, and r_{GS} is the gutter scale coefficient in m^2. r_{GS} defines, how many m^2-s of substrate the gutter scale measures, and this is the coefficient which needs to be calculated.

The idea behind the solution is, that during the early hours, when the first irrigation shot is applied, the plants are the least active in terms of water uptake. Because of this, the assumption is made:

Assumption A.1 (Constant plant weight on small interval)**.** *(i) For sufficiently short time interval* $[k, k+\bar{N}]$*,* $y_{PW} = y_{PW}(k+\bar{N}) =$ *constant*
(ii) The water uptake of the plant on this interval is 0

This assumption is made to avoid circular dependencies between the Water Deficit (WD) and Plant Water Uptake (PWU) calculations. The first assumption is reasonable, because high-wire crops like tomatoes and cucumbers only have a small amount of weight on their stems, which after a few complete trusses are hanging horizontally beside the gutter, as can be seen in Figure A-2

Figure A-2: Substrate grown tomato crop at the end of the crop cycle

Now if the difference is made between (A-1) at time $k+\bar{N}$ and k, we can write:

$$\begin{aligned} y_{\mathrm{VWC}}(k+\bar{N}) - y_{\mathrm{VWC}}(k) &= \frac{\Big(y_{\mathrm{GS}}(k+\bar{N}) - y_{\mathrm{PW}}(k+\bar{N}) - c_{\mathrm{SW}}\Big) - \big(y_{\mathrm{GS}}(k) - y_{\mathrm{PW}}(k) - c_{\mathrm{SW}}\big)}{r_{\mathrm{GS}} \cdot \rho} \\ y_{\mathrm{VWC}}(k+\bar{N}) - y_{\mathrm{VWC}}(k) &= \frac{y_{\mathrm{GS}}(k+\bar{N}) - y_{\mathrm{GS}}(k)}{r_{\mathrm{GS}} \cdot \rho} \end{aligned} \tag{A-2}$$

Using the assumption, that the plant does not take up water in the time interval, the change in Volumetric Water Content (VWC) can come from only irrigation and drain, and so:

$$\begin{aligned}\Big(y_{\text{IRR}}(k+\bar{N}) - y_{\text{DRN}}(k+\bar{N})\Big) - \left(y_{\text{IRR}}(k) - y_{\text{DRN}}(k)\right) &= \frac{y_{\text{GS}}(k+\bar{N}) - y_{\text{GS}}(k)}{r_{\text{GS}} \cdot \rho} \\ r_{\text{GS}} = \frac{y_{\text{GS}}(k+\bar{N}) - y_{\text{GS}}(k)}{\Big(y_{\text{IRR}}(k+\bar{N}) - y_{\text{IRR}}(k) - y_{\text{DRN}}(k+\bar{N}) + y_{\text{DRN}}(k)\Big) \cdot \rho}\end{aligned} \tag{A-3}$$

Where r_{GS} can be calculated using the available data, and the assumption that the irrigation water has the density of $\rho = 1kg/l$. Through experimentation, $\bar{N}$ was chosen to be 3, which corresponds to $3 \cdot h_1 = 15$ minutes. After doing the calculation for different days, and checking the distribution of the collected coefficient values $\boldsymbol{r}_{\text{GS}}$, there were some relatively big outliers, which deviated from the mean of the data considerably. Assuming, that the noise on the estimates was unbiased, a Gaussian kernel density function was fitted on the estimates, and the value chosen for r_{GS} was the value where the density function had its maximum.

Now with r_{GS} known, the right side of (A-1) has only 2 other unknown parts: the plant weight $y_{\text{PW}}(k)$ and the constant substrate weight c_{SW}. Because not any of them is available, and no gutter scale measurements are available of a dry substrate, a trick is used. Taking the highest gutter scale measurement $y_{\text{GS}}^{\max}$ each day, the current gutter scale measurement $y_{\text{GS}}(k)$ is subtracted from it, and corrected with ρ and r_{GS} a new variable emerges: the water content deficit $y_{\text{WD}}(k)$:

$$y_{\text{WD}}(k) = \frac{y_{\text{GS}}^{\max} - y_{\text{GS}}(k)}{r_{\text{GS}} \cdot \rho} \tag{A-4}$$

Let us now assume, that the maximum of the gutter scale measurements happened at time j_0. If (A-4) is expanded, we get:

$$\begin{aligned} y_{\text{WD}}(k) &= \frac{y_{\text{GS}}(j_0) - y_{\text{GS}}(k)}{r_{\text{GS}} \cdot \rho} \\ y_{\text{WD}}(k) &= \frac{(r_{\text{GS}} \cdot \rho \cdot y_{\text{VWC}}(j_0) + y_{\text{PW}}(j_0) + c_{\text{SW}}) - (r_{\text{GS}} \cdot \rho \cdot y_{\text{VWC}}(k) + y_{\text{PW}}(k) + c_{\text{SW}})}{r_{\text{GS}} \cdot \rho} \\ y_{\text{WD}}(k) &= \frac{r_{\text{GS}} \cdot \rho \cdot (y_{\text{VWC}}(j_0) - y_{\text{VWC}}(k)) - (y_{\text{PW}}(j_0) - y_{\text{PW}}(k))}{r_{\text{GS}} \cdot \rho} \end{aligned} \tag{A-5}$$

Because the plant weight changes much slower than the water content of the substrate, a simplification is made, that in a day's interval, $y_{\text{PW}}(j_0) - y_{\text{PW}}(k) \approx 0$. With it, $y_{\text{VWC}}^{\max} = y_{\text{VWC}}(j_0)$, and so:

$$y_{\text{WD}}(k) = y_{\text{VWC}}^{\max} - y_{\text{VWC}}(k) \tag{A-6}$$

$y_{\text{WD}}(k)$ has the physical meaning of how much water is missing until the substrate has reached saturation, and its unit is in l/m^2. Two weaknesses of this derived measurement are substantial: In a predictive setup, $y_{\text{GS}}^{\max}$ is not known beforehand, so an assumption has to be made on its value:

Assumption A.2 (Similarity of daily maximal slab weight)**.** *Today's y_{GS}^{max} is equal to yesterday's y_{GS}^{max}*

This is a risky assumption, because sometimes there can be considerable changes between days. However, the value of $y_{\mathrm{GS}}^{\mathrm{max}}$ can be updated late morning, when the maximal water content was reached. The second problem is, that $y_{\mathrm{PW}}(j_0) - y_{\mathrm{PW}}(k) \approx 0$ could not hold all the time. Adjustment in the wiring of the crop, harvest, and just regular plant growth can all influence the instantaneous value of $y_{\mathrm{PW}}(k)$. However, as Figure A-2 shows, in the generative growth phase of high-wire crops, the weight of plant growth does not greatly affect the gutter scale measurements, because it is mostly held by the wires. With these changes recorded and with c_{SW} known beforehand however, a more accurate estimate on the VWC of the substrate can be given.

A-2-2 Plant water uptake

To accurately estimate, how much water the crop takes up, a reference (or ground truth) value needs to be available. From Table A-1 the cumulative absorption is the closest to describe the water uptake process.

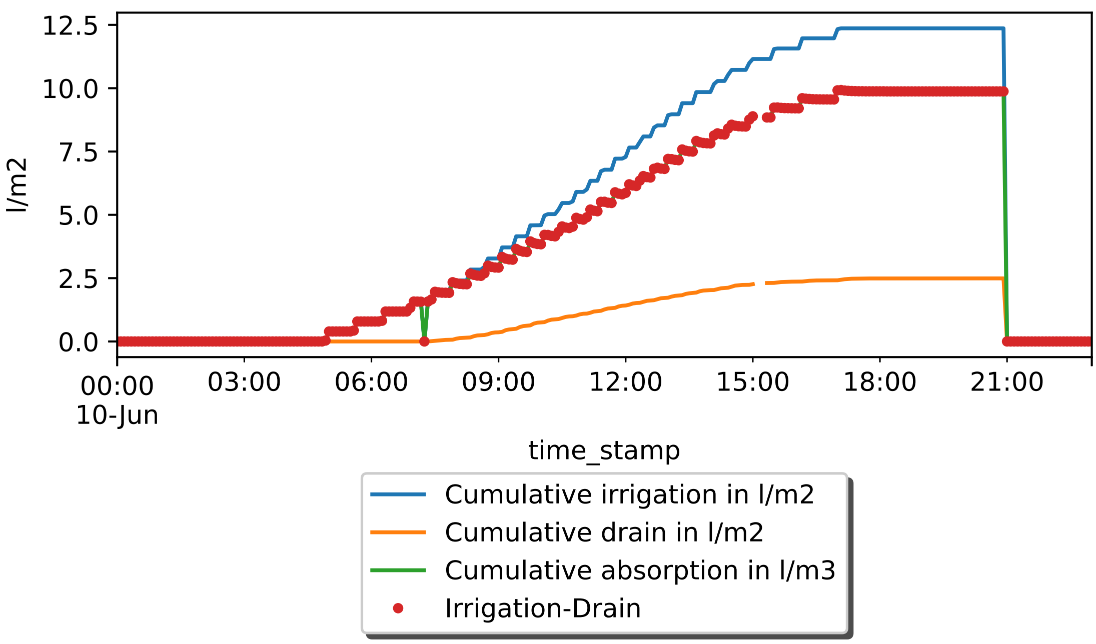

Figure A-3: Cumulative absorption data

As it can be seen on Figure A-3 The absorption is nothing else, but the difference between the cumulative irrigation and drain measurements. Because of some delay between the irrigation and drain measurements, the absorption can have negative slope. This is not favorable.

To calculate, how much water the plants take up, the following water balance equation can be used:

$$y_{\mathrm{WC}}(k+1) \;= y_{\mathrm{WC}}(k) + Q_{\mathrm{IRR}}(k) - Q_{\mathrm{PWU}}(k) - Q_{\mathrm{DRN}}(k) + \varepsilon(k) \tag{A-7}$$

Where y_{WC} is the water content, $Q_{\mathrm{IRR}}(k)$ is the amount of irrigation water supplied, $Q_{\mathrm{DRN}}(k)$ is the amount of water lost through drain and $Q_{\mathrm{PWU}}(k)$ is the amount of water taken up by the plant in the interval between $[k, k+1]$. $\varepsilon(k)$ is an unmeasured disturbance which can come from rewiring of the hanging crop, plant swelling, fruit harvest, delays between the sensors, etc.

Because $\varepsilon(k)$ can not be estimated accurately, an assumption is made:

Assumption A.3 (Declining cumulative water balance error)**.** *with enough time as $N \to \bar{N}$, $\sum_{i=k}^{N} \varepsilon(i) \to 0$.*

Reformulating the water balance equation:

$$Q_{\mathrm{PWU}}(k) = y_{\mathrm{WC}}(k+1) - y_{\mathrm{WC}}(k) - Q_{\mathrm{IRR}}(k) + Q_{\mathrm{DRN}}(k) - \varepsilon(k) \tag{A-8}$$

It is not easy to guess, when the influence of $\varepsilon(k)$ has been mitigated. One way, is to make the assumption, that $Q_{\mathrm{PWU}}(k) > 0$, which is reasonable, because tomato plants should not be able to push water out through their roots. The best guess for the other Q variables is the difference between their corresponding measurements at time k and $k+1$:

$$\begin{aligned} Q_{\mathrm{PWU}}(k) = y_{\mathrm{WC}}(k+1) - y_{\mathrm{WC}}(k) - (y_{\mathrm{IRR}}(k+1) - y_{\mathrm{IRR}}(k)) + \\ + (y_{\mathrm{DRN}}(k+1) - y_{\mathrm{DRN}}(k)) - \varepsilon(k) \end{aligned} \tag{A-9}$$

With $y_{\mathrm{IRR}}(k)$ being the cumulative irrigation and $y_{\mathrm{DRN}}(k)$ the cumulative drain measurement at time k. The problem arises: finding a small enough $\bar{N} \in \mathbb{Z}^+$, s.t. the percentage of negative plant water uptake values is minimal:

$$\begin{aligned} \bar{Q}_{\mathrm{PWU}}(k) = y_{\mathrm{WC}}(k+\bar{N}) - y_{\mathrm{WC}}(k) - (y_{\mathrm{IRR}}(k+\bar{N}) - y_{\mathrm{IRR}}(k)) + \\ + (y_{\mathrm{DRN}}(k+\bar{N}) - y_{\mathrm{DRN}}(k)) - \underbrace{\sum_{i=k}^{\bar{N}} \varepsilon(i)}_{\approx 0} \end{aligned} \tag{A-10}$$

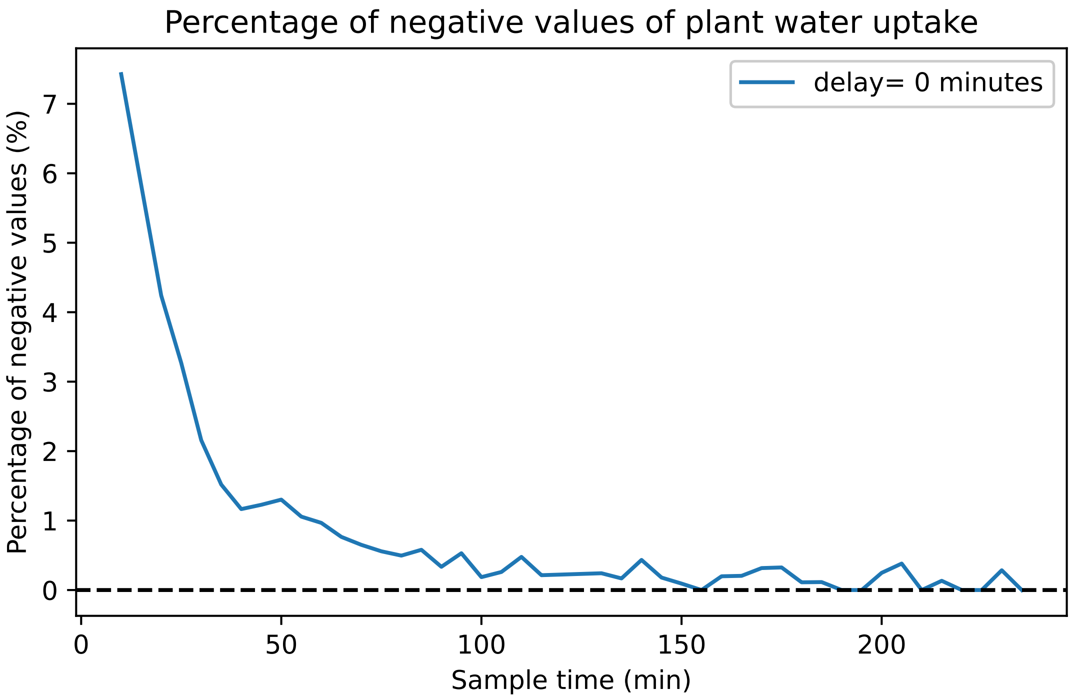

Figure A-4: Ratio of negative PWU values

As Figure A-4 depicts, the percentage of negative values sinks more slowly after around 1 hour. This is ideal, because the meteorological forecasts are usually provided per hour. Now, with h_2=1 hour=60 minutes, $\bar{N} = 12$, and so, the water taken up by the plant is calculated per hour, using the following equation:

$$\begin{aligned} \bar{Q}_{\mathrm{PWU}}(l) = y_{\mathrm{WC}}(k+12) - y_{\mathrm{WC}}(k) - (y_{\mathrm{IRR}}(k+12) - y_{\mathrm{IRR}}(k)) + \\ + (y_{\mathrm{DRN}}(k+12) - y_{\mathrm{DRN}}(k)) \end{aligned} \tag{A-11}$$

With $l = 12k$. Because of some irregularities, the estimated water uptake is still very noisy. Using Fast Fourier Transform (FFT), the series of estimated hourly plant water uptake values $\hat{\boldsymbol{q}} = \left[\bar{Q}_{\mathrm{PWU}}(0), \bar{Q}_{\mathrm{PWU}}(1), \ldots, \bar{Q}_{\mathrm{PWU}}(n/12-1)\right]$ are transformed into frequency domain. This was done using the process described in [34]. Figure A-5 shows the Power Spectral Density (PSD) of the series.

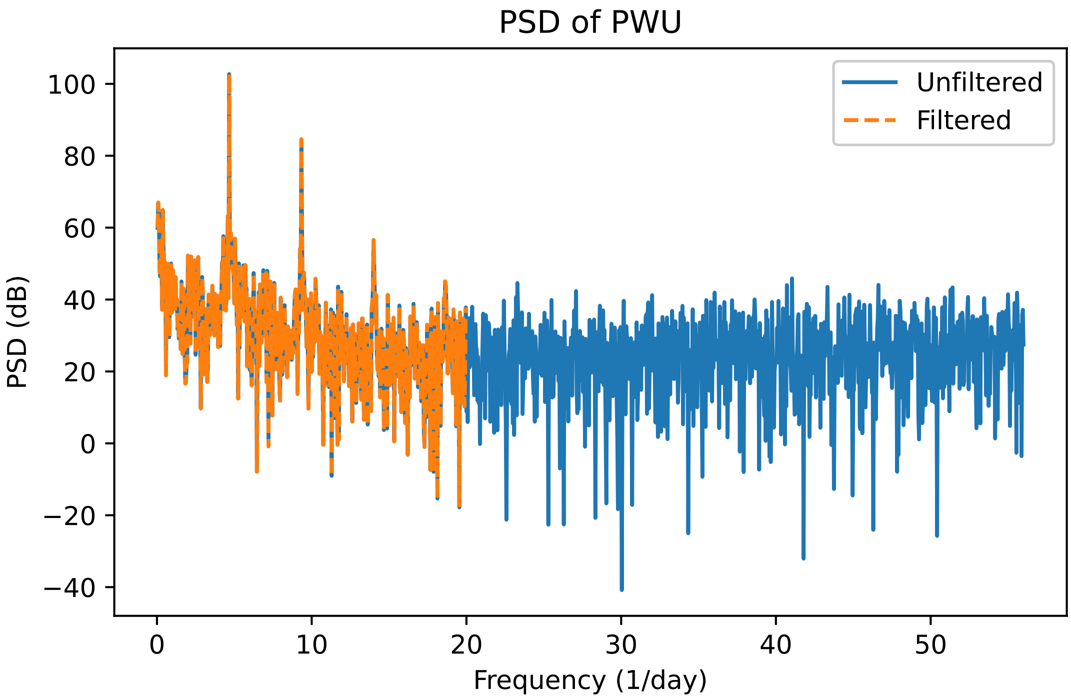

Figure A-5: Power spectral density of the plant water uptake calculations

As it can be seen in Figure A-5, there are well defined peaks in the power spectrum of the signal. After experimentation, the frequency content above 20/day was cut, and the series was transformed back to time-domain by an inverse FFT. A sample of the resulting filtered series can be seen in Figure A-6, where it is both compared to the original unfiltered calculation, and the absorption data.

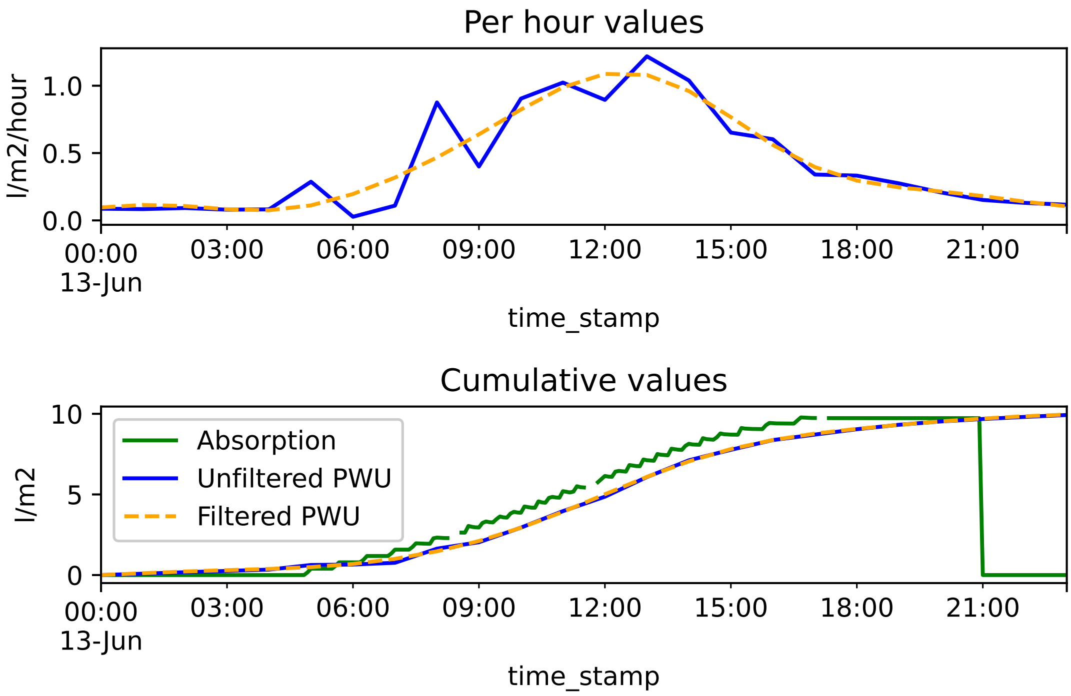

Figure A-6: Filtered PWU values

As it can be seen, the PWU curve has smoothed out considerably, which is more in line with the expectations on how plants consume water. The smoothing did not change long term accuracy however, so the cumulative water uptake curve follows both the absorption and the unfiltered PWU curves.

A-3 Resampling

The calculation of PWU requires the resampling of the data to h_2= 1 hour. The resampling was done using the cumulative sums of the incremental data (drain, irrigation, gutter scale measurements and water deficit). After the resampling, the PWU was calculated using (A-11), and then filtered as described in Section A-2-2. Non cumulative measurements, like the humidity deficit and temperature were resampled taking the average of the intersample period.

Appendix B

Validation on other greenhouse data

B-1 Greenhouse 2

The greenhouse is located in the Netherlands. Long truss tomatoes were grown in the compartment, and the data was recorded between 11/04/2021-01/08/2021. The figures show validation data between 27/06/2021-01/08/2021

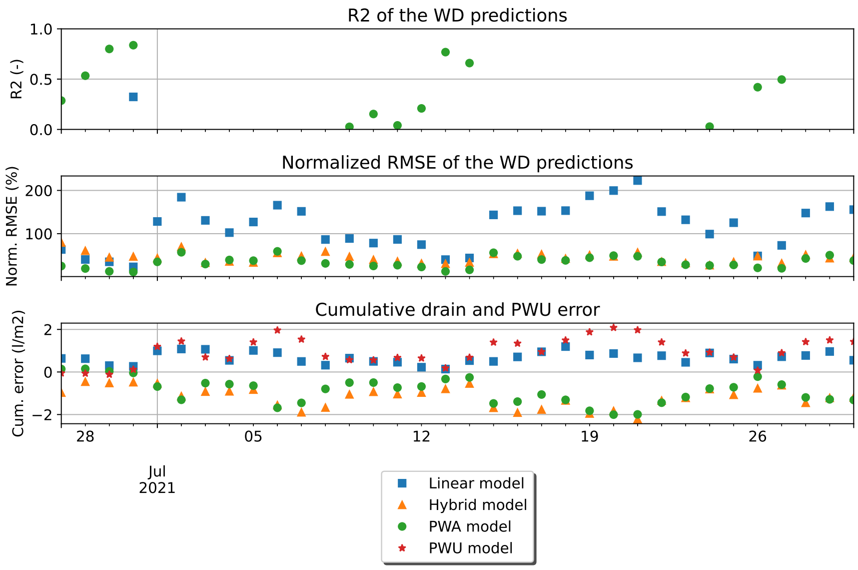

Figure B-1: Accuracy with the static identification approach

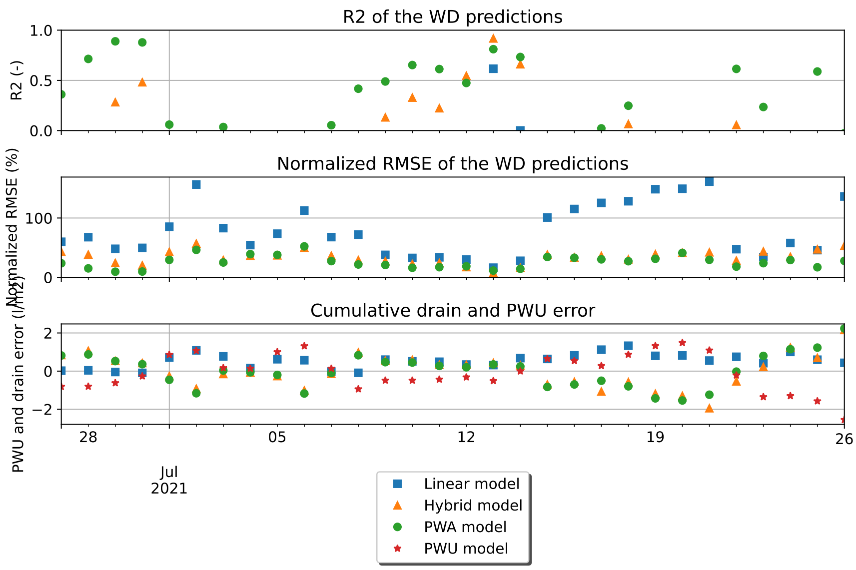

Figure B-2: Accuracy with the progressive identification approach

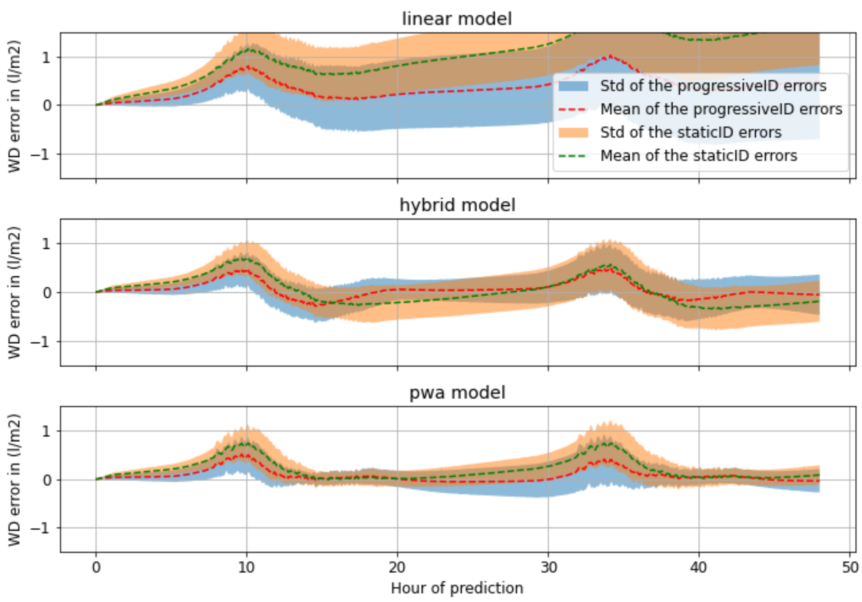

Figure B-3: Error distribution of the predictions

B-2 Greenhouse 3

The greenhouse is the same as greenhouse 2, and the crop is also long truss tomato. The data was recorded between 11/04/2022-20/06/2022. The figures show validation data between 01/06/2021-22/06/2022. This greenhouse data was used during the validation of forecast climate parameters because it was the only dataset where it was possible to access the forecast parameters. The figures below show the accuracy measures regarding historical data, not forecast.

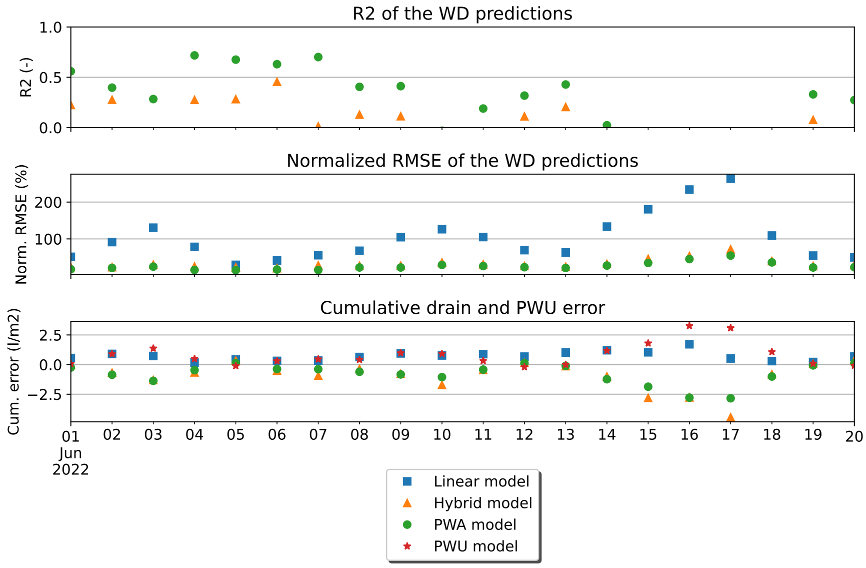

Figure B-4: Accuracy with the static identification approach

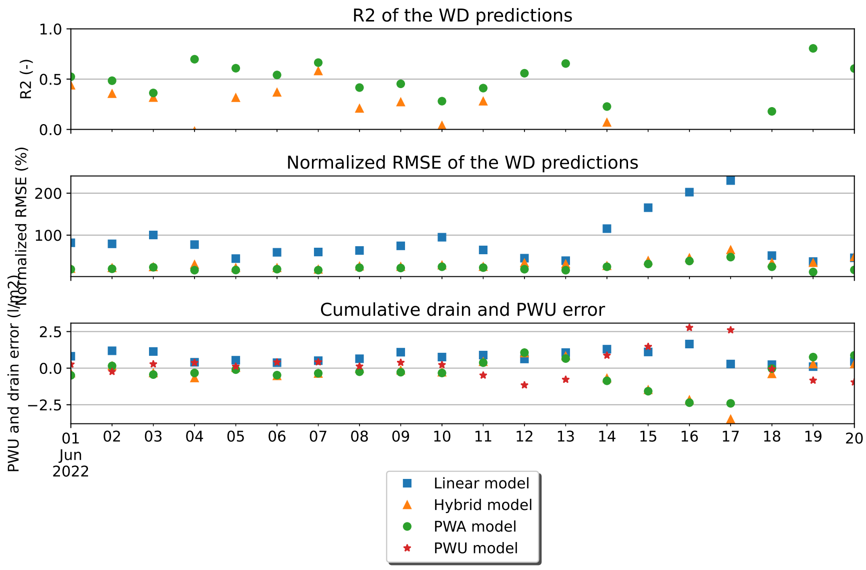

Figure B-5: Accuracy with the progressive identification approach

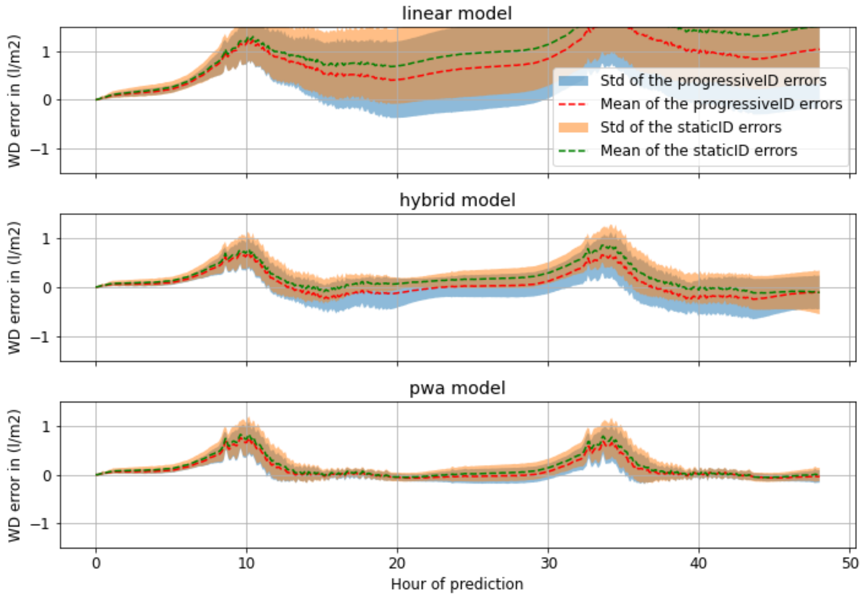

Figure B-6: Error distribution of the predictions

B-3 Greenhouse 4

The greenhouse is located in Canada. The crop is cherry tomato. The data was recorded between 16/04/2022-27/06/2022. The figures show validation data between 01/06/2021-27/06/2022.

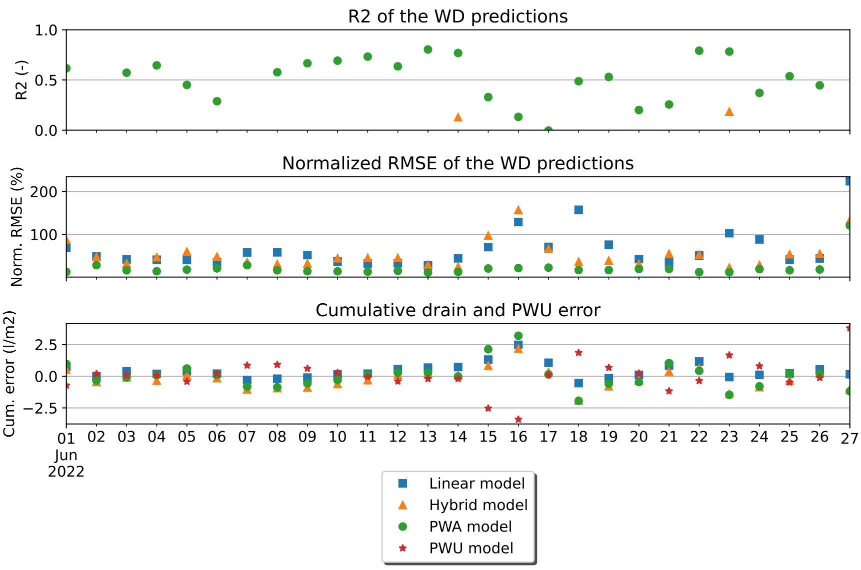

Figure B-7: Accuracy with the static identification approach

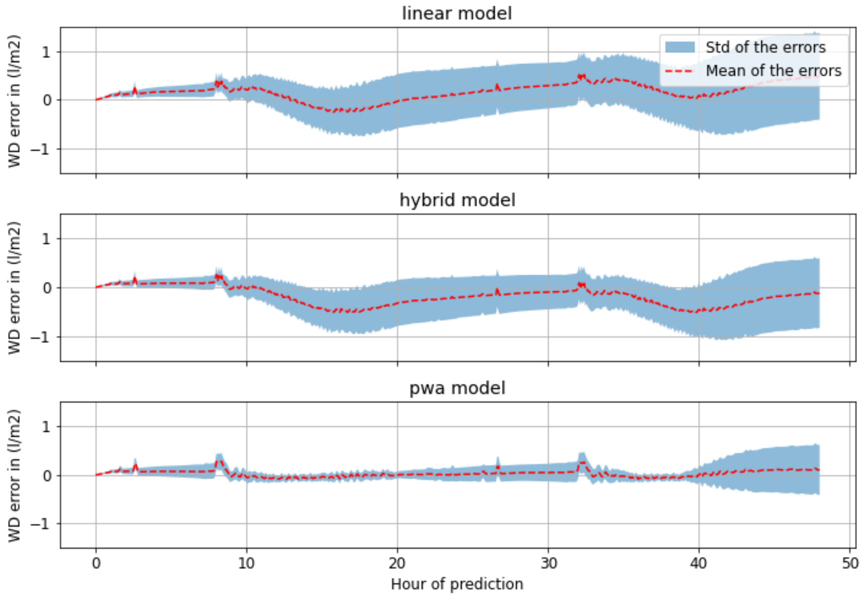

Figure B-8: Error distribution of the predictions

Bibliography

[1] W. H. Organization, *The state of food security and nutrition in the world 2018: building climate resilience for food security and nutrition.* Food & Agriculture Org., 2018.

[2] Rabo Bank, "World Vegetable Map 2018." [Online]. Available: https://research.rabobank.com/far/en/sectors/regional-food-agri/world_vegetable_map_2018.html

[3] "Cuesta Roble releases 2019 global greenhouse statistics - Produce Grower." [Online]. Available: https://www.producegrower.com/article/cuesta-roble-2019-global-greenhouse-statistics/

[4] C. Stanghellini, "Horticultural production in greenhouses: Efficient use of water," *Acta Horticulturae*, vol. 1034, pp. 25–32, 2014.

[5] "Upscaling of greenhouse vegetable production." [Online]. Available: https://www.cbs.nl/en-gb/news/2018/16/upscaling-of-greenhouse-vegetable-production

[6] G. Avigad and M. Gholami, "Automated monitoring and irrigation of plants in a controlled growing environment," 1 2020. [Online]. Available: https://www.patentguru.com/WO2020014773A1

[7] "The State Of Labor In The Greenhouse Industry - Greenhouse Grower." [Online]. Available: https://www.greenhousegrower.com/management/grow-initiative/the-state-of-labor-in-the-greenhouse-industry/

[8] "What Is the Current State of Labor in the Greenhouse Industry? - Greenhouse Grower." [Online]. Available: https://www.greenhousegrower.com/management/what-is-the-current-state-of-labor-in-the-greenhouse-industry/

[9] "8 Trends Shaping the Greenhouse Structures Industry - Greenhouse Grower." [Online]. Available: https://www.greenhousegrower.com/technology/8-trends-shaping-the-greenhouse-structures-industry/

[10] S. Hemming, F. De Zwart, A. Elings, I. Righini, and A. Petropoulou, “Remote control of greenhouse vegetable production with artificial intelligence—greenhouse climate, irrigation, and crop production,” *Sensors (Switzerland)*, vol. 19, no. 8, 4 2019.

[11] S. Hemming, F. de Zwart, A. Elings, A. Petropoulou, and I. Righini, “Cherry tomato production in intelligent greenhouses-sensors and ai for control of climate, irrigation, crop yield, and quality,” *Sensors (Switzerland)*, vol. 20, no. 22, pp. 1–30, 11 2020.

[12] A. Pardossi, G. Carmassi, C. Diara, L. Incrocci, R. Maggini, and D. Massa, “Fertigation and substrate management in closed soilless culture,” *Pisa: University of Pisa*, 2011.

[13] A. Frija, A. Chebil, S. Speelman, J. Buysse, and G. Van Huylenbroeck, “Water use and technical efficiencies in horticultural greenhouses in Tunisia,” *Agricultural Water Management*, vol. 96, no. 11, pp. 1509–1516, 11 2009.

[14] P. van Weel, P. Geelen, and J. Voogt, *PLANT EMPOWERMENT The Basic Principles*, 3rd ed., E. Hardy, Ed. Letsgrow.com, 8 2021.

[15] L. P. Schoonen, A. T. Cobbenhagen, and W. P. Heemels, “Optimal irrigation management for large-scale arable farming using model predictive control,” in *IFAC-PapersOnLine*, vol. 52, no. 30. Elsevier B.V., 2019, pp. 56–61.

[16] S. K. Saleem, D. K. Delgoda, S. K. Ooi, K. B. Dassanayake, L. Liu, M. N. Halgamuge, and H. Malano, “Model predictive control for real-time irrigation scheduling,” in *IFAC Proceedings Volumes (IFAC-PapersOnline)*, vol. 46, no. 18 PART 1. IFAC Secretariat, 2013, pp. 299–304.

[17] C. Lozoya, C. Mendoza, L. Mejía, J. Quintana, G. Mendoza, M. Bustillos, O. Arras, and L. Solís, “Model predictive control for closed-loop irrigation,” in *IFAC Proceedings Volumes (IFAC-PapersOnline)*, vol. 19. IFAC Secretariat, 2014, pp. 4429–4434.

[18] E. A. Abioye, M. S. Z. Abidin, M. N. Aman, M. S. A. Mahmud, and S. Buyamin, “A model predictive controller for precision irrigation using discrete lagurre networks,” *Computers and Electronics in Agriculture*, vol. 181, 2 2021.

[19] A. Pawlowski, J. A. Sánchez-Molina, J. L. Guzmán, F. Rodríguez, and S. Dormido, “Evaluation of event-based irrigation system control scheme for tomato crops in greenhouses,” *Agricultural Water Management*, vol. 183, pp. 16–25, 2017.

[20] D. N. Moriasi, J. G. Arnold, M. W. Van Liew, R. L. Bingner, R. D. Harmel, and T. L. Veith, “Model evaluation guidelines for systematic quantification of accuracy in watershed simulations,” *Transactions of the ASABE*, vol. 50, no. 3, pp. 885–900, 2007.

[21] J. E. Nash and J. V. Sutcliffe, “River flow forecasting through conceptual models part I—A discussion of principles,” *Journal of hydrology*, vol. 10, no. 3, pp. 282–290, 1970.

[22] H. V. Gupta, S. Sorooshian, and P. O. Yapo, “Status of automatic calibration for hydrologic models: Comparison with multilevel expert calibration,” *Journal of hydrologic engineering*, vol. 4, no. 2, pp. 135–143, 1999.

[23] N. Katsoulas and C. Stanghellini, “Modelling crop transpiration in greenhouses: Different models for different applications,” *Agronomy*, vol. 9, no. 7, 7 2019.

[24] R. G. Allen, L. S. Pereira, D. Raes, M. Smith, and others, "Crop evapotranspiration-Guidelines for computing crop water requirements-FAO Irrigation and drainage paper 56," *Fao, Rome*, vol. 300, no. 9, p. D05109, 1998.

[25] G. Stanhill and J. S. Albers, "Solar radiation and water loss from glasshouse roses," *Journal of the American Society for Horticultural Science*, 1974.

[26] M. Baille, A. Baille, and D. Delmon, "Microclimate and transpiration of greenhouse rose crops," *Agricultural and Forest meteorology*, vol. 71, no. 1-2, pp. 83–97, 1994.

[27] C. Stanghellini, *Transpiration of greenhouse crops: an aid to climate management.* Wageningen University and Research, 1987.

[28] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, no. 3, pp. 407–427, 1999.

[29] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model predictive control: theory, computation, and design.* Nob Hill Publishing Madison, WI, 2017, vol. 2.

[30] Gurobi Optimization LLC, "Gurobi Optimizer Reference Manual," 2022. [Online]. Available: https://www.gurobi.com

[31] H. Yan, S. Huang, C. Zhang, M. C. Gerrits, G. Wang, J. Zhang, B. Zhao, S. J. Acquah, H. Wu, and H. Fu, "Parameterization and application of stanghellini model for estimating greenhouse cucumber transpiration," *Water (Switzerland)*, vol. 12, no. 2, 2 2020.

[32] H. Arbabi, "Introduction to Koopman operator theory of dynamical systems," *Introduction to Koopman operator theory of dynamical systems*, 2018.

[33] M. Korda and I. Mezić, "Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control," *Automatica*, vol. 93, pp. 149–160, 2018.

[34] "IPython Cookbook: Analyzing the frequency components of a signal with a Fast Fourier Transform." [Online]. Available: https://ipython-books.github.io/101-analyzing-the-frequency-components-of-a-signal-with-a-fast-fourier-transform/

Glossary

List of Acronyms

ARX	Autoregressive with Exogenous Input
EC	Electrical Conductivity
ET	Evapotranspiration
FFT	Fast Fourier Transform
HD	Humidity Deficit
LAI	Leaf Area Index
MILP	Mixed Integer Linear Programming
MIMO	Multiple Input Multiple Output
MIQP	Mixed Integer Quadratic Programming
MLD	Mixed Logical Dynamical
MPC	Model Predictive Control
NSE	Nash-Sutcliffe efficiency
PBIAS	Percent-bias
PSD	Power Spectral Density
PWA	Piecewise Affine
PWS	Plant Water Status
PWU	Plant Water Uptake
QP	Quadratic Programming
RMSE	Root Mean Square Error
RSR	Ratio of the Root Mean Square Error to the standard deviation of measured data
SS	State-Space
VPD	Vapor Pressure Deficit
VWC	Volumetric Water Content
WD	Water Deficit