TALLINN UNIVERSITY OF TECHNOLOGY
SCHOOL OF ENGINEERING
Department of Electrical Power Engineering and Mechatronics

DESIGN AND IMPLEMENTATION OF MODEL PREDICTIVE CONTROL

Mudel-ennetava juhtimise projekteerimine ja rakendamine

MASTER THESIS

Ardan APAYDIN

Student code: 194361MAHM

Supervisor: Professor Eduard Petlenkov
Department of Computer Systems/ TalTech

Co-Supervisor: Dr. Dmitry Shvarts,
Ph.D. Researcher at Mechatronics Department /TalTech

Tallinn 2021

(*On the reverse side of title page*)

AUTHOR'S DECLARATION

Hereby I declare that I have written this thesis independently.
No academic degree has been applied for based on this material. All works, major viewpoints and data of the other authors used in this thesis have been referenced.

"10" JUNE 2021

Author: Ardan APAYDIN
/signature /

Thesis is in accordance with terms and requirements

"......."................... 20….

Supervisor: …........................
/signature/

Accepted for defence

"......."...................20… .

Chairman of theses defence commission: ....
/name and signature/

Non-exclusive Licence for Publication and Reproduction of GraduationTthesis[1]

I, Ardan APAYDIN (name of the author) (date of birth: 29.08.1994) hereby

1. grant Tallinn University of Technology (TalTech) a non-exclusive license for my thesis Design and Implementation of Model Predictive Control,

(*title of the graduation thesis*)

supervised by Eduard PETLENKOV,

(*Supervisor's name*)

1.1 reproduced for the purposes of preservation and electronic publication, incl. to be entered in the digital collection of TalTech library until expiry of the term of copyright;

1.2 to be published via the web of TalTech, incl. to be entered in the digital collection of TalTech library until the expiry of the term of copyright.

1.3 I am aware that the author also retains the rights specified in clause 1 of this license.

2. I confirm that granting the non-exclusive license does not infringe third persons' intellectual property rights, the rights arising from the Personal Data Protection Act or rights arising from other legislation.

---

[1] *Non-exclusive Licence for Publication and Reproduction of Graduation Thesis is not valid during the validity period of restriction on access, except the university's right to reproduce the thesis only for preservation purposes.*

______________ (*signature*)

______________ (*date*)

THESIS TASK

Student: Ardan APAYDIN, 194361MAHM (name, student code)
Study programme, MAHM02/18 - Mechatronics (code and title)
main speciality: Mechatronics
Supervisor(s): Professor Eduard Petlenkov (position, name, phone)
Consultants: Vitali VANSOVITŠ, Service Manager (name, position)
Valmet Automation, vitali.vansovits@taltech.ee (company, phone, e-mail)

Thesis topic:

(in English) DESIGN AND IMPLEMENTATION OF MODEL PREDICTIVE CONTROL

(in Estonian) Mudel-ennetava juhtimise projekteerimine ja rakendamine

Thesis main objectives:

1. Study and implement Data-Driven identification methods to the Multi-tank System in Alpha-lab Control laboratory.
2. Obtain accurate state-space models to design a model predictive control
3. Study the tuning of the model predictive controller to obtain offset-free control for a multivariable system.
4. Design an external model-based controller to control the system externally from another computer.
5. Define a real-time communication protocol between Multitank system and the external controller.
6. Design case studies to analyse the closed-loop performance of the external model-based controller

Thesis tasks and time schedule:

No	Task description	Deadline
1.	Studying the Identification methods used in industrial plants	December 20
2.	Implementation of identification methods to the laboratory equipment	February 21
3.	Studying the Model Predictive Controllers	March 21
4.	Closed-loop control of the single-input single-output system with a model-based controller	March 21
5.	Identification of the multivariable system	April 21
6.	Closed-loop control of the multivariable system with a model-based controller	April 21

7.	Design and Implementation of the External MPC and Closed-loop control of the multivariable system	May 21

Language: English **Deadline for submission of thesis:** "10" JUNE 2021

Student: Ardan APAYDIN .... "9" JUNE 2021
/signature/
Supervisor: Eduard PETLENKOV "9" JUNE 2021
/signature/
Consultant: .... "......."......................20….a
/signature/
Head of study programme: Mart TAMRE "10" JUNE 2021
/signature/

Terms of thesis closed defence and/or restricted access conditions to be formulated on the reverse side

CONTENTS

Table of Figures

Abbreviations

CV	Controlled Variable
DCS	Distributed Control Systems
DMC	Dynamic Matrix Control
FIR	Finite impulse response
FSR	Finite step response
GPC	General Predictive Controller
HIECON	Hierarchical Constraint Control
IDCOM	Identification and Command
LQG	Linear Quadratic Gaussian
MIMO	Multi-Input Multi-Output
MISO	Multi-Input Single-Output
MPC	Model Predictive Control
MPC-RG	Reference Governor MPC
MV	Manipulated Variable
NNARX	Neural Network AutoRegressive with eXogonous Input
NNOE	Neural Network Output Error
PCT	Profit Controller
PEM	Predictive Error Minimalization
PID	Proportional - Integral - Derivative
QDMC	Quadratic Dynamic Matrix Control
RMPC	Robust Model Predictive Control
RT-DAC/PCI	Real-Time Data Acquisition Board
SISO	Single-Input Single-Output
SMCA	Single Multivariable Control Algorithm
SMOC	Shell Multivariable Optimizing Controller
TS	Takagi-Sugeno

1 INTRODUCTION

The signs of Global Warming are determining the way to obtain a comprehensive design of industrial applications. Finding the most efficient and sustainable solution has become an engineering challenge, explicitly focusing on reducing consumption and energy waste to ensure ecological sustainability. In process automation, developing high-performance and reliable industrial controllers is one way of increasing plant efficiency. A significant portion of current industrial controllers is constituted by proportional – integral - derivative (PID) controllers due to their simple favorable implementation and their applicability to a wide range of industrial control tasks. Although PID controllers provide a simple controller solution, their simple structure vanishes for multivariable control tasks yielding a complex and inefficient controller structure[1]. Considering the increasing complexity in industrial applications with the envisioned design of Industry 4.0, data-driven intelligent control systems offer more extensive and advanced control schemes to ensure optimal plant efficiency.

Model Predictive Control is an advanced intelligent control method that provides optimal model-based input manipulation. Model Predictive Control, by its nature, is highly ideal to govern control tasks for multivariable systems[2]. MPC consists of techniques to utilize predictions for the optimal computation of variables in a process and even for future states. MPC algorithm predicts the future behavior of the process, by computing the adjustment of the sequence of future manipulated variables, in each sampling time of the control [3]. MPC concepts have found a vast field of application in industry and gained popularity due its high performance and reliability for long operating times [4].

This thesis comprises the studies for developing and validating model-based control approaches to facilitate multivariable MPC implementation to a district heating plant. The district heating plants operate boiler to generate heating energy. The heating plant and the process within the boiler constitute a multivariable process as gas flow, boiler water flow, plant water flow, inlet water temperature as input, boiler outlet temperature, and plant outlet temperature as output. It is denoted in Statistics Estonia that there exist 3050 boilers in Estonia in 2017, and most of them are employed by district heating plants[5][6]. Adjusting MPC solutions in heat production is a promising advancement in boosting the efficiency of conventional heating plants and ultimately reducing carbon emissions.

The thesis is the follow-up study for the PhD thesis of Vitali VANSOVITŠ "*Control Advanced Control of District Heating Processes*" [7] constituting the Model Predictive

Control branch of the Intelligent Control System Studies of TalTech Alpha-Lab Control Laboratory. In the previous studies, a MPC Software was proposed to externally control the district heating plant with distributed control system (DCS) configuration. The MPC Software successfully controlled a multi-input single-output (MISO) system and provided promising results for implementing to an actual plant. Studies in this thesis focus on validating the MPC Software's control performance over an identified multi-input multi-output (MIMO) Multi-tank laboratory system [8]. The validation method compares two MIMO MPC performance results; one designed and tuned in MATLAB Model Predictive Control application, and the other is intended to be the external MIMO MPC controller using the MPC Software. Results will show the applicability of the MPC Software to any industrial control task if the state-space model of the process is identified. The studies give a detailed insight into the design and implementation of a model predictive controller, which consists of multivariable model identification and validation of the identified model, tuning approaches for a goal-oriented model-based controller, data transfer protocols for real-time multivariable controllers.

The contributions of this thesis include:

1. Studies of data driven identification techniques.
2. The implementations of identification experiments on SISO and MIMO systems, and validation studies on the obtained state-space models.
3. Studies of tuning approaches of a Model Predictive Control.
4. Configuration of communication interface between the MPC Software and the Multi-tank Laboratory system [8].
5. The Practical implementation of the MPC Software as an external controller, generation of case studies, analysis of the results, and formulations on designing the process control.

Thesis includes 7 Chapters:

Chapter 1
The introduction of the work is provided. This Chapter includes the problem statements and objective of the thesis.

Chapter 2
Contains the discovery and development MPC with its previous implementations in various industry. Current shortcomings of the MPC are discussed and justification of this work is stated.

Chapter 3

Gives a detailed introduction to the feature of MPC. The algorithm of state-space based MPC is detailed with mathematical equations.

Chapter 4

The chapter gives overview of model identification techniques and includes detailed description of the considered identification method to be used for obtaining the multivariable model of the Multi-tank system.

Chapter 5

Details the identification methodology used for SISO, MISO and MIMO systems.

Chapter 6

The chapter details the implementation and tuning of designed model predictive controllers. The case studies are introduced, and the results are monitored.

Chapter 7

In the final chapter, results of the thesis are reviewed, and overall conclusions are drawn.

2 LITERATURE REVIEW AND BACKGROUND

2.1 History of Industrial MPC in Literature

The applicability and effectualness of MPC have been denoted in a vast number of surveys related to various fields of industry. The increasing number of reported applications and remarkable improvements in the computing power of controllers have granted MPC a prospect of becoming a control method having favorable commercial values [3], [9]. In this chapter, the background of Model Predictive Control is reviewed regarding previous technical studies and journals in control theory literature.

MPC systems have gained popularity lately in the industry, although initial implementations of MPC go down to the late 70s. First examples of model-based control were formed as Linear Quadratic Gaussian (LQG) control. LQG describes the dynamics of the controlled system as a set of linear differential equations. The problem of LQ concerns the cost function, which is the sum deviation of measurements from their desired values. The objective of the LQ problem is to minimize the cost function. Kalman proposed[10] that, representing the states of the linear system as a discrete-time state-space model enables a function that can estimate the future states. Implementation of discrete state-space model into LQ problem generates weights matrices as the coefficient matrices for states and inputs. Optimization, so-called "tuning" of weights matrices, became the objective to obtain a minimum cost function. It is depicted in[11] LQG algorithm provides successful results in getting offset-free outputs in steady-state targets. Although the initial LQG algorithm showed successful outcomes in their particular applications, the initialized model was unsuitable for control applications in the industrial level. The significant problems of the initial LQG models were their feasibility for only linear systems; moreover, they lacked systematic tuning guidelines, and most importantly, physical constraints of the controlled process were not modeled [12][13].

First examples of modern model-based control approaches in the industry have been proposed as IDCOM (Identification and Command) [14] and DMC (Dynamic matrix control) [15]. They use a model-based control methodology that comprises online optimization of manipulated variables (MV). According to the past MV values, the proposed models predict and optimize the future behavior of the plant over an interval known as prediction horizon. Additionally, two model-based control approaches utilize test data of the controlled system to generate the heuristic dynamical model by identification methods. Richalet's s IDCOM approach models the plant-based impulse response and performs quadratic objectives over the prediction horizon. Optimal inputs and outputs are computed using a heuristic model algorithm, and output is acquired based on a reference trajectory. It was reported that IDCOM was implemented into industrial control of a fluid catalytic cracking unit and a steam generator.

On the other hand, Cutler & Ramaker described linear identification method based on step response for their DMC. Optimal inputs are calculated with respect to a least-square problem. Future plant behavior is specified by manipulating the input variables to reach the set point as close as possible. DMC approach was designed to be implemented in petrochemical processes. DMC and IDCOM are the pioneer studies describing multi-level hierarchical control function, which is considered fundamental to advanced control applications. However, due to their heuristic algorithm, both control approaches were not comprehensive enough to deal with systems with constraints. Therefore, they are considered as the First Generation MPC in surveys in automation literature [3], [9], [13], [16].

Constraint handling is crucial for controlling a system concerning its physical capabilities. Several developments were made to DMC algorithm to address the constraint handling. QDMC algorithm was described in [17] as a quadratic program (QP) combined with DMC, which explicitly reveals input/output constraints. Such a result is achieved by posing the MPC problem as a QP. Additionally, improved QDMC is published with detailed optimization and tuning algorithms [16]. The QDMC requires a linear step response model of the controlled plant; based on this model, future plant output behavior is predicted by optimizing the inputs according to the set point. Optimal inputs are calculated as the solution of a quadratic problem, while the objective is defined over a finite prediction horizon. In [16] the QDMC was introduced to industrial pyrolysis furnace application. The task was to control the stream temperature in three locations in the furnace by adjusting fuel gas pressure in three burners. The QDMC algorithm showed successful results in executing the control process according to the input/output constraints. With the deliberate introduction of constraint handling and detailed implementation methods, the QDMC is considered Second Generation MPC.

After the innovation of MPC technology, surveys reported the economic advantages of MPC. It was mentioned in a model predictive technology survey that Richalet's IDCOM based model predictive heuristic control provided $150.000/yr profit in fractionator application [3]. The economic benefits of MPC granted popularity in the market, and MPC gained commercial value. As the popularity of MPC increased, the second-generation MPC algorithms were implemented on more extensive and more complex systems. This revealed the incompleteness of second-generation MPC in many practical aspects; thus third-generation MPC is developed to address the problems. Setpoint Inc. developed Single Multivariable Control Algorithm (SMCA), QDMC was bought and improved by Aspentech, Shell Oil developed Shell Multivariable Optimizing Controller (SMOC), Adersa introduced Hierarchical Constraint Control (HIECON), Profimatics developed Profit Controller (PCT) and Honeywell had their Robust Model Predictive Control (RMPC).

The innovational features of third-generation MPC algorithms are addressed in three parts. The first innovation is their mechanism to prioritize the constraints. The second-generation MPC algorithm was found to be insufficient for violating input/output constraints to some extent. In real applications, hard constraints define the system's physical capabilities, and some hard constraints can have critical importance. Third-generation MPC technologies explained priority concept to constraints, so that hard constraints cannot be excluded in the formulation. The constraint formulation idea allows some violation of soft constraints by depicting them as quadratic penalties, while hard constraints are not being violated.

The second innovation of third-generation MPC is the fault tolerance by determining the controlled sub-process. The controller determines which input variable to be manipulated and which controlled variable (CV) to be controlled. For a given control action, MV must meet the constraint conditions and good measurement status for a CV must be ensured. Additionally, variables in lower control loops must be available for manipulation (no hardware issues, missing signal connection etc.). In practice, these issues are expected, and control specifications must change in real-time concerning changing dynamics caused by such disturbances. Third-generation MPC algorithms address these problems by dynamically dropping MV and related sub-process CV from the control objective and handling them as disturbance variables.

The third innovation is the mechanism to remove the ill-conditioned processes. At any control execution, controlling the outputs may require excessive input movements. This problem can be exemplified if two outputs respond similarly to available inputs. It is important to realize this problem as a feature of the process. Control algorithms attempting to control such an ill-conditioned process will end up having a large number of variable manipulation. This control problem is usually specified during the identification stage; however, it is challenging to specify such a problem on larger systems with multiple sub-processes. Therefore, third-generation MPC is designed to have the capability to examine ill-conditioning sub-processes in the model at each control execution. In practice, MPC detects ill-conditioned sub-processes and screens out from the formulation before resulting in an undesired control action.

MPC gained a vast application area in the late 1990s; fourth-generation MPC technology was developed according to the market's needs, while the acquisition of MPC providers formed some new MPC solutions. Honeywell merged with Profimatics, Inc. and constituted Honeywell Hi-Spec Solutions. RMPC algorithm offered by Honeywell was merged with the Profimatics PCT controller to generate their current RMPCT product. Aspen Technology Inc purchased Setpoint Inc. and DMC Corporation. The SMCA and DMC technologies were merged to develop AspenTechs current product, DMC-plus technology. With technological advancements in computer technologies, fourth-generation MPC can execute complex formulation, and non-linearities can be identified. Contemporary developments in MPC technology involve:

- Integration of MPC into Discrete Control Systems (DCS).
- Data-driven concept and introduction of modern identification methods for multivariable state-space models; subspace, predictive error methods.
- Introduction of Nonlinear MPC to market by Dot Products NOVA-NLC. Practical implementation of MPC for highly nonlinear continuous stirred tank reactor had shown successful results [18].
- Combinations and integrations of modern control approaches into MPC. The use of a neural network model with nonlinear MPC is proposed in [19]. A combination with the fuzzy logic controller is stated in [12]. Adaptive MPC is proposed in [20]. Implementation of Koopman theory in model-based control introduced in [21]

Since MPC implementation can vary for each application, there is no agreement about deciding the best MPC algorithm. Yet, there is a conclusion on the existence of empty rooms to develop new MPC algorithms. MPC showed promising results for the optimal control approach, and pushing the available MPC technology to new applications could yield novel results.

Choosing and implementing an MPC method for a given application is an engineering challenge. One must decide the necessary variables used in control, the proper identification methods, and test models. The basic design should be based on the answers to the following questions: How to design the hierarchical control model for an entire plant? Is the identified model accurate enough to estimate the expected states of the system? What are the hard and soft constraints? How to tune the control parameters for optimal set-point tracking? How to determine the ill conditions that may decrease the performance?

2.2 MPC applications in Relevant Boiler Systems

Studies focusing on the implementation of MPC in Boiler systems are available in the literature. The implementation of MPC may vary with respect to the design approach of the proposed control algorithm. In this section, Boiler Control studies implementing various modern control approaches to MPC are briefly explained.

In [22], control of a Boiler-Turbine unit with a closed-loop MPC algorithm is proposed to improve safety and economic performances. The control of the system, designed as a hierarchical model, supervisory level MPC as the reference governor optimizes the set-points to low-level interconnected PI controllers. The manipulated variables of the boiler system are fuel valve, feedwater valve, and steam valve. The controlled variables are the liquid level in the boiler, the steam pressure, and the generated power. Three interconnected PI controllers individually control the manipulated variables. The task of the MPC reference governor is to provide optimal set-points to the PI controllers concerning system constraints, maximum valve position, and valve slew rate. The MPC approach is executed to predict future dynamics; therefore, the tracking reaction can be optimized. The identification of the system is made by mathematical modelling as set of three differential equations. Kalman filter is used for state estimation and disturbance modelling. The implementation of MPC as reference governor (MPC-RG) showed a favorable result. The performance characteristics of the MPC-RG are also compared with the PI control algorithm in the case studies. The authors concluded that the proposed idea provided safe and economic results over conventional PI control.

In [23], an economic model predictive control is proposed. The authors state that the tracking performance of MPC should not be the only concern of the control and optimization. It is depicted those traditional processes that have accurate dynamic tracking may neglect the optimum economical way. The supervisory MPC in hierarchical control systems may reach the steady-state optimum instead of reaching the global economic optimum. The proposed economic model predictive control directly utilizes the economic index of the boiler-turbine system as the cost function and manipulates the tracking action with respect to the optimal economic result. The system uses the Sontag controller and Lyapunov-based cost function. The optimization problem is solved with online Laguerre functions. The results indicate that the proposed economical MPC can adopt a global optimum routine therefore, it can provide an economic benefit to thermal power plants using traditional MPC.

A hierarchical control system structure with Takagi-Sugeno fuzzy model is proposed to achieve optimal control in a boiler-turbine system [24]. Similar to [22], this model also uses a reference governor, however, the reference governor is proposed as a steady-state target calculator in model predictive controller extended to a fuzzy disturbance model. With this implementation, the TS-fuzzy model can deal with the non-linearities in cost function caused by unknown disturbances and modelling mismatches. A stable model predictive controller is developed to the lower layer of the hierarchy to track the optimized set-points calculated by the upper layer. This algorithm was conducted on simulations and the authors stated that implementation of such complex control approach requires accurate analytical modelling and reliable alternative in case of computational failure. It is also depicted that Data-driven identification is an option for obtaining the model of such a complex system, and PI controllers must be available in the lower layer system for the sake of reliability.

A nonlinear model predictive control is designed for boiler systems [18]. The predictive controller is built with a recurrent neural network acting as a one-step ahead predictor. An optimization problem is derived from the neural network predictor. Two methods of neural network identification methods are used: NNARX (Neural Network AutoRegressive with eXogonous Input), NNOE (Neural Network Output Error). The author stated that the NNOE model provides better prediction quality. The study case of the research is to verification of fault-tolerant control of MPC. Faulty scenarios are described as follows; leakage from boiler, outflow choking, change in internal pipe diameter, leakage from pipe, level transducer failure, positioner fault, valve head or servo fault, pump productivity reduction. The author obtained successful results and stated that the proposed predictive control possesses good fault-tolerant features. It is also stated that MPC can hide faults from being observed; therefore fault detection block is needed for observing any unusuality.

Off-set Free Fuzzy MPC based on genetic algorithm is introduced for a boiler-turbine system in [25]. Takagi-Sugeno fuzzy is implemented to the system generate the approximate behavior of the boiler-turbine system. A genetic algorithm is used to solve the constrained MPC problem. This approach provides good results for nonlinear systems; however, it requires complex analytical modelling and significant amount of data. This work proposes a novel approach to cost function calculations by using genetic algorithms.

Authors stated in [26] that economic-optimal control should be a concern while designing a controller. They propose Hierarchical Model Predictive Control (HMPC) that regulates plant-wise economic optimization while computing optimal set-point at the same time. It is also stated that such kind of MPC application is computationally extensive and can yield nonlinear or non-convex cost function problems. To model the non-linearities, the system's changing dynamics are computed with a fuzzy model representing the input-output relations. Thereby, the MPC system facilitates the quadratic function, including future state, calculated precisely. This control approach is implemented on a steam-boiler generator. The proposed idea is compared with commercial General Predictive Controller (GPC) and PID. The comparison test included step off-set response and dynamic set-point response with different values. According to the results, the proposed HMPC has shown better results (good tracking, fast and robust responses), and the design purpose of the controller is satisfied. The authors conclude that accurate modelling of the controlled system is vital to obtain successful MPC results; however, it is challenging, and mistakes in modelling can cause undesired outcomes.

State-Space-based MPC is proposed for controlling nonlinear MIMO boiler systems in [27]. An online successive model linearization method is designed to obtain efficient cost function and set-point optimization, and this algorithm is named MPC with successive linearization. Kalman Filter handles the estimation of the state matrix. The authors stated that nonlinear control approaches in real-time systems are practically challenging and computationally heavy. This is the main challenge of nonlinear MPC; thus, MPC systems still employ less accurate linear models. The advantage of using online system optimization is that linearization can be made on the current operating point. Therefore, MPC with successive linearization can obtain good control accuracy while not violating constraints. The authors compared their successive linearization algorithms with computationally heavier nonlinear MPC algorithms. It is stated that successive linearization received accurate results similar to the nonlinear approach. This study details the linearization method in MPC systems and reflects the practicality of the nonlinear MPC algorithm.

Lastly, unlike to previous MPC on boiler-turbine control papers, this study reflects the importance of data-driven identification [28]. Authors propose Performance-Oriented Model Learning for Data-Driven MPC design. The idea is to provide the best prediction model to obtain the best closed-loop performance instead of using an adaptive or static robust controller. Data-driven identification is applied by realizing Bayesian optimization. The authors stated that unlike the traditional idea of identification of obtaining the highest input-output fit, the proposed method calculates an identification model yielding the best closed-loop performance for MPC. Additionally, this method is implemented on hierarchical type MPC.

2.3 Conclusion

Literature Review is made according to provide essential ideas behind the cause of MPC development. Automation Surveys focusing on MPC are analyzed and recent studies related to MPC applications in boiler systems are studied. Previous studies reflect the variety of methods to design each essential aspect of MPC. Studies certify the superiority of MPC over PID in complex multivariable systems while depicting the availability of multiple MPC algorithms in the industry. Commercial MPC applications are generated from IDCOM and DMC algorithms; therefore, commercial MPCs suffer from limitations inherited from such algorithms. According to literature review in chapter 2.1, the significant limitations of modern MPC algorithms are:

- Usage of Least-squares type identification. This type of identification model is optimal for linear systems. Studies are reflected in section 2.2; using fuzzy control or successive linearization to model non-linearities can be a solution. However, the implementation requires extensive and complex modelling analysis that might yield an additional cost to plant identification. Moreover, they still provide estimates of model uncertainties.

- Lack of analytically accurate validation methods to verify the identified model is accurate enough to use in MPC. This is a critical factor to prevent signal deterioration and obtain a reliable control system.

- There is a reported deficiency of systematic approaches for building closed-loop identification models. The main reason is the difference in the plant models. A survey about industrial identification by Gevers [29] proposes an identification technique called control-oriented iterative identification and the implementation examples in the industry. This identification method asserts a systematic identification approach to the closed-loop processes that could identify a nominal model from data and optimize from the online model updates. This way of identification can be viewed as an indirect adaptive controller but with an advantage of time-scale separation for analysis for performance-oriented MPC.

- The industry lacks a definition of safety criteria for given Data-Driven identification methods. The Data-Driven models are favorable to enable online identification, generating favorable reliability for long-term utilization of MPC. However, the plant's safety must be ensured while collecting informative identification data meeting the demand of industrial practice [30]. The study by Rivera explains how to accomplish plant-friendly identification addressing the regarding metrics to be monitored.

2.4 The Objective of the Thesis

One significant technological advancement of Industry 4.0 is the utilization of Big Data; control engineers now obtain sufficient information in an industrial process, and information can be used to generate Data-Driven MPC controllers. The data-driven method provides a systematic approach to the identification stage of MPC. The system model can be generated with a 'black-box' test followed by identification. Additionally, this method enables the implementation of modern identification methods such as Predictive Error Method (PEM) or Multivariable Output-Error State Space (MOESP) [31].

The purpose of the thesis is to validate the applicability of the proposed MPC Software while introducing a way of designing an external MPC. The thesis provides studies on identifying a multivariable system by realizing Data-Driven methods and implementations of model-based controllers to utilize a MIMO control. Therefore, the studies are shaped to:

- Obtain a systemic approach for multivariable system identification experiment, which can also be suitable for online identification to obtain an optimal model for changing configuration.

- Realize Predictive error minimalization and Subspace Identification methods and comparing their fitting.

- Generate a robust Data-Driven MPC that can be compared to the external MPC designed in the MPC Software.

3 . An Overview to Model Predictive Control

This chapter provides an overview of critical concepts in model predictive control and briefly describes the algorithm for predictions with the discrete state-space model. The algorithm of MPC is detailed as described in [2], [32], [33].

3.1 Principles of Model Predictive Control

Model Predictive Control is an advanced control method that considers the future behavior of a system over a prediction horizon. Compared to PID, where the control action is made concerning current errors and expected closed-loop dynamics, the MPC algorithm explicitly computes the possible future inputs based on future implications of current and predicted states of the system. Therefore, the MPC algorithm provides observability for the system's future states, performing restrictions to proposed input trajectories for avoiding possible over-reactions that might yield low efficient control performance. This means that the predictive controller continually receives information within a control horizon and uses this information to update control actions over a control horizon. As the new information becomes available at each successive sample, the MPC algorithm automatically modifies control actions accordingly. This process merges under the name of Receding Horizon, constituting the essence of a predictive control algorithm.

In order to predict the future behavior of a process, the control actions must be made according to the actual model of the system. The performance of model predictive control depends on the accuracy of the process model; therefore, identification has significant importance in designing MPC. MPC algorithm is computationally heavy, noting that a quadratic cost function is computed within each sampling interval. To reduce the computational burden, one can use linearization when identifying a model. There are examples of non-linear MPC implementations in[18], [26], [27], [34]–[38]; they will be discussed in discussion chapter.

In practice, MPC algorithms use linear models to facilitate a linear prediction for future inputs (manipulated variable) choices. Additionally, it is also important to mention there are terms so-called 'fit for purpose' and 'identification for control' that are detailed in[29], [30]. These terms highlight the purpose of identification is to obtain a model, providing accurate enough predictions. The implementations of these identification approaches are discussed in chapter 3.

Online constraint handling is the prominent ability of MPC. An optimum control action can be made by realizing constraints on the control signals. These constraints can be defined according to the physical capabilities of the controlled system. The most common constraints are in the form of saturation, valves with finite ranges, PWM signals, flow rates limited by the diameter of a pipe, or even slew rates. MPC predicts future manipulated variables satisfying the defined constraints. Hence MPC provides a natural control even for the complex systems since the system constraints are explicitly stated as a part of the problem formulation.

Tuning is a critical aspect in the designing phase of controllers. In MPC, the tuning serves as a tool to define the optimal control actions according to the financial or operational viewpoint specifications. This means that by tuning MPC, one can design an intelligent controller that considers the operating cost of each control action. Tuning is achieved by introducing a weight matrix in the cost function. The weight matrix emphasizes the relative weight of each parameter in the cost function; thus, the controller prioritizes the regulation of parameters. Using these means, tuning the MPC is a supervisory step to obtain a controller that drives the output to the desired points with convenient inputs, achieving defined performance metrics.

MPC algorithms are formulated for finite impulse response (FIR), finite step response (FSP), and state-space models. In the industrial implementation of MPC, FIR models are more common due to more accessible interpretation of the system with process step response, and they have low sensitivity to measurement noise. However, as [33] states, FIR identification requires serious investigation effort to obtain the most significant data to build the system. On the other hand, MPC with State-Space models extends the flexibility of state-space representation of the model developed by modern black-box identification techniques such as subspace and predictive error minimalization. The central selling point of state-space models is their ease of formulation for multivariable systems. Moreover, state-space benefits from theoretical results to obtain observers to ensure stability and flexibility to be analyzed within control laws.

3.2 The Algorithm of Model Predictive Control

3.2.1 Obtaining Augmented Model

MPC algorithm with the augmented discrete state-space model is detailed. The discrete state space represents the one step ahead states of the system with current state variables described as:

$$\begin{gathered} x(k+1) = A.x(k) + B.u(k), \\ y(k) = C.x(k) + D.u(k) \end{gathered} \qquad 3.1$$

Where *u(k)* denotes the vector of manipulated variables or input, *y(k)* is the process output, and *x(t)* is the state variable vector. Note that in equation 3.2 current state and input directly affect the output. However, due to the receding horizon principle of predictive control, where current information is used to obtain input trajectories, the *D* matrix is assumed to be 0.

The difference between the next state and current state as per 3.1 is formulated as:

$$x(k+1) - x(k) = A.\big(x(k) - x(k-1)\big) + B.(u(k) - u(k-1)). \qquad 3.2$$

The differences of the state variables are denoted by variables:

$$\begin{gathered} \Delta x(k+1) = x(k+1) - x(k); \\ \Delta x(k) = x(k) - x(k-1), \end{gathered}$$

And the difference of manipulated variables is denoted as:

$$\Delta u(k) = u(k) - u(k-1).$$

Δu(k) and *Δx(k)* are the increments of *u(k)* and *x(k).* By substituting the increments into 3.1 yields:

$$\Delta x(k+1) = A.\Delta x(k) + B.\Delta u(k). \qquad 3.3$$

Increments of the output matrix is obtained as:

$$\begin{aligned} y(k+1) - y(k) = C.\big(x(k+1) - x(k)\big) &= C.\Delta x(k+1) \\ &= CA.\Delta x(k) + CB.\Delta u(k). \end{aligned} \qquad 3.4$$

To generate the augmented state-space representation for equations 3.4 and 3.5 new state vector is defined as:

$$x_a(k) = [\ \Delta x(k)^T \ \ y(k)^T\ ]^T \qquad 3.5$$

Putting 3.4, 3.5, 3.6 together yields the augmented state-space model:

$$\overbrace{\begin{bmatrix} \Delta x(k+1) \\ y(k+1) \end{bmatrix}}^{x_a(k+1)} = \overbrace{\begin{bmatrix} A & 0 \\ CA & 1 \end{bmatrix}}^{A} . \overbrace{\begin{bmatrix} \Delta x(k) \\ y(k) \end{bmatrix}}^{x_a(k)} + \overbrace{\begin{bmatrix} B \\ CB \end{bmatrix}}^{B} . \Delta u(k),$$

$$y(k) = \overbrace{\begin{bmatrix} 0 & 1 \end{bmatrix}}^{C} . \begin{bmatrix} \Delta x(k) \\ y(k) \end{bmatrix}. \qquad 3.6$$

A, B, C matrices indicated with curly brackets denote state-space matrices of the augmented model in 3.6, and they are used in the following calculations.

3.2.2 Prediction of States and Future Outputs

The inherent structure of discrete state space is favorable to generate prediction control algorithms as future states can be computed with current state information. To examine this, the future control trajectory is defined by:

$$\Delta u(k), \Delta u(k+1), \ldots, \Delta u(k+Nc-1),$$

As Nc represents the parameter of the control horizon, the number that indicates future control actions. The future states are defined by:

$$x(k+1), x(k+2), \ldots, x(k+Np)$$

Np is the prediction horizon; the parameter depicts the number of optimized future states. The calculation of successive future states is formulated as follows,

$$
\begin{aligned}
x(k+1) &= Ax(k) + B\Delta u(k) \\
x(k+2) &= Ax(k+1) + B\Delta u(k+1) \\
&= A^2x(k) + AB\Delta u(k) + B\Delta u(k+1) \\
&\ \ \vdots \\
x(k+Np) &= A^{Np}\,x(k) + A^{Np-1}B\Delta u(k) + A^{Np-2}B\Delta u(k+1) + \ldots + A^{Np-Nc}B\Delta u(k \\
&\quad + Nc - 1).
\end{aligned}
\tag{3.7}
$$

From predicted state variables in 3.7, future output values are obtained,

$$
\begin{aligned}
y(k+1) &= CAx(k) + CB\Delta u(k) \\
y(k+2) &= CA^2x(k) + CAB\Delta u(k) + CB\Delta u(k+1) \\
y(k+3) &= CA^3x(k) + CA^2B\Delta u(k) + CAB\Delta u(k+1) + CB\Delta u(k+2) \\
&\ \ \vdots \\
y(k+Np) &= CA^{Np}\,x(k) + CA^{Np-1}B\Delta u(k) + CA^{Np-2}B\Delta u(k+1) \\
&\qquad + \ldots + CA^{Np-Nc}B\Delta u(k+Nc-1).
\end{aligned}
\tag{3.8}
$$

To relate equations 3.7 and 3.8 together, Y and ΔU vectors are defined. Y is the future outputs vector based on the state predictions bounded by the prediction horizon, and ΔU is the future input movement bounded by the control horizon.

$$Y = [\,y(k+1), y(k+2), y(k+3) \ldots y(k+Np)]^T$$

$$\Delta U = [\Delta u(k), \Delta u(k+1), \Delta u(k+2) \ldots \Delta u(k+Nc-1)]^T$$

$$Y = F\,x(k) + \Phi \Delta U \tag{3.9}$$

$$
F = \begin{bmatrix} CA \\ CA^2 \\ CA^3 \\ \vdots \\ CA^{Np} \end{bmatrix}
\quad
\Phi = \begin{bmatrix}
CB & 0 & 0 & \ldots & 0 \\
CAB & CB & 0 & \ldots & 0 \\
CA^2B & CAB & CB & \ldots & 0 \\
\vdots & \vdots & \vdots & \ldots & \vdots \\
CA^{Np-1}B & CA^{Np-2}B & CA^{Np-3}B & \ldots & CA^{Np-Nc}B
\end{bmatrix}
\tag{3.10}
$$

The objective of MPC is to obtain the best control parameter vector, ΔU, with an equation, is formulated as a squared error function between set-point references and predicted outputs in 3.9. This error function is the cost function that is minimized in every iteration of MPC. To extend the formula, Rs is denoted as the $Np \times 1$ set-point vector, $\bar{R}$ is the tuned cost matrix of steady-state error, and θ depicts the $Nc \times Nc$

diagonal weight matrix that is used to tune for desired closed-loop performance. Therefore, the cost function is defined as:

$$J = (Rs - Y)^T \bar{R}(Rs - Y) + \Delta U^T \theta \Delta U \quad 3.11$$

Substituting 3.9 into 3.11 yields:

$$J = (Rs - F\,x(k))^T (Rs - F\,x(k)) - 2\Delta U^T \Phi^T (Rs - F\,x(k)) + \Delta U^T (\Phi^T \Phi + \theta)\Delta U \quad 3.12$$

To find the optimal control parameters following condition is derived, the derivative of the cost function with respect to ΔU is zero.

$$\frac{\delta J}{\delta \Delta U} = -2\Phi^T (Rs - F\,x(k)) + 2(\Phi^T \Phi + \theta)\Delta U = 0 \quad 3.13$$

The optimum control parameters are obtained with the following formula:

$$\Delta U = (\Phi^T \Phi + \bar{R})^{-1} \Phi^T (Rs - F\,x(k)) \quad 3.14$$

When the next sample period k+1 arrives, more recent measurements are pushed into 3.14 to obtain the new sequence of the control signal. This procedure is repeated in real-time as per the receding horizon principle of MPC.

3.2.3 Constraint Handling

The importance of constraint handling is visualized in the following figure 2.1. The scenario is to set the output to the reference by MPC without constraints and MPC with constraints. The system without constraints (MPC1) reaches the set-point significantly faster than the system with constraints (MPC2). The input values of MPC1 are not limited, whereas MPC2 limits the input values by 1. For a scenario where "1" is the maximum physical capability of the input, MPC1 providing input values reaching "5" will become an ill-designed controller.

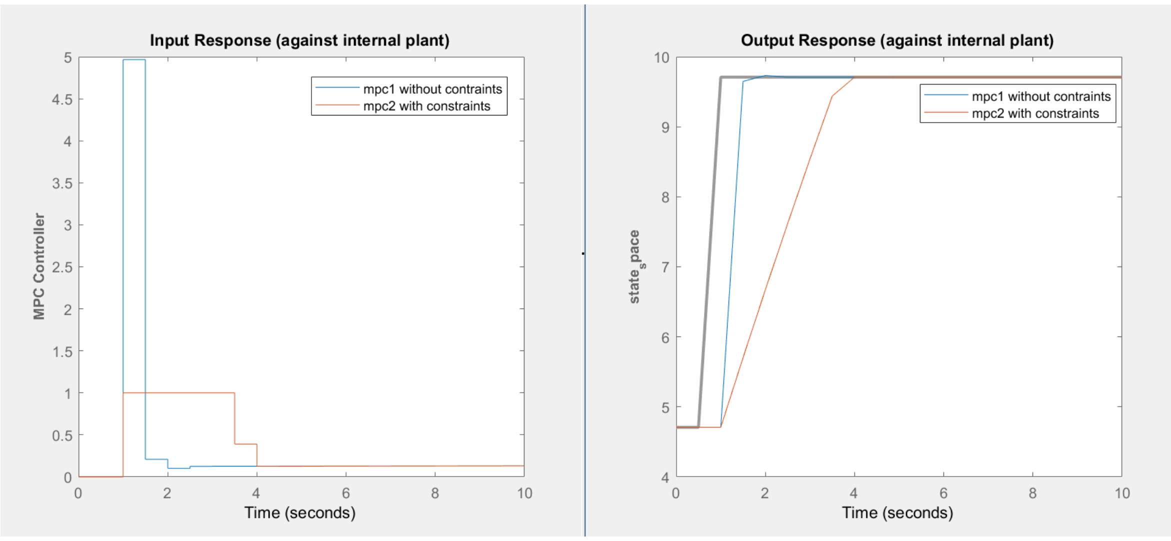


Figure 3.1 Results of MPC with Constraints (Orange) and Without Constraints (Blue)

It is possible to set constraints on input increments ΔU, input value U and output Y:

$$\Delta U_{min} \leq \Delta U(k) \leq \Delta U_{max}$$
$$U_{min} \leq U(k-1) + E_I \Delta U\,(k) \leq U_{max} \qquad 3.15$$
$$Y_{min} \leq F\,x(k) + \Phi\Delta U \leq Y_{max}$$

Please note that equation 2.15 is generated by substituting $U(k-1) + E_I \Delta U\,(k)$ Into *U(k)* and $F\,x(k) + \Phi\Delta U$ into Y. Combined inequality has the following form:

$$E_I = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ 1 & 1 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} I \\ -I \\ E_I \\ -E_I \\ \Phi \\ -\Phi \end{bmatrix} \Delta U\,(k) \leq \begin{bmatrix} \Delta U_{max} \\ -\Delta U_{min} \\ U_{max} - U(k-1) \\ -U_{min} + U(k-1) \\ Y_{max} - F\,x(k) \\ -Y_{min} + F\,x(k) \end{bmatrix} \qquad 3.16$$

The inequality problem in 2.16 is solved with the interior point method. The interior-point method is an algorithm that solves convex optimization problems. The algorithm prevents the constraint violation by introducing a barrier function that defines the optimal unconstrained values in a feasible space to the objective function. A basic barrier function algorithm is shown below; the detailed implementation of the interior-point method is found in[2], [7].

$$\min_{x \in R^n} f(x) \quad for\, x \geq 0 \qquad \Rightarrow \qquad \min_{x \in R^n} f(x) \, - \, \mu \sum_{i=1}^{n} \ln{(x_i)}$$

To summarize the MPC algorithm, the performance of the controller is purely related to the mathematical model of the system. Unlike PID controllers, where the robust control is obtained by manipulating the signal amplifications, MPC controller provides the required signals directly to the system by parallelly computing them within each sampling interval. For that reason, the designing of a high-performance MPC is an identification challenge.

4 Identification

4.1 Design of Identification

System Identification is a subject, compromising techniques to learn the linear dynamics of a system based on measured data. System Identification includes statistical and analytical methods to generate a model and decisive experiments that capture how the process behaves. In fact, System Identification possesses an experimental value to obtain the best possible model for control. Therefore, the offset-free Model Predictive Control construction strongly depends on the accurate identification of the process model.

Obtaining the true system with identification yields the most accurate model-based controller theoretically; however, controlling the true system with high dimensional degrees of freedom is computationally heavy and impractical. In this way, one of the main exercises of identification for control is to obtain a simpler model that is accurate enough within an intended application. The idea is to design an identification experiment finding the dominant dynamics for a particular configuration. This approach is denoted as "goal-oriented identification", "fit for purpose identification" and "control-relevant identification" in [29], [30], [33] respectively, note that in thesis it will be generalized as localized identification.

The localized identification aims to obtain a reduced model that the performance achieved by the model-based controller on the true system is as high as possible within desired set-point ranges. This means the identification can be systematically designed with respect to the application. In literature, [39] an identification design called "iterative identification" is known, explicitly formulates the generation of an accurate reduced-order model. The formulation of iterative identification is as follows:

- Implement a localized identification that obtains a nominal model of the unknown system with open-loop identification.
- Generate a model-based controller to achieve closed-loop stability of the generated nominal model and perform closed-loop identification for the required performance.
- Based on closed-loop identification, design a new model-based controller that is optimal to the true system within control ranges.

The first step of iterative identification aims to minimalize the uncertainties of the true system bounded by a data filter. Data-filter defines the set-points that are dependent on to control objective. The obtained result of the first step is a nominal model that is accurate enough to generate a model-based controller. In the second step, the nominal model-based controller is implemented to perform the closed-loop control within the control objective. By identifying the nominal closed-loop system, a model with fewer uncertainties can be derived, which will be used to obtain the high-performance controller in the third step. The third step includes the validation of the previous actions and the tuning of the controller for desired performance metrics.

Iterative Identification provides flexibility to the designer to experiment and shape the identification process. In fact, one can start identifying the system from a black-box model, gradually to obtain a gray-box model that could practically pair up with the white-box model. The feasibility of the first step on the black-box model extends the applicability of the iterative identification over the industrial process if the data log of the plant is available. Each step in the identification process provides valid results for the next step, so that plant safety can be almost guaranteed while experimenting with the process. The resulting model is the simplified model that yields a relatively simple model-based controller. Simplicity in a controller is an essential factor in process control, considering the popularity of PID controllers in industrial utilization. Due to these reasons, the iterative identification will be referred to as designing the identification experiments.

4.2 Prediction Error Identification Method

Prediction Error Minimization (PEM) is considered as the most successful and popular identification method. PEM is a parametric linear system identification method that can capture the essential dynamics favorable for predicting future states and controller design. The prediction error denotes the output difference between the actual system and predicted by the estimated model. The function of PEM minimalizes the prediction error by tuning the parametrized model set.

Let the true system is modeled with input $u(t)$, $G_0(q)$ as a transfer function, $H_0(q)$ denotes the colored noise with respect to white noise $e(t)$:

$$\mathrm{y}(t) = G_0(q).u(t) + H_0(q).e(t) \tag{4.1}$$

Parametrized model is derived in equation 4.2, as θ is the parameter vector and D_θ is the subset of permissible values for θ.

$$M = \{(G(q,\theta),(H(q,\theta)\ \ \theta\ \in D_\theta\} \qquad 4.2$$
$$\hat{y}(t) = \ G(q,\theta).u(t) + H(q,\theta).e(t) \qquad 4.3$$

In the black-box model case, each sample data collected from the identification experiment is the response of the actual system for a given input. Hence, *y(t)* in equation 4.1 is the measured output of the system in time sample t. Subtracting equation 4.1 from 4.3 yields the prediction error for the given input at t.

$$e(t,\theta)\ =\ y(t)\ -\ \hat{y}(t,\theta), \qquad 4.4$$

The measurement noise affecting the prediction error can be filtered by a data-filter *D(q)*; thus, filtered PE is defined as $e^F(t,\theta) = D(q).e(t,\theta)$. The tuning of parameter vector θ is carried out by the least-squares method. Therefore, the least-squares function is defined as:

$$V_N = \frac{1}{N}\sum_{t=1}^{N}[e^F(t,\theta)]^2 \qquad 4.5$$
$$\hat{\theta} = \arg min_\theta\, V_N(\theta)$$

Where N is the number of input-output samples and V_N is the least-squares cost function. The estimated model is obtained as values in the parameter vector $\hat{\theta}$ is substituted into $\{(G(q,\theta)\ (H(q,\theta)\}$ model.

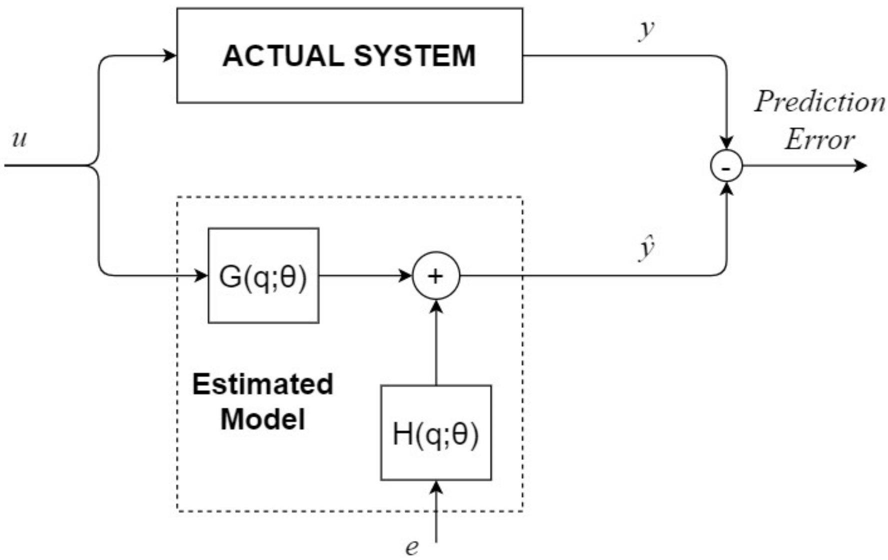


Figure 4.1 Scheme of Prediction Error Minimalization

As the initialization step for PEM identification, parametrization is defined according to the structure of the desired model. To identify a state-space model with PEM, coefficients in state-space matrices are determined as in subspace identification. After generating the initial model with the subspace method, the PEM method is executed to identify the model.

Matlab Identification Toolbox [40] provides state-space identification with subspace-PEM method. The user interface of the toolbox provides selection for identification methods and solutions for the least-squares functions. Data from the identification experiments are pushed into the Identification Toolbox. The PEM method is used to identify the state-space model, and least-squares functions are solved with the Gauss-Newton method.

5 Identification of Multi-tank System and Development of Model Predictive Controller

5.1 An Overview to Multi-tank System

The Multi-tank System (Figure 5.2) comprises three separate tanks fitted with controllable drain valves and manual drainage. The separate tank mounted in the base of the setup acts as a water reservoir for the system. First Tank has a constant cross-section, the second tank has a conical, and the third tank has a spherical cross-section. A variable speed tank pumps the water in the reservoir to the first pump, and the water outflows from the first tank to the second and third tanks due to gravity. The controllable drain valves act as flow resistors, and controlling the valves are used to vary the outflow characteristics. Manual drainages are like controllable drain valves, yet they are operated manually. Manual drainages can increase the systems' overall drainage speed, affecting the system's time constant in control. Each tank is equipped with a level sensor based on hydraulic pressure measurement. The Multi-tank system is operated with an external PC-based controller that communicates with the level sensors, valves, and pump by a dedicated RT-DAC/PCI board. The I/O board is controlled by the real-time windows target driver, which operates in MATLAB/SIMULINK. The driver possesses four PWM inputs; The first input (Pump) controls the pump. The inputs Valve1, Valve2, and Valve3 control the valves of the upper, middle, and lower tanks. There are three analog outputs of the driver: Level1, Level2, Level3, and one digital output: Alert. Each analog output represents the liquid level in the tank displayed in metric units. Alert signal has a safety purpose, indicating whether tank levels in the system are within the safety boundaries or not. The frequency signals are converted into metric units inside the driver block.

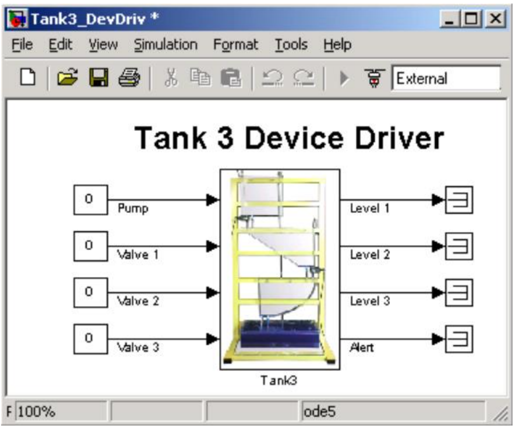


Figure 5.1 Multi-tank Simulink Driver Block

Pump value varies between 0 and 1; 0 means 0% and 1 means 100% Duty-cycle. Due to the height distance between the reservoir tank and the inlet of the first tank, water can be pumped with Pump values greater than 0.4. Valve 1, Valve 2, and Valve 3 are initialized as 0 in the driver shown in figure 5.1; however, the operation value range of the valves are bounded with 0.5 and 1, as 0.5 is the fully closed 1 is the fully-open state. Valve values can vary between the operating ranges, providing variable outflow. The level limit of each tank is 25 centimeters, and if the limit is exceeded, the Alert signal turns off the device.

Table 5.1 Multi-tank Operating Ranges

Input/OUTPUT	Operating Minimum	Operating Maximum
PUMP (%)	0.4	1
Valve1	0.5	1
Valve2	0.5	1
Valve3	0.5	1
Level1 (cm)	0	25
Level2 (cm)	0	25
Level3 (cm)	0	25

The goal of the Multi-tank system is to enable a real-time platform for iterative identification experiments. Based on systems' real-time data obtained from the experiments, the PEM identification method will be applied to obtain state-space models and ultimately design an offset-free model predictive control. The objective of the control is to reach and stabilize the water level in the tanks by an adjustment of the pump capacity rate and valve settings. The multi-parameter architecture of inputs and outputs allows the user to realize the system according to the number of parameters chosen in the desired system to be controlled. Thereby, the system can be realized as single-input single-output, multiple-input single-output, and multi-input multi-output.

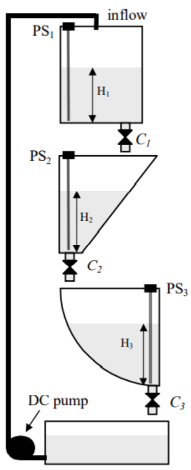


Figure 5.2 The Scheme of Multi-tank System

5.2 Identification Methodology of Multi-tank System

As stated in Chapter 4, Iterative identification experiments consist of 3 steps:

1. Implementation of Localized Identification to the black-box model within the desired working ranges and the development of nominal MPC based on the localized identification.

2. Identification of the closed-loop system, controlled by the nominal MPC.

3. Improving the high-performance MPC, based on the identification results in the second step.

The implementation of each step of iterative identification is studied in SISO, MISO and MIMO models. The results of identification of these models are critical for generating SISO-MPC, MISO-MPC and MIMO-MPC, respectively. Note that, the third step of iterative identification will be analyzed in chapter 6.

Water levels in each tank can reach 25 centimeters, thus the outflow rates with a constant outlet water resistance causing non-linearity. Considering the nonlinear characteristic of the Multi-tank system, linearization in each model is handled by choosing relatively small working ranges. Therefore, for every model, the working range is defined between 4 and 11 centimeters. This means, the localized identification generates the model of the system operating between 4 and 11 centimeters.

Initial challenge of the experiments is to choose an input set, that yields high quality identification results. Providing random inputs with large samples to the black-box models might give abundant range of data, however the quality of the experiments can be negatively affected. Moreover, safety of the model is a critical constraint, therefore initial input sets must provide sufficient information about system dynamics while ensuring model safety. According to safety criteria in identification experiments that are described in [30], one approach to ensure model safety is to keep input transitions low as possible (where u(k-1) ≠u (k)). This approach is uses friendliness index "f" shown in equation 5.1. Consequently, experiments are designed to satisfy the quality of identification and plant friendliness index.

$$f = 100 \times \left(1 - \frac{Number\ of\ input\ transitions}{Number\ input\ samples}\right) \qquad 5.1$$

5.2.1 Identification of Single-Input Single-Output System

To simplify the implementations of identification methods, the initial experiments were conducted on a SISO system. The SISO system is modeled by defining Pump PWM gain as input and the first tank level (Level1) as output. Note that other parameters in the Multi-tank system were not taken into consideration. The model's simplicity enabled practical implementations of the iterative identification experiments, forming the adopted methodology in the identification stage.

Figure 5.3 SISO Model

Experiments were done with two data sets; one is for identification, and the second is for validation. The data set for identification is shown in figure 5.4. The graph indicates the input-output relation of the black-box model. As following the steps of iterative identification, a nominal model is obtained according to the data set. PEM method is applied to the data set by using MATLAB Identification Toolbox. The result of the calculation yields a discrete state-space model. To validate the resulting model, the data set for validation is fed to the discrete state-space model. The method of validation is to compare the known output of the actual model with the output of the identified model from the corresponding input. Note that it is a good practice to use different data sets for the validation step to prevent over-fitting.

Step 1: Identification of Nominal SISO Model

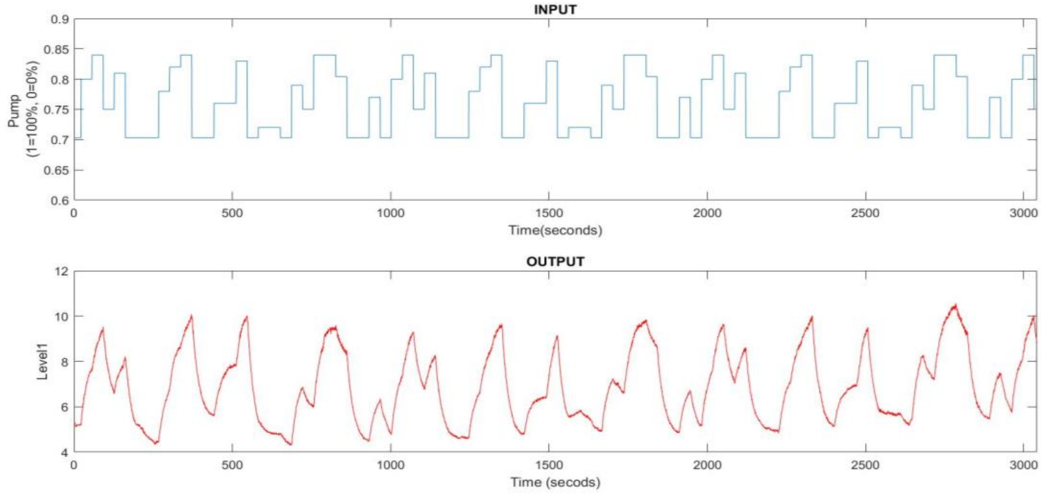


Figure 5.4 Identification Data Set of SISO Model

The nominal model was identified using the data set in figure 5.4. The data set consists of 6082 samples obtained from the system, with 0.5 seconds sampling time. The plant friendliness index was also taken into consideration. Equation 5.1 is used to calculate the "f" index, and it is equal to 98.87%, which is highly optimal. The resulting nominal model is:

$$A = \begin{bmatrix} 0.9807 & -0.0263 \\ 0.0022 & 0.9941 \end{bmatrix} \quad B = \begin{bmatrix} -1.2027 \\ -1.3934 \end{bmatrix}$$

$$C = \begin{bmatrix} -0.3307 & -0.3962 \end{bmatrix} \quad D = 0 \tag{5.2}$$

The fit rate to the validation data is 79.36% which is indicated in figure 5.5. The next important step is to check the observability and controllability of the obtained model. If the state-space model is not observable or controllable, MPC cannot be implemented in the system. The system is observable and controllable if the ranks of both observability and the controllability matrices are equal to the length of the n-by-n state coefficient matrix A. In this case, the length of the A matrix is 2, and as shown in equation 5.3, the ranks of observability and controllability matrices are 2; thus, the model is observable and controllable.

$$Controllability = [B \;\; AB \;\; A^2B \; \dots \; A^{1-n}B] = \begin{bmatrix} -1.2027 & -1.1428 \\ -1.3934 & -1.3877 \end{bmatrix} \rightarrow Rank\ 2$$

$$Observability = \;[C \;\; CA \;\; CA^2 \; \dots \; CA^{1-n}]^T = \begin{bmatrix} -0.3307 & -0.3962 \\ -0.3252 & -0.3852 \end{bmatrix} \rightarrow Rank\ 2 \tag{5.3}$$

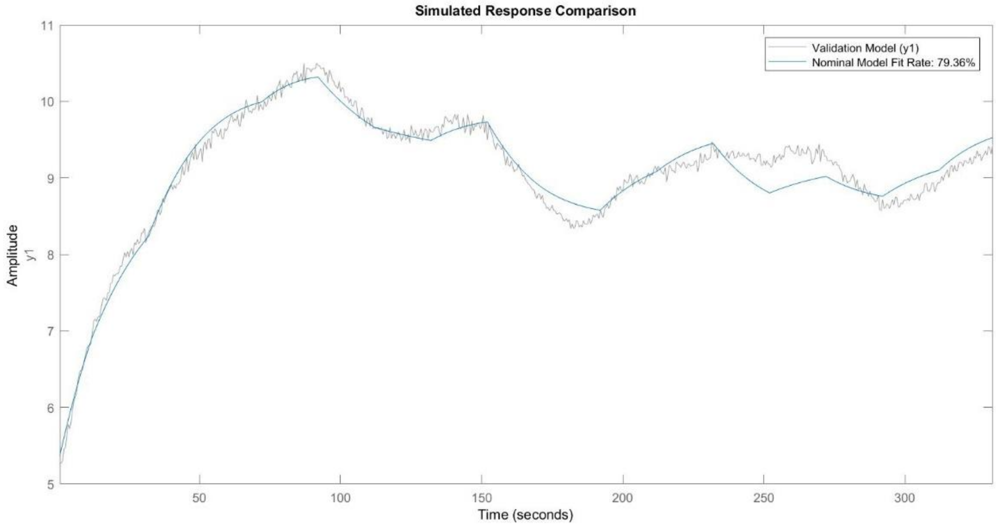


Figure 5.5 Fit Rate to Validation Data

The last step to validate the model is to check the finite impulse response. The aim is to observe how a kicked input is affecting the output. Impulse response serves as a tool to justify the input and output relation. Since the model is a water tank, it is expected to observe an increase in output followed by an exponential decay slope. The impulse response is obtained by the "impulse" command in MATLAB, and the results are shown in figure 5.6. The impulse response reflects that the kick in the input directly affects the output, and due to the outflow, the output level decreases in time. However, at t = 50 seconds, the output level drops below 0 centimeters which is a shortcoming of the nominal model dynamics.

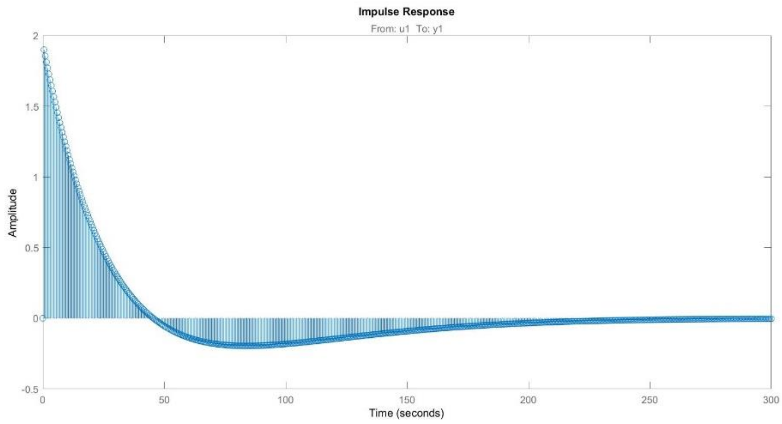


Figure 5.6 Impulse Response of the Nominal Model

Step 2: Closed-Loop Identification of Nominal SISO Model

Nominal MPC is generated from the MATLAB/SIMULINK Model Predictive Controller function block. The controller function block requires the discrete state-space model obtained in step1. After the model-based controller is initialized as illustrated in figure 5.7, controller is designed as follows:

- Prediction Horizon = 12
- Control Horizon = 3
- Control Sampling = 0.5 seconds
- Hard Constraints for Pump input value Max = 1, Min = 0.4
- Constraints for Pump input rate Max =1, Min = -1
- Soft Constraints for Output value Max = 15, Min = 0

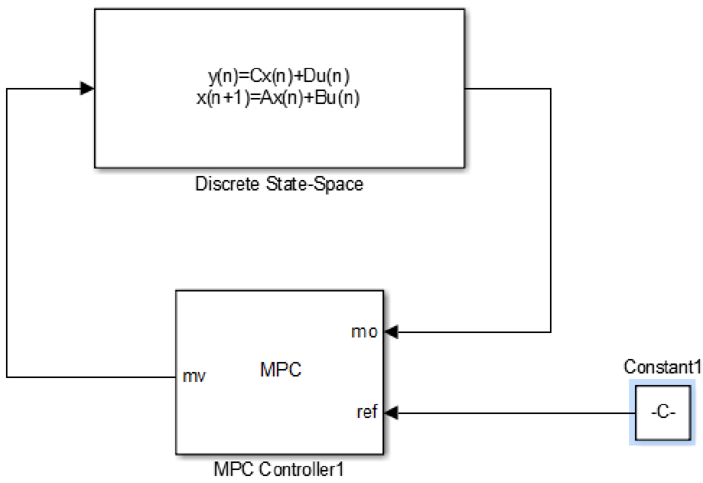


Figure 5.7 Initialization of Nominal SISO MPC

After the design of nominal MPC, the closed-loop system in figure 5.8 is generated, as the nominal MPC manipulates the required pump value regarding the difference between the current level and reference level. Level Reference is the set of reference data for the controller. Closed-loop identification is designed to obtain the most relevant system dynamics within the desired control range. Therefore, Level Reference provides set-point values between 4 and 11 centimeters.

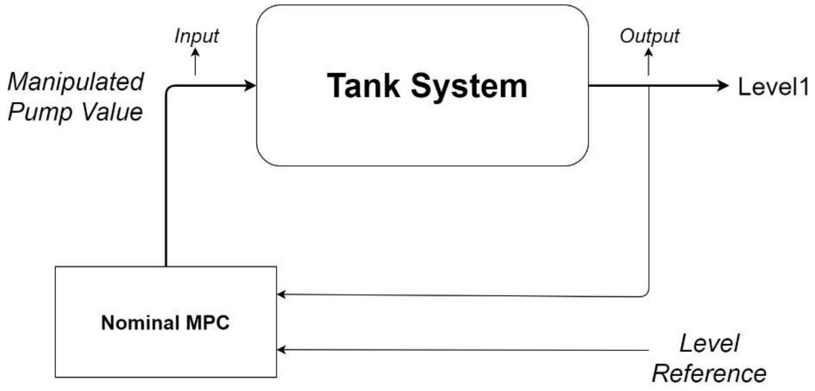


Figure 5.8 Scheme of Closed-Loop System with Nominal SISO MPC

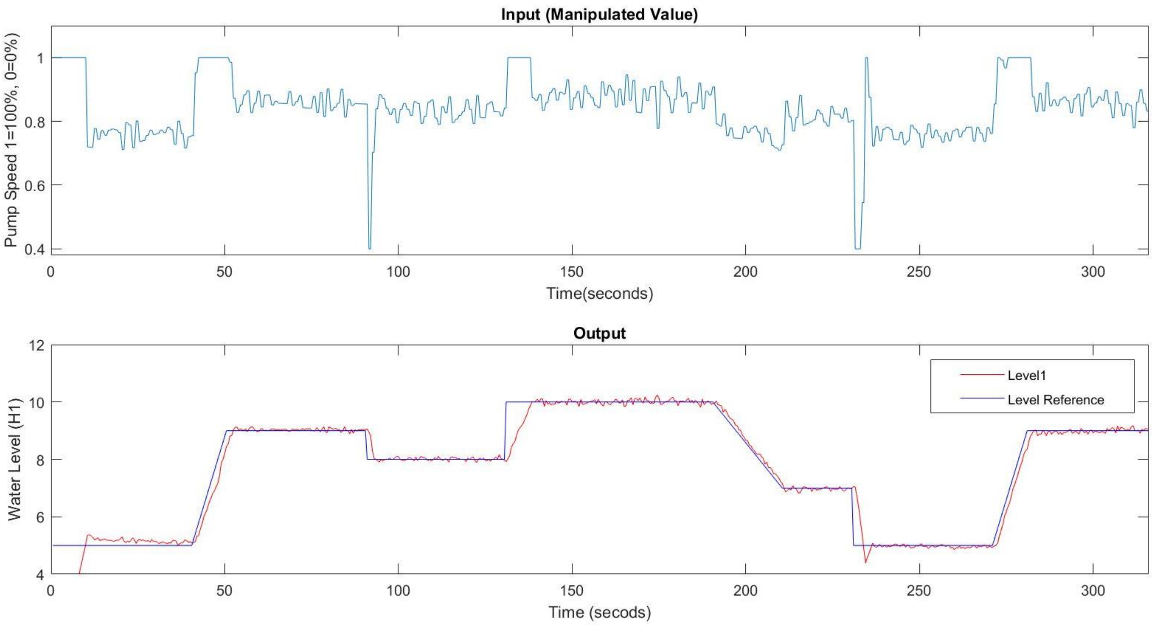


Figure 5.9 Closed-loop Control with Nominal SISO MPC

Figure 5.9 illustrates the result of the control with the nominal MPC. The controller's performance is sufficient to execute the closed-loop identification since input manipulation can track the reference values. The data set obtained from the closed-loop experiment is used to identify the system for the second time. The result of closed-loop identification is:

$$A = \begin{bmatrix} 0.9830 & -0.2282 \\ -0.0005 & 0.9931 \end{bmatrix} \quad B = \begin{bmatrix} 3.2232 \\ 0.0980 \end{bmatrix}$$
$$C = \begin{bmatrix} 0.3094 & 0.4185 \end{bmatrix} \quad D = 0 \tag{5.4}$$

The validation fit of the closed-loop model is 93.67% which highly optimal for off-set free MPC design. The ranks controllability and observability matrices of the closed-loop model are calculated as 2, which makes the model controllable and observable. The impulse response of the closed-loop model is shown in figure 5.10. Compared to nominal model where the output level goes below zero, the closed-loop model achieves more accurate dynamic characteristics.

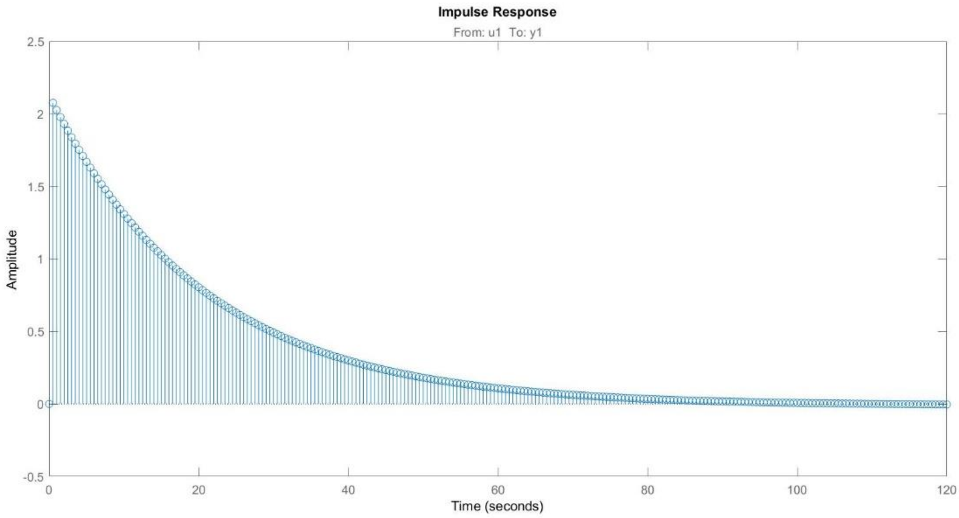


Figure 5.10 Impulse Response of the Closed-Loop SISO Model

The result of the iterative identification yields the SISO model in equation 5.4. The closed-loop model provides significant fitting within the identification ranges, and it is able reflect the true dynamics accurately. Therefore, the closed-loop model will be used to design the high-performance MPC.

5.2.2 Identification of Multi-Input Single-Output System

MISO system is created by adding Valve1 of the first tank as input into the SISO model. MISO model consists of Pump gain and Valve1 as inputs, Level1 as output. Additional to the SISO model, outflow rate of the tank can be controlled with Valve1 signal, this defines the function of level with two input parameters. MISO model provides an insight to identification of multivariable systems. Multivariable systems have more degrees of freedom, in this sense designing the identification experiment requires strategic analysis of the black-box model of a multivariable system. Since the objective of the identification is to build a model-based controller, experiments must reflect the desired control task.

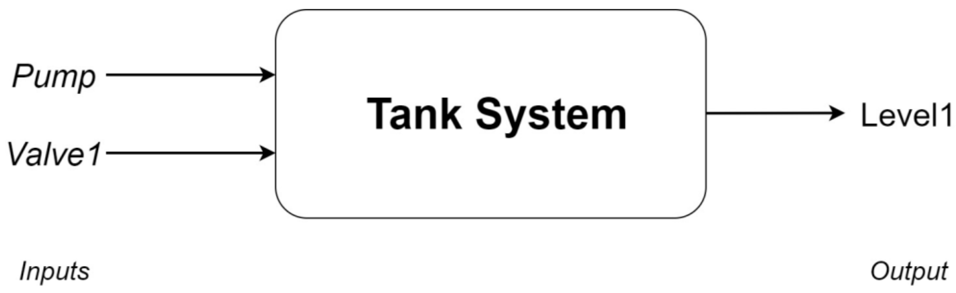


Figure 5.11 MISO Model

From the previous identification experiment the dynamics of the SISO model was known. Therefore, initial task of the MISO identification is to observe the effects of different Valve1 values to the system. The goal is to minimalize the possible non-linearities by identifying the multivariable system in three stages. Each stage corresponds the different values of the Valve1 input. During first stage of identification, Valve1 value is 1 which means the outlet valve is fully opened. In the second stage Valve1 is 0.75, 50% opened. In the third stage Valve1 value is 0.5 meaning the outlet value is fully closed. Each stage was combined to form the identification data set illustrated in figure 5.12.

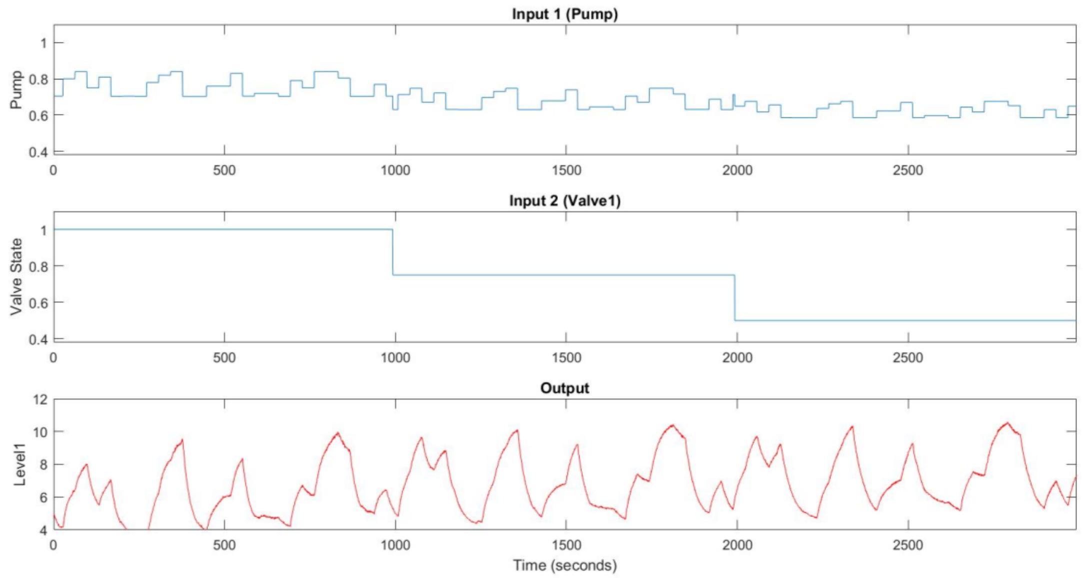


Figure 5.12 Identification Data Set of MISO Model

Stage 1 corresponds to the time instance between 0 and 1000 seconds, Stage 2 is between 1000 and 2000 seconds, Stage 3 is between 2000 and 3000 seconds. Note that the during each stages Pump input keeps the water level between 4 and 11 centimeters. The identification data set consist of 6000 samples and the sampling time is 0.5 seconds. Plant friendliness metric of the experiment was calculated as 98.83%.

The identification is done with PEM method, the resulting MISO discrete-state-space model is:

$$A = \begin{bmatrix} 0.9830 & -0.2282 \\ -0.0005 & 0.9931 \end{bmatrix} \quad B = \begin{bmatrix} 3.2232 \\ 0.0980 \end{bmatrix}$$

$$C = \begin{bmatrix} 0.3094 & 0.4185 \end{bmatrix} \quad D = 0 \tag{5.5}$$

$$U = \begin{bmatrix} Pump & Valve1 \end{bmatrix}$$

The resulting model is observable and controllable with validation fit rate of 77.93%. The impulse response of the model is shown in figure 5.13. The left plot shows the impulse response of the Pump input, and the right plot is the impulse response of the Valve1 input. The kick in Pump input pushes the system to positive values while the Valve1 kick pushes the system to negative values. This means, Pump input has a positive effect on filling the system and Valve1 input has positive effect on draining it.

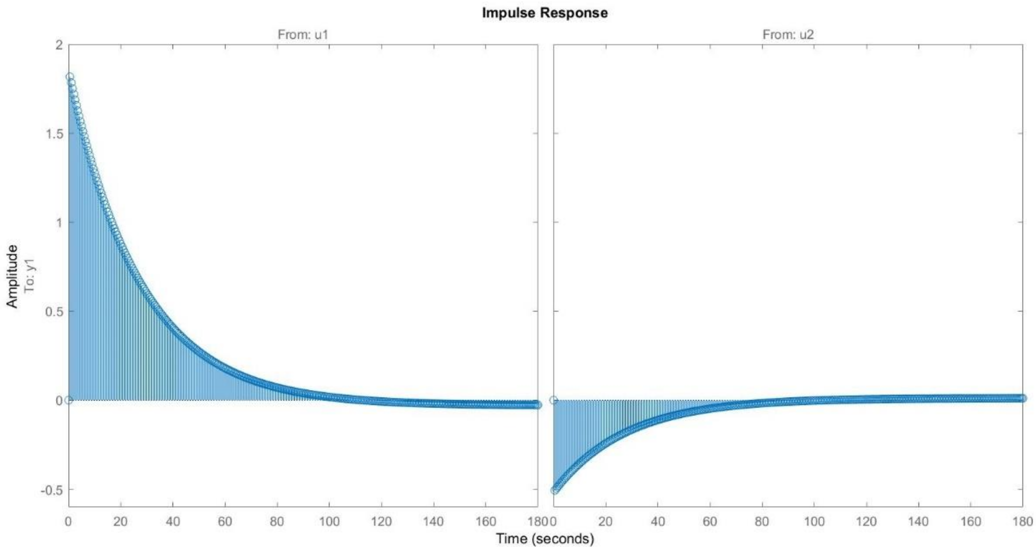


Figure 5.13 Impulse Response of the MISO Model

The identified MISO model is validated to be accurate enough to be implemented in a nominal MPC. After the initialization of MATLAB MPC, the nominal MISO MPC is designed as follows:

- Prediction Horizon = 12
- Control Horizon = 3
- Control Sampling = 0.5 seconds

- Hard Constraints for Pump input value Max = 1, Min = 0.4
- Constraints for Pump input slew rate Max =1, Min = -1
- Hard Constraints for Valve1 input value Max = 1, Min = 0.5
- Constraint for Valve1 input slew rate Max = 0.5, Min = -0.5
- Soft Constraints for Output value Max = 15, Min = 0

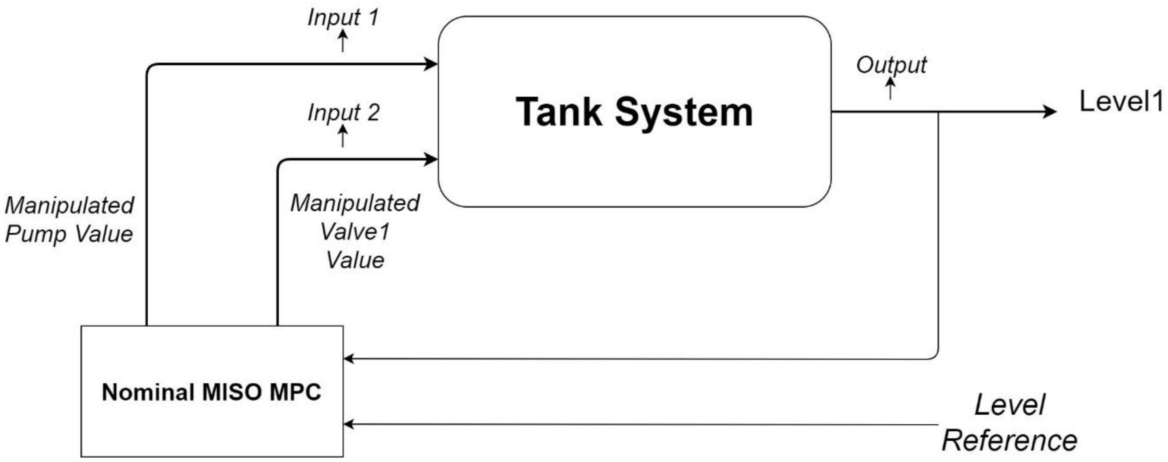


Figure 5.14 Scheme of Closed-Loop System with Nominal MISO MPC

The closed-loop performance of the nominal MISO MPC is illustrated in figure 5.15. The controller can track the reference trajectories by manipulating the Pump and Valve1 inputs simultaneously. Based on the observation, the nominal MISO MPC provides satisfactory closed-loop performance, for this reason second step of iterative identification is skipped for this model. Further analysis of the MISO MPC and the tuning effects of the controller will be discussed in chapter 6.

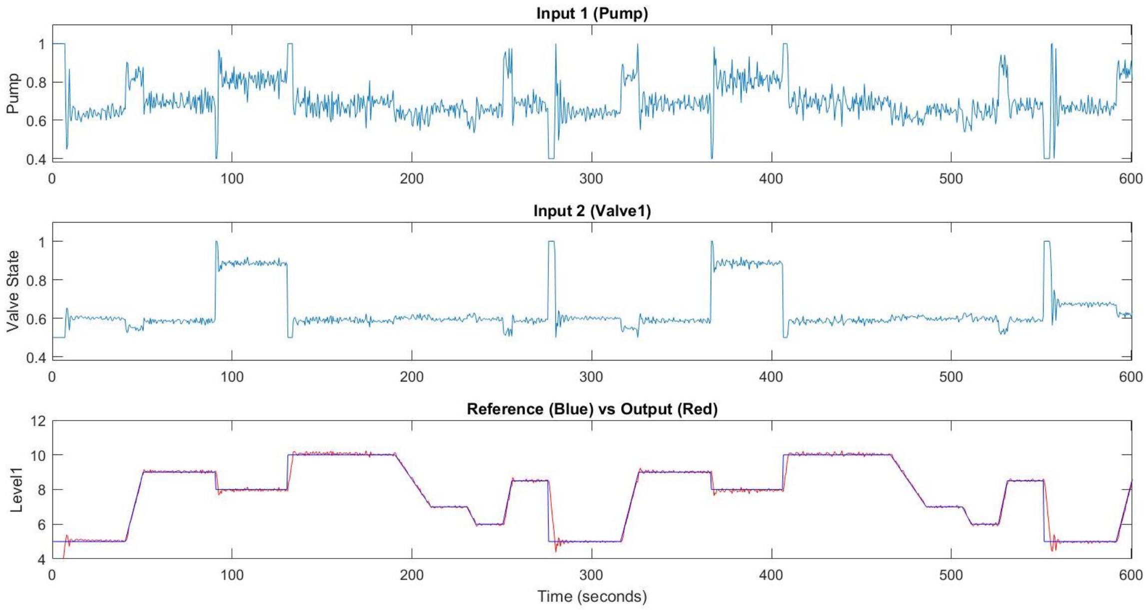


Figure 5.15 Closed-loop Control with Nominal MISO MPC

5.2.3 Identification of Multi-Input Multi-Output Model System

MIMO model is a complex system, having 3 inputs and 2 outputs. Pump gain, Valve1, Valve2 (the outlet valve of tank 2) are the inputs and Level1, Level2 (water level in tank 2) are the outputs. The control objective of the MIMO system is to control the water levels in tank 1 and tank 2 simultaneously, by manipulating the input values of Pump, Valve1 and Valve2. The conical shape of the second tank grants system additional non-linearity. The motivation of designing the MIMO model is to prove the effectualness of MPC over controlling multivariable systems.

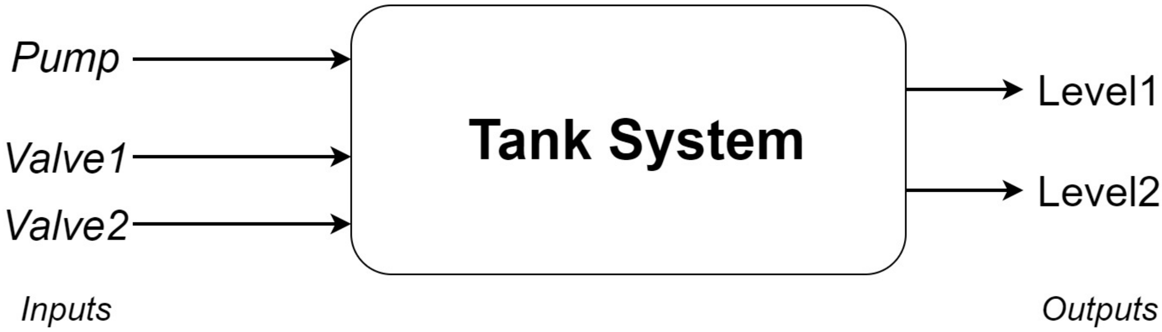


Figure 5.16 MIMO Model

Step1: Identification of the nominal MIMO model

The identification process of the MIMO starts from the black-box identification experiment. As in MISO experiment where the identification process is separated into 3 different stages, MIMO experiment is conducted similarly. Besides, MIMO experiment is divided into 3 stages each having sub-stages. Each sub-stage is executed multiple times within the corresponding stage. Pump input value is adjusted to hold the Level1 within 0 and 10 centimeters, while Valve2 input value tries to keep the Level2 between 0 and 18 centimeters.

Table 5.2 Stage and Sub-Stage Diagram

<table>
<tr><td colspan="3">Stage 1
Valve1 = 1
Interval = 0 – 900 sec</td><td colspan="3">Stage 2
Valve1 = 0.75
Interval = 900 – 1800 sec</td><td colspan="2">Stage 3
Valve1 = 0.5
Interval = 1800 – 2700 sec</td></tr>
<tr><td colspan="3">SUB-Stages</td><td colspan="3">SUB-Stages</td><td colspan="2">SUB-Stages</td></tr>
<tr><td>Valve2 = 1</td><td>Valve2 = 0.75</td><td>Valve2 = 0.5</td><td>Valve2 = 1</td><td>Valve2 = 0.75</td><td>Valve2 = 0.6</td><td>Valve2 = 1</td><td>Valve2 = 0.75</td></tr>
</table>

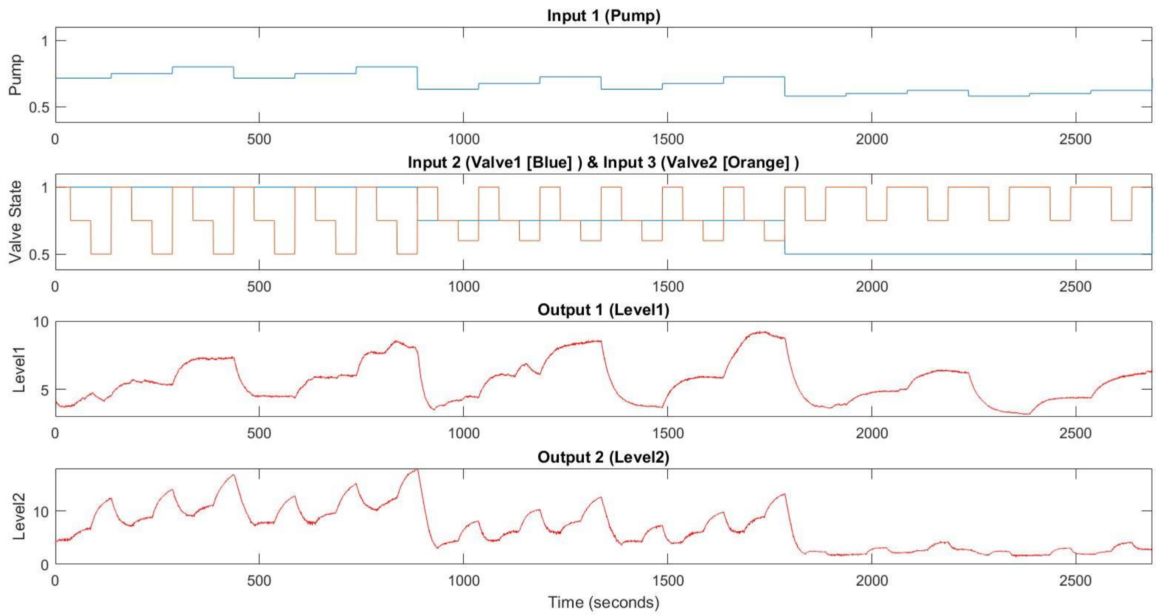


Figure 5.17 Identification Data Set of MIMO System

Identification data set in figure 5.17 consist of 5374 samples with 0.5 seconds of sampling time. After the identification with PEM method, resulting discrete state-space model is:

$$A = \begin{bmatrix} 0 & 0 & 0.9577 \\ 1 & 0 & -2.9149 \\ 0 & 1 & 2.9572 \end{bmatrix} \quad B = 10^3 \times \begin{bmatrix} 0.001 & 1.0054 & -1.5217 \\ 0 & -2.0672 & 3.1299 \\ 0 & 1.0624 & -1.6095 \end{bmatrix}$$

$$C = \begin{bmatrix} 1.1289 & 1.1018 & 1.0753 \\ 0.4379 & 0.4415 & 0.4447 \end{bmatrix} \quad D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{5.6}$$

$$U = \begin{bmatrix} Pump & Valve1 & Valve2 \end{bmatrix} \quad Y = \begin{bmatrix} Level1 \\ Level2 \end{bmatrix}$$

The fit rate of the nominal MIMO model to Level1 signal is 71% and Level2 81.42%. The plant friendliness is calculated as 92.61% which is satisfactory. The model is observable and controllable; therefore, the nominal MIMO is validated to design a nominal MPC for closed-loop identification.

Step2: Closed-Loop Identification of the Nominal MIMO Model

Discrete State-Space model in 5.6 is used to initialize the nominal MIMO MPC. The scheme of the closed-loop system is shown in figure 5.18. The nominal MPC is designed as follows:

- Prediction Horizon = 10
- Control Horizon = 2
- Control Sampling = 0.5 seconds
- Hard Constraints for Pump input value Max = 1, Min = 0.4
- Constraints for Pump input slew rate Max =1, Min = -1
- Hard Constraints for Valve1 input value Max = 1, Min = 0.5
- Constraint for Valve1 input slew rate Max = 0.5, Min = -0.5
- Hard Constraints for Valve2 input value Max = 1, Min = 0.5
- Constraint for Valve2 input slew rate Max = 0.5, Min = -0.5

- Soft Constraints for Output 1 value Max = 15, Min = 0
- Soft Constraints for Output 2 value Max = 20, Min = 0

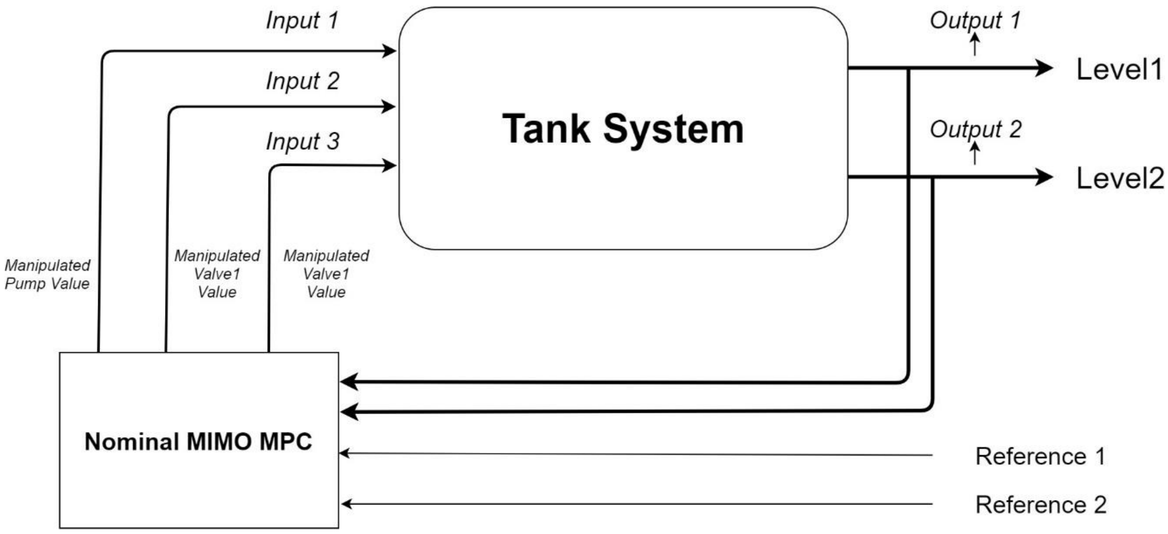


Figure 5.18 Scheme of Closed-Loop System with Nominal MIMO MPC

The result of closed-loop control is plotted in figure 5.19. The controller can track the reference trajectories for Level1, on the other hand there is a consistent off-set error for Level2. This off-set error is caused by the uncertainties in the nominal MIMO model; therefore, the nominal MIMO MPC performs poorly. The data obtained from the model is used to perform the closed-loop identification. Input data obtained the control experiment are shown in figure 5.20. The closed-loop identification data set consist of 1801 samples with 0.5 seconds of sampling interval. The identification of the closed-loop system yields the following model:

$$A = \begin{bmatrix} 0.9665 & -0.0122 & 0.0699 \\ 0.0093 & 0.9888 & 0.0053 \\ 0.0039 & -0.0028 & 0.9911 \end{bmatrix} \quad B = \begin{bmatrix} -1.9556 & 0.8039 & -0.6100 \\ -0.9791 & -0.2322 & 0.6292 \\ 0.0190 & -0.0545 & 0.0404 \end{bmatrix}$$

$$C = \begin{bmatrix} -0.4744 & -0.1758 & 0.0536 \\ 0.1703 & -0.4382 & -0.5695 \end{bmatrix} \quad D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{5.7}$$

$$U = [Pump \quad Valve1 \quad Valve2] \quad Y = \begin{bmatrix} Level1 \\ Level2 \end{bmatrix}$$

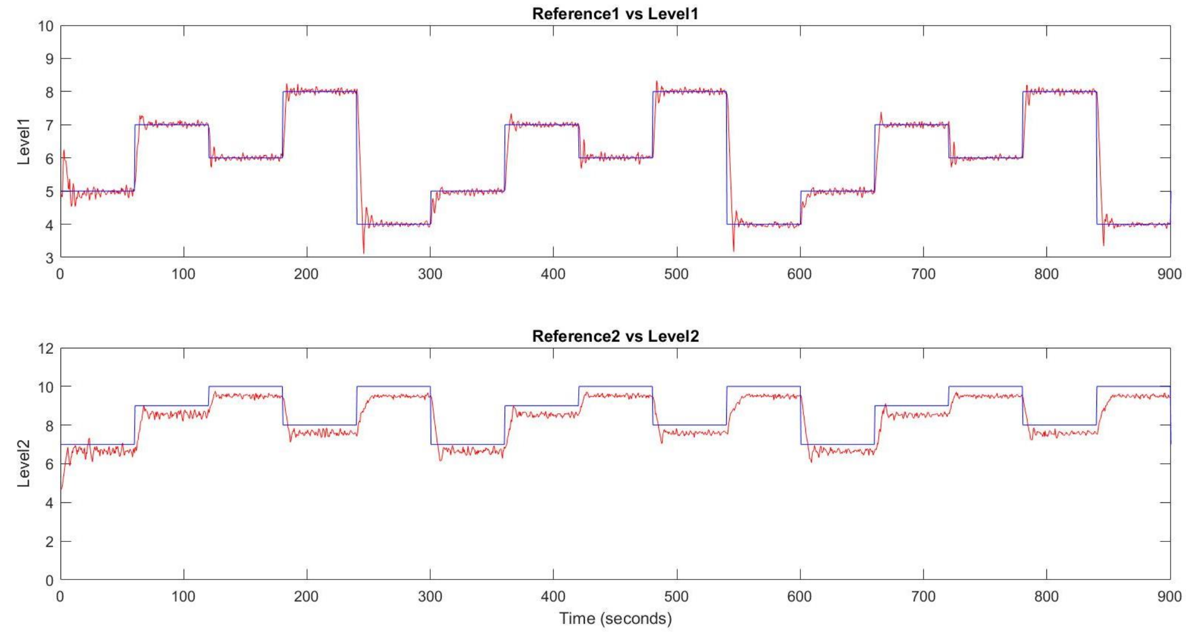


Figure 5.19 Closed-loop Control with Nominal MIMO MPC

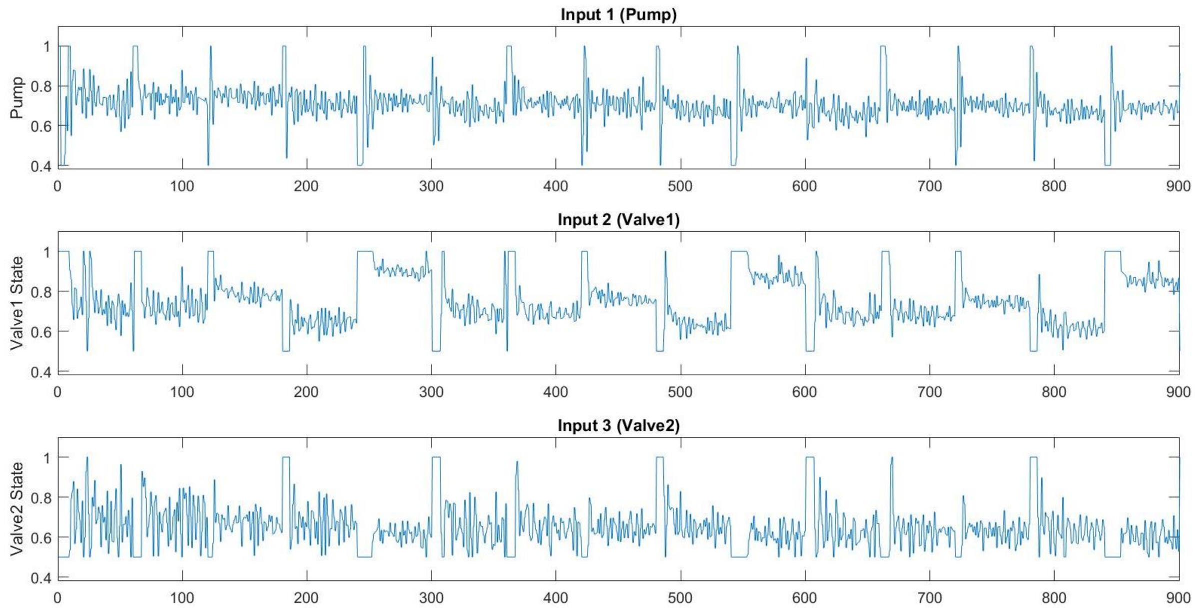


Figure 5.20 Input Data from the Closed-Control with Nominal MIMO MPC

The fit rate of the identified closed-loop MIMO model to the validation date is 79.41% for Level1 and 76.09% for Level2. For validation, the impulse response of the closed-loop MIMO model is analyzed. According to the impulse response, the kick in the Pump signal initially pushes both output signals towards the positive margin. This means, the pump input affects both tanks as it pushes water to the system. Impulse Valve1 signal pushes Level1 output towards negative margin while pushing Level2 output to the positive margin. This explains as the outflow in the Valve1 increases, water level in the first tank decreases and since the water flows to the second tank, water level in the second tank increases. Lastly, the kick the in the Valve2 signal shows no effect to Level1 output but pushes Level2 output to the negative margin. This result shows the significant effect of Valve2 in draining the second tank. Moreover, since the Valve2 is not connected to the first tank, the change in Valve2 signal will not considerably affect Level1.

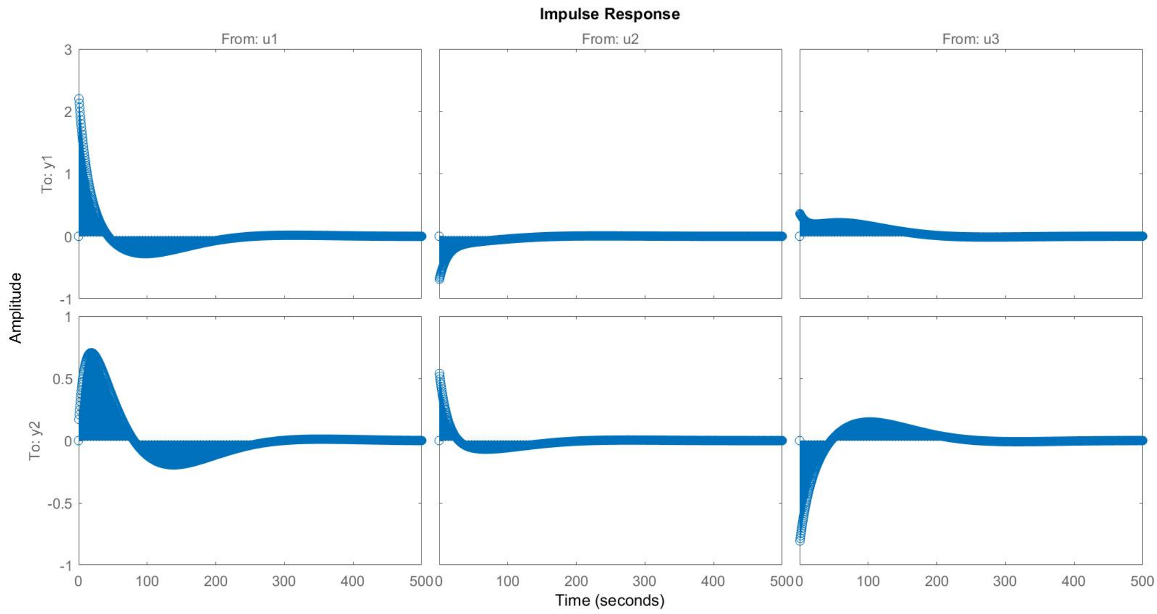


Figure 5.21 Impulse Response of the Closed-Loop MIMO Model

The Closed-loop MIMO Model in 5.7 is chosen to be implemented in the high-performance MIMO MPC. It is important to remark that complex MIMO model is obtained with the same iterative identification experiments applied to the simpler SISO and MISO models. Further analysis on the control performance of the MIMO model and its implementation to the proposed MPC Software are conducted in Chapter 6.

6 Case Studies for Control and the Results

The motivation for the case studies is to validate the applicability of the MPC Software [7]. Hence, the tuning of the high-performance MPC is required. The MPC Software has an interface where the user can define MPC parameters such as control horizon, prediction horizon, state matrices, weight matrix, and controller sampling time. The MPC Software is an external controller that is connected to the target computer with UDP communication protocol. Data transfer between the external MPC and the Multi-tank drivers were made with UDP packet interface of MATLAB/Simulink. Please not that UDP is used only because of Simulink real-time engine support, it does not support Modbus therefore it is not optimal for industrial level usage.

Case studies are designed to give an insight into the closed-loop control performances of model-based controllers obtained by the iterative identification experiments explained in Chapter 5. The case studies were performed initially on the internal MATLAB MPC function block, and the outcomes of tunings were studied. If the tuned MATLAB MPC provides off-set free control, the parameters of the particular controller are then pushed into the external MPC software. The flow chart of the design process of the external MPC Software is illustrated in figure 6.1. After the external controller is designed, the same case studies are applied, and the results are compared to MATLAB MPC to justify the performance of the proposed MPC Software.

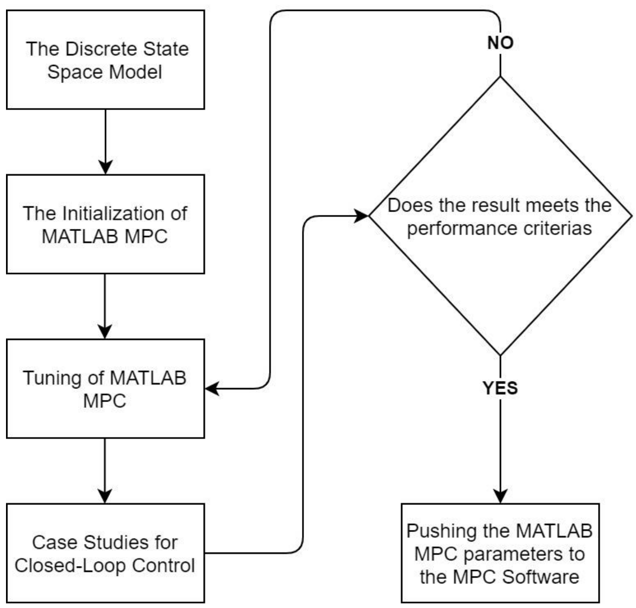


Figure 6.1 Design Process of the Controller with the MPC Software

Case studies are individually designed for each SISO, MISO, and MIMO model. Before continuing to the case studies, it is important to note that the implementation of the external MPC is made for SISO and MISO models. Case studies for the MISO model are designed to reflect the cost-efficient tuning approach of MPC.

6.1 Case Studies for the SISO Model

The state-space model in equation 5.4 is used to initialize the MATLAB SISO MPC. Design parameters are shown in Table 6.1. Based on the tune parametrization, three different SISO MPC models are created. Each SISO MPC has different weight coefficients that are indicated in table 6.2.

Table 6.1 Design Parameters of MATLAB SISO MPC

Design Parameters	Sampling Time	0.5 sec
	Prediction Horizon	12
	Control Horizon	5
	Pump input Constraints	Max = 1
		Min = 0.4
	Level Output Constraints	Max = 15
		Min = 0
	Input Slew Rate Constraints	Max = 1
		Min = -1

Table 6.2 Individual Weight Tuning Parameters

MPC 1			**MPC 2**			**MPC 3**		
Input	***Input Rate***	***Output***	***Input***	***Input Rate***	***Output***	***Input***	***Input Rate***	***Output***
0	0.001	2	0	0.5	0.1	0	0.5	2

The verbalization of the tunings of each MATLAB SISO MPC is made as follows:

1) For MPC 1:
    a) High values of Input do not penalize the cost function in control.
    b) High Rates of input manipulation have a negligible effect on the cost function.
    c) Off-set error in the output has a significant penalization on the cost function.
2) For MPC 2:
    a) High values of Input do not penalize the cost function in control.
    b) High Rates of input manipulation have a big penalization on the cost function.
    c) Off-set error in the output has smaller penalization on the cost function.
3) For MPC 3:
    a) High values of Input do not penalize the cost function in control.
    b) High Rates of input manipulation have a penalization on the cost function.
    c) Off-set error in the output has bigger penalization on the cost function.

Greater penalization on a parameter means corresponding squared error for a given parameter has bigger cost, thus controller effort will focus on such parameters to reduce the cost.

6.1.1 Case Study 1

Case Study 1 is a dynamic reference tracking test comprising increasing ramps, step jumps, step drops, decreasing ramps, and static references. The performance criteria are made with respect to the Root Mean Square Error (RMS), Response Time (T_r), mean value of the Inputs $(Input^{mean})$, and standard deviation of the manipulated variables (STD_{MV}). The case study takes 300 seconds, and 600 data samples were collected. The closed-loop graph of each MPC model output and control efforts are plotted, and performance metrics are shown in table 6.3.

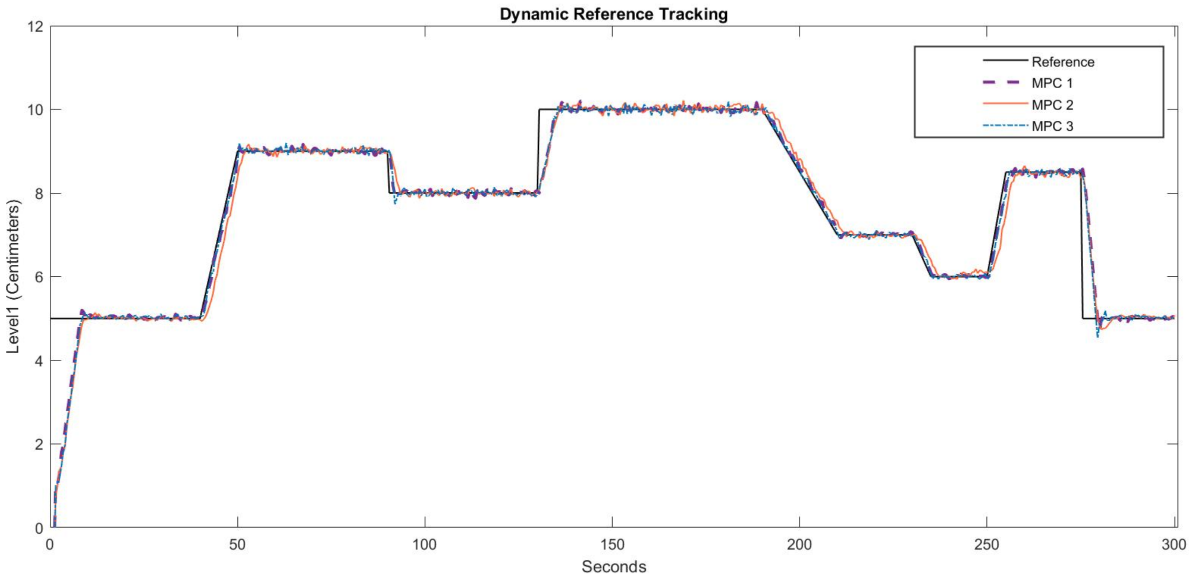


Figure 6.2 Graph of Control Output

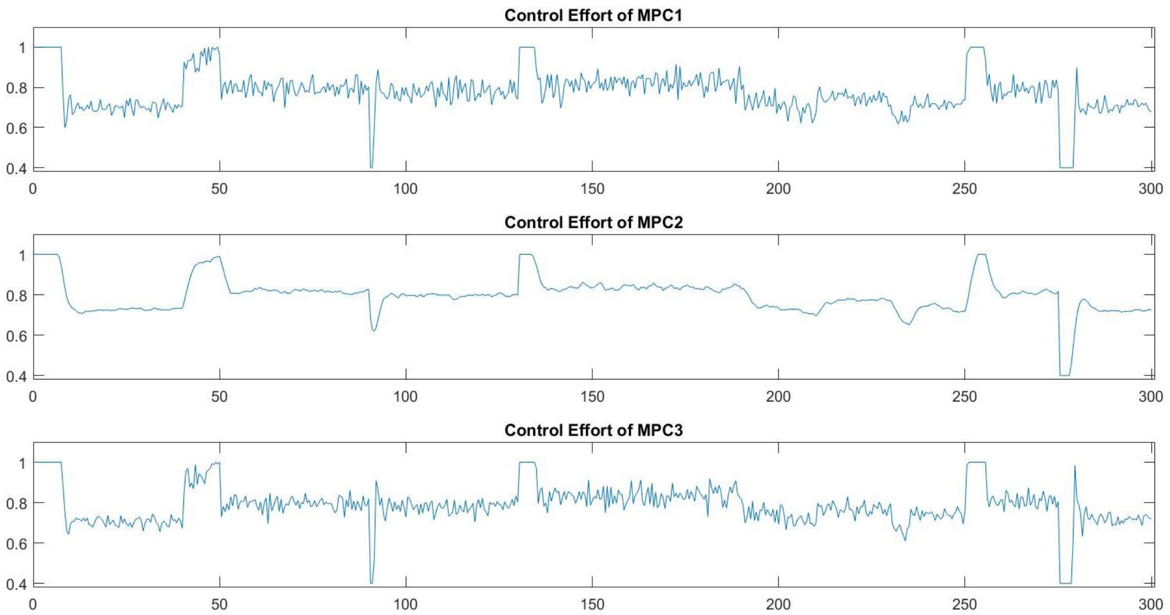


Figure 6.3 Control Effort Comparison of MPC 1 (top), MPC 2 (middle), MPC3 (bottom)

Table 6.3 Performance Results of the MATLAB MPC

Performance Criteria	MPC 1	MPC 2	MPC 3
$RMS\ Error$ (Ref Mean = 7.7102)	0.1822	0.2362	0.1874
T_r	Responsive	Less Responsive	Responsive
$Input^{mean}$	0.7765	0.7949	0.7870
STD_{MV}	0.0983	0.0856	0.0949

According to graph 6.2 and the result table 6.3, every MATLAB MPC provided offset-free control. However, the performance metrics in the result table indicate the effects of tuning. MPC 1 had no cost on using a high rate of input manipulation. Therefore, the controller could find the most optimal input values with the lowest input value mean and lowest RMS error. Yet, the input had a significant ripple that might not be mechanically optimal for the pump. A similar case is observed with MPC 3; however, MPC 3 realizes the rate of input manipulation as a cost factor. Consequently, there is a minor deviation in input values; still, the trade-off is visible in a slightly bigger RMS error. Lastly, MPC 2 provides the most mechanically friendly control effort, but it suffers from responsiveness and higher RMS error. Each controller has upsides and downsides so that the answer to picking the optimal controller depends on the application. Since the thesis intends to identify an offset-free MPC, MPC 1 could be the best choice however maintaining the safety of the Pump is also important for the Multi-tank system sustainability. Therefore, MPC 2 is chosen as the optimal tuned MPC design.

6.1.2 Case Study 2

The second case study was executed only on MPC 2. The objective is to observe the robustness of MPC 2 for a case where the dynamics of the controlled system are changed. Note that the SISO model did not realize the outflow valve; therefore, the valve1 signal is used to create a disturbance to the system. In case study 2, a constant set-point is provided for 300 seconds, at time = 75 seconds, Valve1 value is reduced from 1 to 0.75 and stays at 0.75 for 100 seconds to reduce the outflow. At time = 175 seconds, Valve1 signal increases back to 1 to increase the outflow causing the second disturbance. Figure 6.4 indicates the test result of MPC2. According to the graph, when the disturbance is applied at time = 75 seconds, the controller rapidly reduces the Pump values as a reaction to the disturbance. There exists no significant undesirable rise when the disturbance occurs. Additionally, at time = 175 seconds, disturbance changes the dynamics again, and the controller reacts by increasing the Pump values to keep the output level close to the reference level.

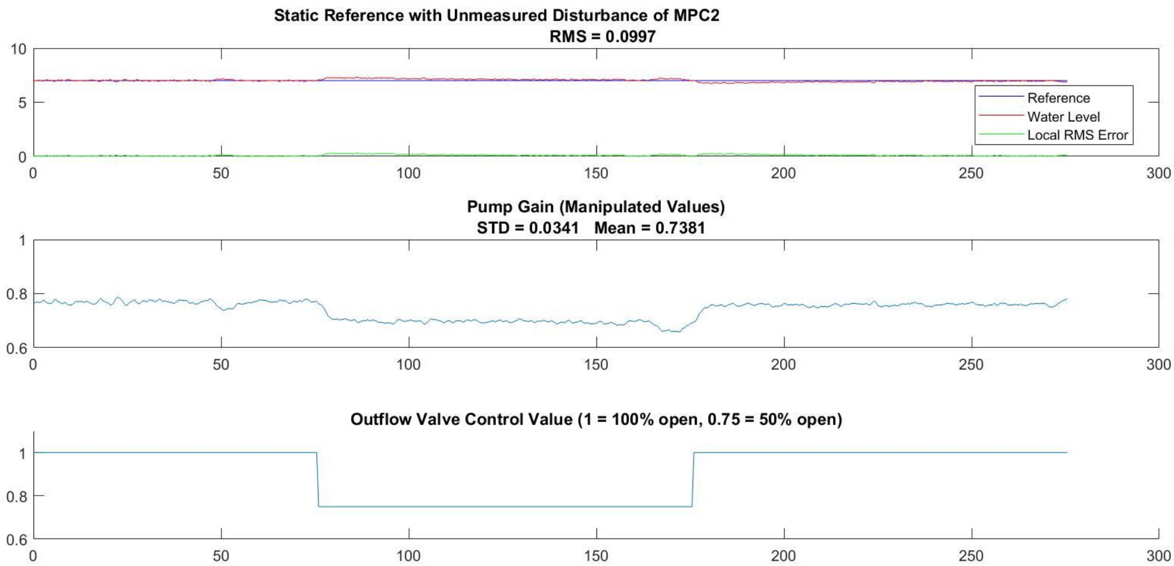


Figure 6.4 Results of Case Study 2

6.1.3 The Design of the External MPC and the Control

The design and tuning parameters of MATLAB MPC 1 is pushed into the MPC Software to generate the external MPC. Additional to the parameters, MPC Software also requires the tuning of the Kalman Observer, due to the impossibilities to obtain it from the MATLAB MPC. The tuning of the Kalman Observer is made by adjusting the Kalman Gain with input noise and output noise covariance matrices. The tuned Kalman Filter is:

$$Kalman\ Gain = \begin{bmatrix} 4.1177 \\ -0.9459 \end{bmatrix}$$

$$Initial\ Predicted\ Estimate = \begin{bmatrix} 0.4107 & -0.2938 \\ -0.2938 & 0.2149 \end{bmatrix}$$

The Case Study Results:

Case study 1 and 2 is applied to the external MPC. The graphical results of each case study are plotted in figure 6.5 and 6.6. The comparison of performance results of case study 1 are reflected in table 6.4.

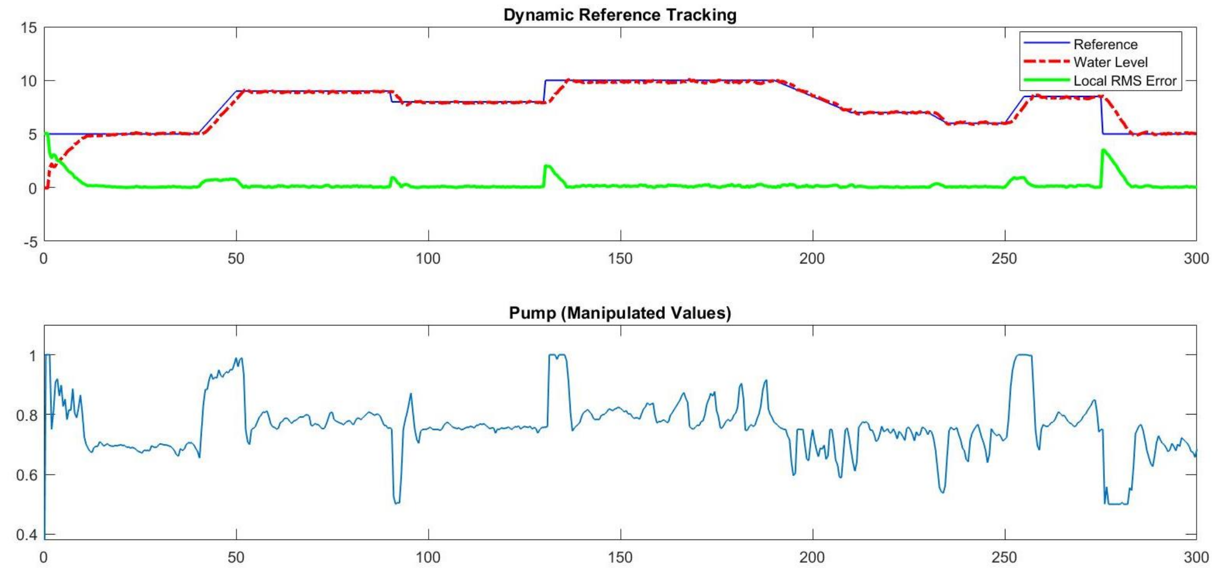


Figure 6.5 Results of Case Study 1 with the External MPC

Table 6.4 Case Study 1 Performance Comparison

Performance Criteria	**MPC 2**	**External MPC**
$\boldsymbol{RMS\ Error}$ (Ref Mean = 7.7102)	0.2362	0.2838
$\boldsymbol{T_r}$	-	Slower
$\boldsymbol{Input^{mean}}$	0.7949	0.7596
$\boldsymbol{STD_{MV}}$	0.0856	0.0969

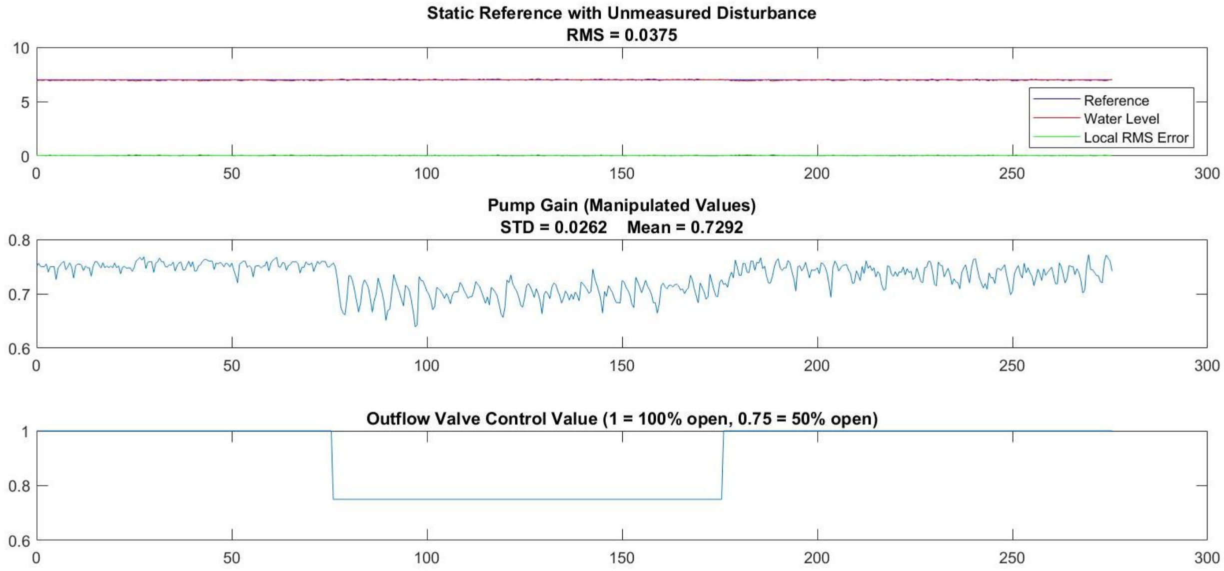


Figure 6.6 Results of Case Study 1 with the External MPC

Results indicate that, even though the weight matrix tuning of the MATLAB MPC2 and the external MPC are the same, there are some differences in the input manipulation. External MPC has more ripple as it manipulates the Pump input, and the responsiveness is slightly slower than MATLAB MPC 2. On the other hand, the RMS error in Case Study 1 is similar, and in Case Study 2, the external MPC performed better against the disturbance. Through the results, the design process of the external MPC (figure 6.1) is applicable to obtain an offset-free MPC.

6.2 Case Studies for the MISO Model

In this section, the cost-efficient tuning of an MPC is analyzed with two identical MATLAB MISO MPC. MISO model has additional input that also affects the output dynamics. Therefore, the controller can be tuned to prioritize particular input to push the output level to the set-point. Input prioritization can be made by concerning the financial constraints. In this case, utilization of Pump input is considered more costly than using Valve1 input. The case study will be a dynamic reference tracking, and the cost of utilizing the pump inputs is measured. The difference between individual MATLAB MPCs is their different weight matrices. MISO MPC 1 is tuned to allow high Pump input values, while MISO MPC 2 is tuned to restrict the utilization of high Pump input values.

Table 6.5 Design Parameters of MATLAB MISO MPC

Design Parameters of MISO MPC	Sampling Time	0.5 sec
	Prediction Horizon	10
	Control Horizon	3
	Pump input Constraints	Max = 1
		Min = 0.4
	Valve1 input Constraints	Max = 1
		Min = 0.5
	Level Output Constraints	Max = 15
		Min = 0
	Pump Slew Rate Constraints	Max = 1
		Min = -1
	Pump Slew Rate Constraints	Max = 0.5
		Min = -0.5

Table 6.6 Individual Weight Tuning Parameters

MPC 1			**MPC 2**		
Input	***Input Rate***	***Output***	***Input***	***Input Rate***	***Output***
$\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}$	0.001	1	$\begin{bmatrix} \mathbf{0.2} \\ \mathbf{0} \end{bmatrix}$	0.001	1

6.2.1 MISO Case Study and the Result

Case studies were held for 600 seconds, 1200 data samples were collected, RMS error, Pump input cost, and Pump input mean values were calculated.

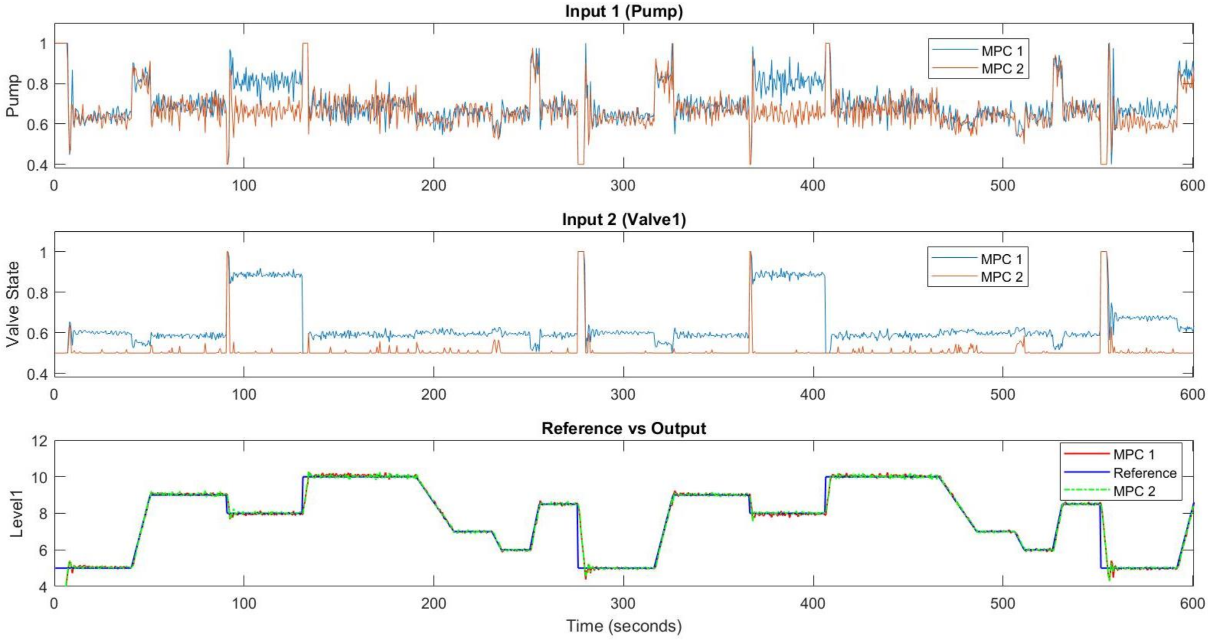


Figure 6.7 Closed-Loop Graph of MISO Case Study

Table 6.7 Comparison of the Results

Performance Criteria	**MPC 1**	**MPC 2**
$\boldsymbol{RMS\ Error}$ (Ref Mean = 7.7408)	0.1390	0.1345
Cost *(Time x Pump)*	840.4206	802.6450
$\boldsymbol{Input^{mean}}$	0.6998	0.6683

In Figure 6.7, it is seen that the MPC1 and MPC 2 had identical control performance for the Case Study. Input 1 graph reflects the difference in Pump input used by each controller for the control task. The significant Pump input difference is between the time 90 – 130 and 370 – 410 seconds. The exact output level can be reached with smaller Pump values by using smaller outflow values reflected in the Input 2 graph. Input 2 graph indicates the valve states of each controller. Valve states reflect the outflow rates, so that MPC 2 used lower outflow rates to control the level output. This difference in the input manipulation is caused by the higher weight in the Pump input of MPC 2, so smaller Pump input values are used to control the level output, and the Valve1 values are calculated by the controller accordingly.

Case Study resulted as follows: *For a given control task with MISO MPC, if the financial constraints are realized for a defined Input, the controller can be optimized to provide the most cost-efficient response*.

6.3 Case Studies for the MIMO Model

Case studies for the MIMO model aim to provide a relatively complex control task to the designed MIMO MPC defined in equation 6.1. Initially, MATLAB MIMO MPC is used to handle the control task. If a successful control is obtained, the MATLAB MIMO MPC parameters are pushed into the MPC Software to generate the external MIMO MPC. The MATLAB MIMO MPC is tuned to provide responsive performance; therefore, the output errors were penalized, and the input weights were neglected. Note that the MIMO has more parameters; thus, the input optimization requires more computational force. To compensate for computational heaviness, the sampling time of the controller is increased to 1 second, and the control horizon is reduced to 2. Table 6.8 shows the design parameters used in MATLAB MIMO MPC.

$$A = \begin{bmatrix} 0.9665 & -0.0122 & 0.0699 \\ 0.0093 & 0.9888 & 0.0053 \\ 0.0039 & -0.0028 & 0.9911 \end{bmatrix} \quad B = \begin{bmatrix} -1.9556 & 0.8039 & -0.6100 \\ -0.9791 & -0.2322 & 0.6292 \\ 0.0190 & -0.0545 & 0.0404 \end{bmatrix}$$

$$C = \begin{bmatrix} -0.4744 & -0.1758 & 0.0536 \\ 0.1703 & -0.4382 & -0.5695 \end{bmatrix} \quad D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{6.1}$$

$$U = [Pump \quad Valve1 \quad Valve2] \quad Y = \begin{bmatrix} Level1 \\ Level2 \end{bmatrix}$$

Table 6.8 Design Parameters of MATLAB MIMO MPC

Design Parameters of MIMO MPC	Sampling Time	1 sec
	Prediction Horizon	10
	Control Horizon	2
	Pump input Constraints	Max = 1
		Min = 0.4
	Valve1 input Constraints	Max = 1
		Min = 0.5
	Valve2 input Constraints	Max = 1
		Min = 0.5
	Level1 Output Constraints	Max = 15
		Min = 0
	Level2 Output Constraints	Max = 20
		Min = 0
	Pump Slew Rate Constraints	Max = 1
		Min = -1
	Valve1 Slew Rate Constraints	Max = 0.5
		Min = -0.5
	Valve2 Slew Rate Constraints	Max = 0.5
		Min = -0.5

6.3.1 MIMO Case Study and the Results

The control task is to hold Level1 output for 60 seconds for a given set-point while controlling the Level2 output with respect to the two-consecutive set-points. The control task gives five different set-points for the Level1 output, so there are ten set-points for the Level2 output. The case study took 300 seconds, and data samples were collected with 0.5 seconds of sampling time. RMS error is the performance criteria, and successful control is defined with 90% accuracy.

Result of MATLAB MIMO MPC:

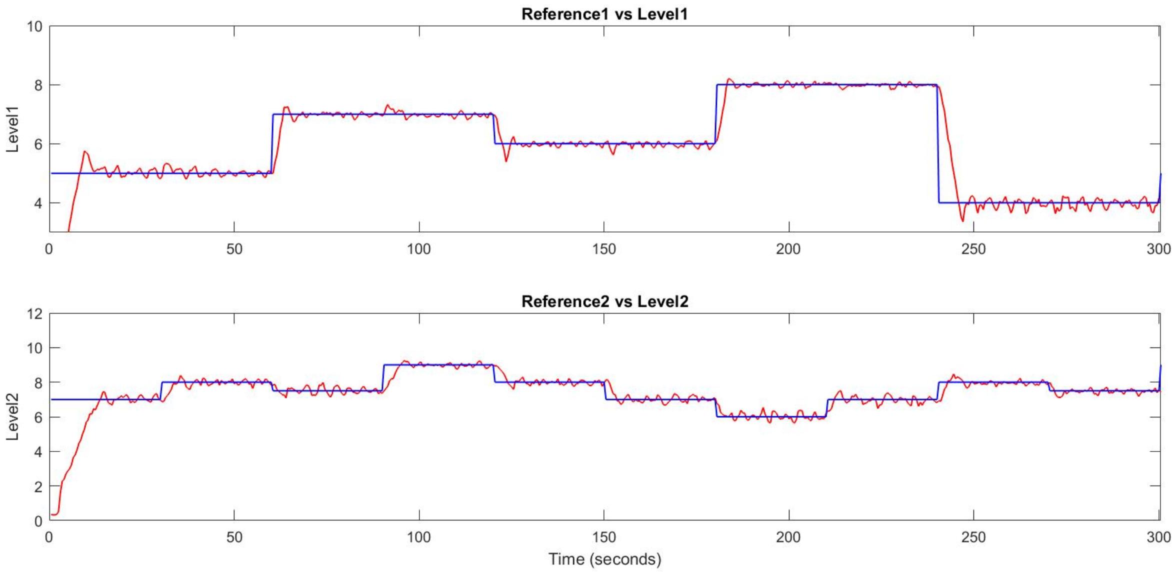


Figure 6.8 Control results of MATLAB MIMO MPC (red signal)

Closed-loop control of the MIMO system resulted in 0.2273 RMS error with 5.9983 average references in tank 1. The accuracy of the control is bigger than 90%; therefore, it is considered as successful. In tank 2 the RMS error is 0.3195 for the given 7.5025 average references, making control accuracy bigger than 90%. The noise in the measurement signal is due to the waves occurring in each tank and relatively bigger control sampling time.

The design parameters are found optimal to be pushed into the MPC Software as defined in figure 6.1. However, the External MPC is designed without optimal Kalman Filter parameters due to the lack of study made on tuning the Kalman Filter. This caused an under-tuned External MPC is used for the Case Study.

Result of the External MPC:

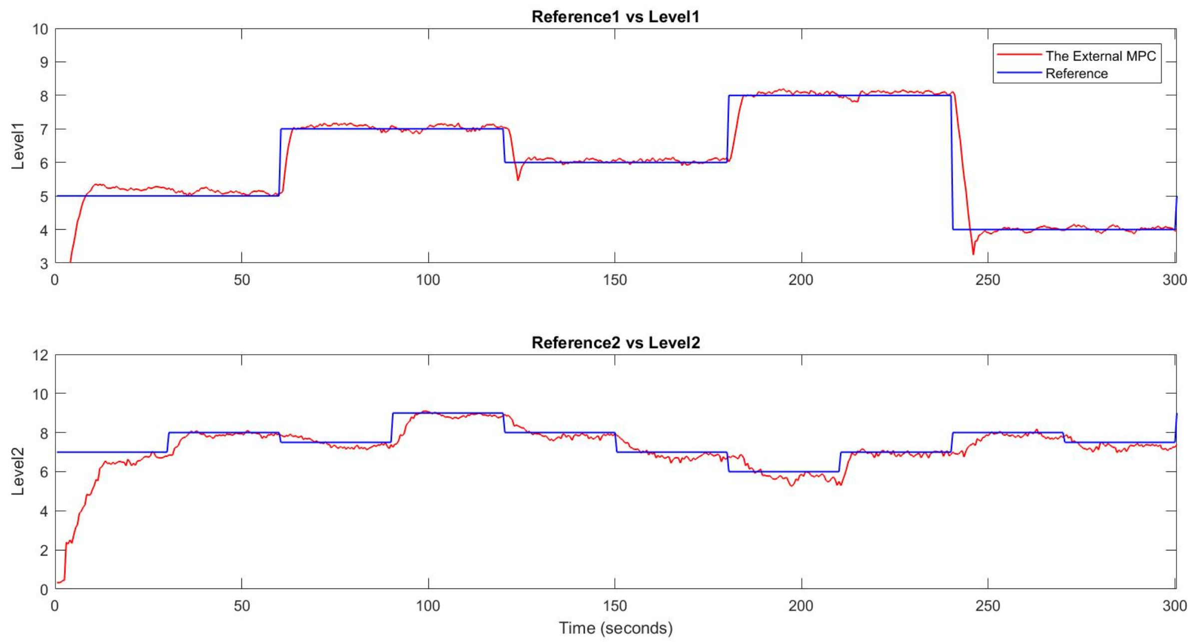


Figure 6.9 Control results of the External MPC (red signal)

Closed-loop control with the proposed MPC Software resulted in 0.2157 RMS error with 5.9933 average references in tank 1. The accuracy of the control is bigger than 90%; therefore, it is considered as successful. In tank 2 the RMS error is 0.4104 for the given 7.5025 average references, making control accuracy bigger than 90%. Even without an optimal Kalman Filter, the external MPC is able to provide optimal input manipulation to track the set-points given for tank 1 and tank 2.

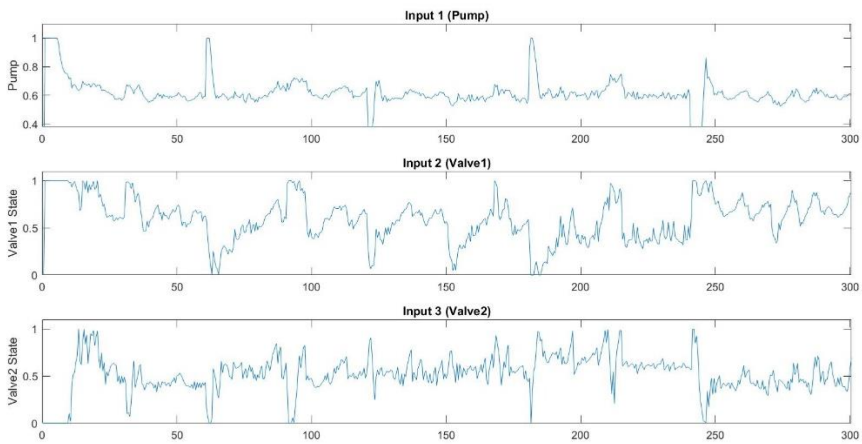


Figure 6.10 Input Manipulations by the External MPC

6.4 Discussion

The results of the case studies highlight that the proposed MPC software provided similar control performances with MATLAB MPC for SISO and MIMO cases. The case studies have validated the design steps of the external MPC illustrated in figure 6.1. As a result, initially, MATLAB MPC was used to optimize the tuning parameters for a given control task. According to the parameters used in MATLAB MPC, the MPC Software generated a high-performance external MPC to control the actual process. It is also important to note that water level readings in the Multi-tank system were excessively noisy due to the waves occurring as the water is pumped into the tanks. Therefore, the tuning of the Kalman Filter had a significant effect on obtaining the high-performance MPC since it was used as the state observer. The design approach used to generate external MPC could be customized to optimize the Kalman Filter. The utilization of MATLAB Kalman Filter application could yield more comprehensive MPC tuning; thus, better performing External MPC could be generated.

7 Conclusion

The thesis focused on controlling the Multi-tank system in TalTech Alpha-lab Control Laboratory with an external MPC designed with the proposed MPC Software. The case studies were designed to validate the applicability of an external MPC intended for a given control task. The results indicated that the external MPC precisely manipulated the input variables of the Multi-tank system and obtained closed-loop control with over 90% accuracy. This outcome may be considered a promising stage of implementing the MPC Software to industrial software such as Valmet DNA (DCS). Implementation of MPC Software to existing industrial software can utilize the process control of a district heating plant with the external MPC. Required works to do the implementations such as process identification, communication configuration between the external MPC and process control system, and MPC tuning were already covered in this study.

Another important outcome is the accuracy of the models obtained from a multivariable black-box system by using the iterative identification method. Local discrete state-space models were the final product of the identification stage, as they were generated by learning the dominant dynamics in the closed-loop system. These dynamics were accurately modelled to design offset-free model-based controllers. Moreover, the straightforward structure of multivariable representation of state-space models is highly beneficial in developing a multivariable model-based controller. In the process industry, numerous measurement data are obtained to increase efficiency and process quality. Continuously growing data archives of an industrial process could be used to model an optimal data-driven model-based controller. Considering the available data pool for a given plant, the experiment steps of the iterative identification method are highly favorable to analysis for an optimal controller.

Additionally, the findings from case studies also indicate the beneficial practicality of strategic linearization for a given control range. Although linearization reduced the model's accuracy over nonlinear curves, linear MPC was able to provide high-performance control within desired control ranges. Regarding the heating plant process, the plant outlet temperatures vary between 74° and 86° degrees Celsius, making 12° degrees Celsius output range. Taking account of the narrow working output range of the heating plant, linear MPC can be optimal for governing such tasks. Success in linear control may be considered a further validation of the MPC Software.

7.1 Kokkuvõte

Käesolev lõputöö keskendub TalTechi Alpha-lab kontroll-labori mitmemahutisüsteemi juhtimisele välise mudel-ennetava kontrolliga (MPC), mis on projekteeritud pakutud MPC tarkvaraga. Juhtumiuuringud kavandati mõttega kinnitada välise MPC rakendatavust etteantud kontrollülesandele. Tulemused näitasid, et väline MPC käsitses täpselt mitmemahutisüsteemi sisendmuutujaid ja saavutas suletud ahela kontrolli 90% täpsusega. Seda tulemust võib arvestada paljulubavaks etapiks MPC tarkvara rakendamisel olemasolevale tööstuslikule tarkvarale nagu Valmet DNA (DCS). MPC tarkvara rakendamine olemasolevale tööstuslikule tarkvarale võib kasutusele võtta kaugküttejaama tootmisjuhtimist koos välise MPC-ga. Antud uurimistöö hõlmas juba rakendamiseks vajalikud tööd nagu protsessi tuvastus, teabevahetuse seadistamine välise MPC ja protsessijuhtimissüsteemi vahel ning MPC häälestus.

Oluliseks tulemuseks on ka iteratiivset identifitseerimismeetodit kasutades mitme muutujaga plokk-kasti süsteemidest saadud mudelite täpsus. Kohalikud diskreetsed olekuruumimudelid olid tuvastamise etapi lõpp-produktiks, sest olid loodud suletud ahelaga süsteemi valitseva dünaamika õppimisel. Dünaamika modelleriti täpselt, et kavandada nihkevabasid mudelipõhiseid kontrollereid. Lisaks on olekuruumi mudeli mitme muutujaga esituse otsekohene struktuur väga kasulik mitme muutujaga mudelipõhise kontrolleri arendusel. Protsessitööstuses kogutakse arvukaid mõõteandmeid, et suurendada tõhusust ja protsessikvaliteeti. Tööstusprotsessi pidevalt kasvavaid andmearhiive on võimalik kasutada optimaalse andme- ja mudelipõhise kontrolleri modelleerimiseks. Arvestades mistahes tehase saadaolevat andmekogu, on iteratiivse identifitseerimismeetodi eksperimendietapid optimaalse kontrolleri analüüsi jaoks soodsad.

Lisaks osutavad juhtumiuuringute järeldused etteantud kontrollpiirkonna strateegilise lineariseerimise tegelikule praktilisusele. Kuigi lineariseerimine alandas mudeli täpsust mittelineaarsetel kurvidel, oli lineaarne MPC võimeline pakkuma kõrgjõudluslikku juhtimist soovitava kontrollpiirkonna raames. Küttejaama väljundtemperatuurid jäävad 74° ja 86° Celsiuse kraadi vahele, andes väljundivahemikuks 12° Celsiuse kraadi. Võttes arvesse küttejaama kitsa väljundivahemiku, võib MPC osutuda optimaalseks taoliste ülesannete haldamiseks. Lineaarse juhtimisega seotud edukus valideerib veelgi MPC tarkvara.

7.2 Recommendations for future work

This thesis validated the feasibility of the MPC Software, which was designed for governing industrial tasks, in a laboratory environment. Further implementations of the MPC software to a boiler system simulator or the actual district heating plant will finalize the validation progress. Despite the considerable work on identifying the multi-tank system, the model dynamics do not relate to the process in boiler systems. Therefore, future works must include a study of identification for a boiler system.

There is an increasing number of works on developing nonlinear system identification and nonlinear model-based controllers. Linear MPC is already being used in the industry, providing reliable, high-quality results. However, due to the limitation caused by linearizing the process, extra tuning effort is needed to optimize the controller. For this reason, it would be more promising to put effort into researching the development of a nonlinear MPC.

Abstract

Developing intelligent controllers is essential to achieve optimal and efficient control in process automation. In the context of district heating plants, this means a decrease in resource consumption and improvement in energy efficiency. With the emergence of Industry 4.0, controllers became a supervisory tool governing more complex tasks with multiple parameters where the conventional PID controllers cannot provide optimal results. On the other hand, model predictive control is a modern control method that can handle multivariable system control by its nature. The content of the thesis aims to successfully implement model predictive controllers to a multivariable water tank system to ensure that the proposed control method satisfies predefined control criteria. The thesis results have significant value in validating the feasibility of the proposed model predictive controller to district heating plants and raising the interest of local industrial entities towards these new developments.

Resümee

Tarkade juhtimisseadmete välja töötamine on tootmise efektiivse automatiseerimise juures esmatähtis. Keskküttejaamade kontekstis tähendab see ressursside tarbimise vähendamist ja energiaefektiivsuse tõstmist. Tööstus 4.0 tekkimisega muutusid kontrollerid järelvalvevahenditeks, mis juhivad keerulisemaid protsesse mitmete parameetritega, kus laialtlevinud PID kontrollerid ei anna optimaalseid tulemusi. Teisest küljest on ennetuslik kontroll moodne meetod, mis on loodud mitme muutujaga süsteemide jaoks. Käesoleva lõputöö eesmärk on edukalt rakendada ennetuslike kontrollerite mudeleid mitme muutujaga veemahutisüsteemidele, et kindlustada välja pakutud kontrollmeetodi sobivus kontrollkriteeriumitega. Lõputöö tulemustel on suur potentsiaal valideerida välja pakutud ennustava kontrolleri mudeli teostatavust keskküttejaamade kontekstis ja tõsta kohaliku tööstuse huvi uute arengute vastu.

LIST OF REFERENCES

[1] K. J. Åström and T. Hägglund, "The future of PID control," *Control Engineering Practice*, vol. 9, no. 11, pp. 1163–1175, Nov. 2001, doi: 10.1016/S0967-0661(01)00062-4.

[2] J. M. Maciejowski, P. J. Goulart, and E. C. Kerrigan, *Predictive control with Constraints*, vol. 346. Pearson College Div; 1st edition (December 1, 2001), 2007.

[3] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," vol. 11, pp. 733–764, 2003.

[4] C. E. Garda, D. M. Preti't, and M. Morari~, "Model Predictive Control: Theory and Practice a Survey*," 1989. Accessed: Jan. 04, 2021. [Online].

[5] "Statistical Database." http://andmebaas.stat.ee/ (accessed Jan. 04, 2021).

[6] "Current and future developments of district heating in Estonia." Accessed: Jan. 04, 2021. [Online].

[7] V. Vansovitš, "Advanced Control of District Heating Processes in Estonia," 2018.

[8] "MULTITANK SYSTEM." https://a-lab.ee/man/Multitank-user-manual.pdf (accessed Jan. 04, 2021).

[9] D. Hrovat, S. di Cairano, H. E. Tseng, and I. v Kolmanovsky, *The Development of Model Predictive Control in Automotive Industry: A Survey*. .

[10] R. E. KALMAN, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, vol. 82, no. D, pp. 35–45, 1960.

[11] R. Sivan and H. Kwakernaak, *Linear Optimal Control Systems*. John Wiley & Sons, Inc.605 Third Ave. New York, NYUnited States, 1972.

[12] J. H. Lee, "Model predictive control: Review of the three decades of development," *International Journal of Control, Automation and Systems*, vol. 9, no. 3, pp. 415–424, 2011, doi: 10.1007/s12555-011-0300-6.

[13] S. K. Lahiri, "Historical Development of Different MPC Technology," *Multivariable Predictive Control*, pp. 43–54, 2017, doi: 10.1002/9781119243434.ch3.

[14] J. Richalet, A. Rault, J. L. Testud, and J. Papon, "Model predictive heuristic control. Applications to industrial processes," *Automatica*, vol. 14, no. 5, pp. 413–428, Sep. 1978, doi: 10.1016/0005-1098(78)90001-8.

[15] C. R. Cutler and B. L. Ramaker, " Dynamic matrix control—a computer control algorithm," *AICHE national meeting*, 1979.

[16] C. E. García, D. M. Prett, and M. Morari, "Model predictive control: Theory and practice-A survey," *Automatica*, vol. 25, no. 3, pp. 335–348, May 1989, doi: 10.1016/0005-1098(89)90002-2.

[17] C. Cutler, A. Morshedi, and J. Haydel, "An industrial perspective on advanced control," *AICHE annual meeting*, Oct. 1983.

[18] K. Patan and J. Korbicz, "Nonlinear model predictive control of a boiler unit: A fault tolerant control study," *International Journal of Applied Mathematics and Computer Science*, vol. 22, no. 1, pp. 225–237, Mar. 2012, doi: 10.2478/v10006-012-0017-6.

[19] L. Dalhoumi, M. Chtourou, M. Djemel, and M. Djemel MohamedDjemel, "On the Fuzzy Model Predictive Control of Interconnected Nonlinear Systems," *Arab J Sci Eng*, vol. 42, pp. 2759–2776, 2017, doi: 10.1007/s13369-016-2412-z.

[20] R. Dollar, L. Melton, A. M. Morshedi, D. T. Glasgow, and K. W. Repsher, " Consider adaptive multivariable predictive controllers.," *Hydrocarbon Processing*, pp. 109–112, 1993.

[21] E. Kaiser, J. N. Kutz, and S. Brunton, "Data-driven discovery of Koopman eigenfunctions for control," *Machine Learning: Science and Technology*, pp. 1–36, 2021, doi: 10.1088/2632-2153/abf0f5.

[22] M. Klaučo and M. Kvasnica, "Control of a boiler-turbine unit using MPC-based reference governors," *Applied Thermal Engineering*, vol. 110, pp. 1437–1447, 2017, doi: 10.1016/j.applthermaleng.2016.09.041.

[23] X. Liu and J. Cui, "Economic model predictive control of boiler-turbine system," *Journal of Process Control*, vol. 66, pp. 59–67, 2018, doi: 10.1016/j.jprocont.2018.02.010.

[24] X. Wu, J. Shen, Y. Li, and K. Y. Lee, "Hierarchical optimization of boiler-turbine unit using fuzzy stable model predictive control," *Control Engineering Practice*, vol. 30, pp. 112–123, 2014, doi: 10.1016/j.conengprac.2014.03.004.

[25] Y. Li, J. Shen, K. Y. Lee, and X. Liu, "Offset-free fuzzy model predictive control of a boiler-turbine system based on genetic algorithm," *Simulation Modelling Practice and Theory*, vol. 26, pp. 77–95, 2012, doi: 10.1016/j.simpat.2012.04.002.

[26] X. Kong, X. Liu, and K. Y. Lee, "Nonlinear multivariable hierarchical model predictive control for boiler-turbine system," *Energy*, vol. 93, pp. 309–322, 2015, doi: 10.1016/j.energy.2015.09.030.

[27] M. Ławryńczuk, "Nonlinear predictive control of a boiler-turbine unit: A state-space approach with successive on-line model linearisation and quadratic optimisation," *ISA Transactions*, vol. 67, pp. 476–495, 2017, doi: 10.1016/j.isatra.2017.01.016.

[28] D. Piga, M. Forgione, S. Formentin, and A. Bemporad, "Performance-oriented model learning for data-driven mpc design," *IEEE Control Systems Letters*, vol. 3, no. 3, pp. 577–582, 2019, doi: 10.1109/LCSYS.2019.2913347.

[29] M. Gevers, "Identification for control: From the early achievements to the revival of experiment design," *European Journal of Control*, vol. 11, no. 4–5, pp. 335–352, 2005, doi: 10.3166/ejc.11.335-352.

[30] D. E. Rivera, H. Lee, M. W. Braun, and H. D. Mittelmann, "'Plant-Friendly' System Identification: A Challenge for the Process Industries," *Proc. of IFAC Symposium on System Identification*, no. Sysid, 2003.

[31] M. Verhaegen, "Identification of the deterministic part of MIMO state space models given in innovations form from input-output data," *Automatica*, vol. 30, no. 1, pp. 61–74, 1994, doi: 10.1016/0005-1098(94)90229-1.

[32] L. Wang, *Model Predictive Control System Design and Implementation Using MATLAB®*, no. 9781848823303. 2009.

[33] J. A. ROSSITER, *MODEL-BASED PREDICTIVE CONTROL: A Practical Approach*. CRC Press, 2003.

[34] E. Kaiser, J. Nathan Kutz, and S. L. Brunton, "Data-driven discovery of Koopman eigenfunctions for control," 2021. Accessed: May 09, 2021. [Online]. Available: https://github.com/eurika-kaiser/KRONIC.

[35] K. Patan and J. Korbicz, "Nonlinear model predictive control of a boiler unit: A fault tolerant control study," *International Journal of Applied Mathematics and Computer Science*, vol. 22, no. 1, pp. 225–237, 2012, doi: 10.2478/v10006-012-0017-6.

[36] B. B. Schwedersky, R. C. C. Flesch, and H. A. S. Dangui, "Practical nonlinear model predictive control algorithm for long short-term memory networks," in *IFAC-PapersOnLine*, Jan. 2019, vol. 52, no. 1, pp. 468–473, doi: 10.1016/j.ifacol.2019.06.106.

[37] D. Bruder, C. D. Remy, and R. Vasudevan, "Nonlinear system identification of soft robot dynamics using koopman operator theory," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2019-May, pp. 6244–6250, 2019, doi: 10.1109/ICRA.2019.8793766.

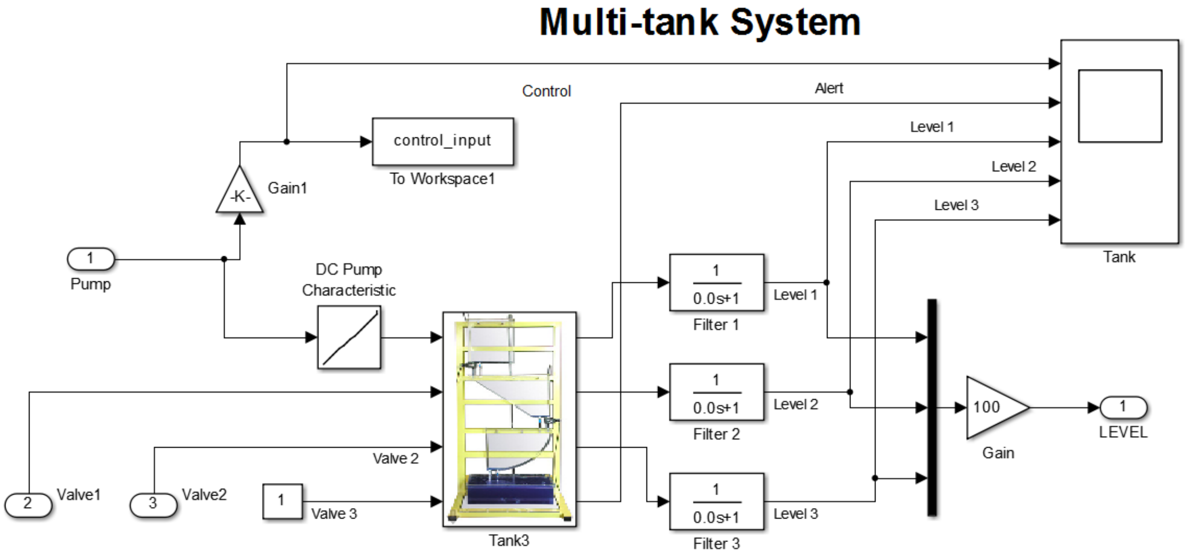
[38] E. H. Guechi, S. Bouzoualegh, Y. Zennir, and S. Blažič, "MPC control and LQ optimal control of a two-link robot arm: A comparative study," *Machines*, vol. 6, no. 3, pp. 1–14, 2018, doi: 10.3390/machines6030037.

[39] H. Hjalmarsson, M. Gevers, and F. de Bruyne, "For model-based control design, closed-loop identification gives better performance," *Automatica*, vol. 32, no. 12, pp. 1659–1673, 1996, doi: 10.1016/S0005-1098(96)80003-3.

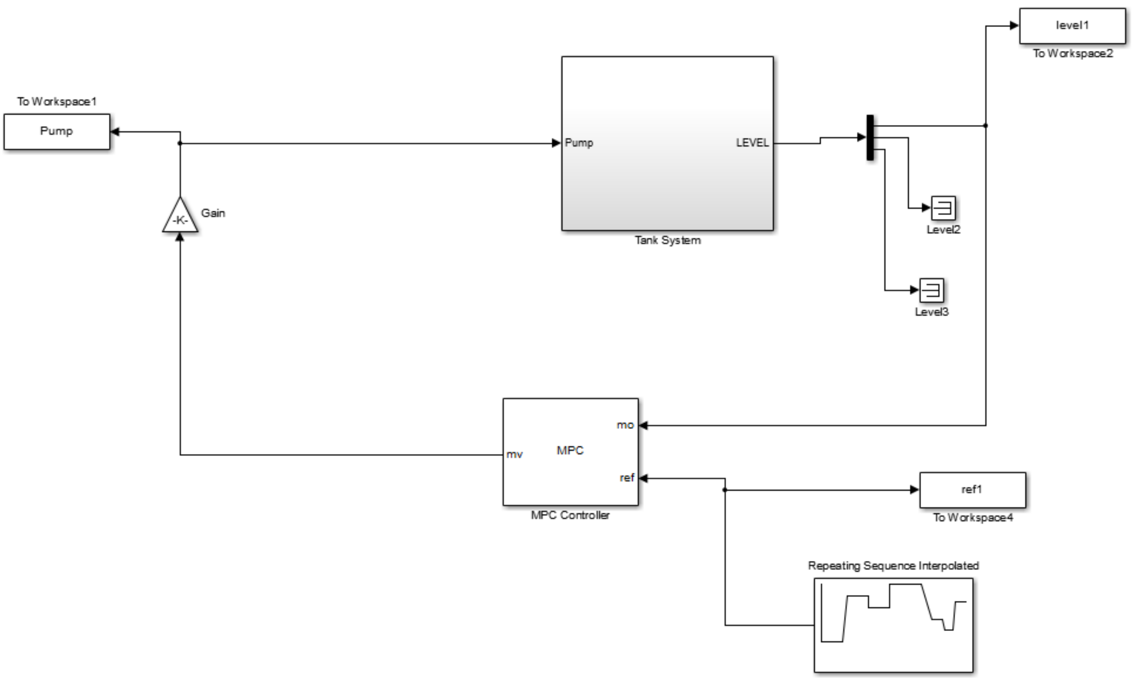
[40] L. Ljung, *System Identification Toolbox ™ User's Guide*. The MathWorks, Inc, 2016.

Appendix

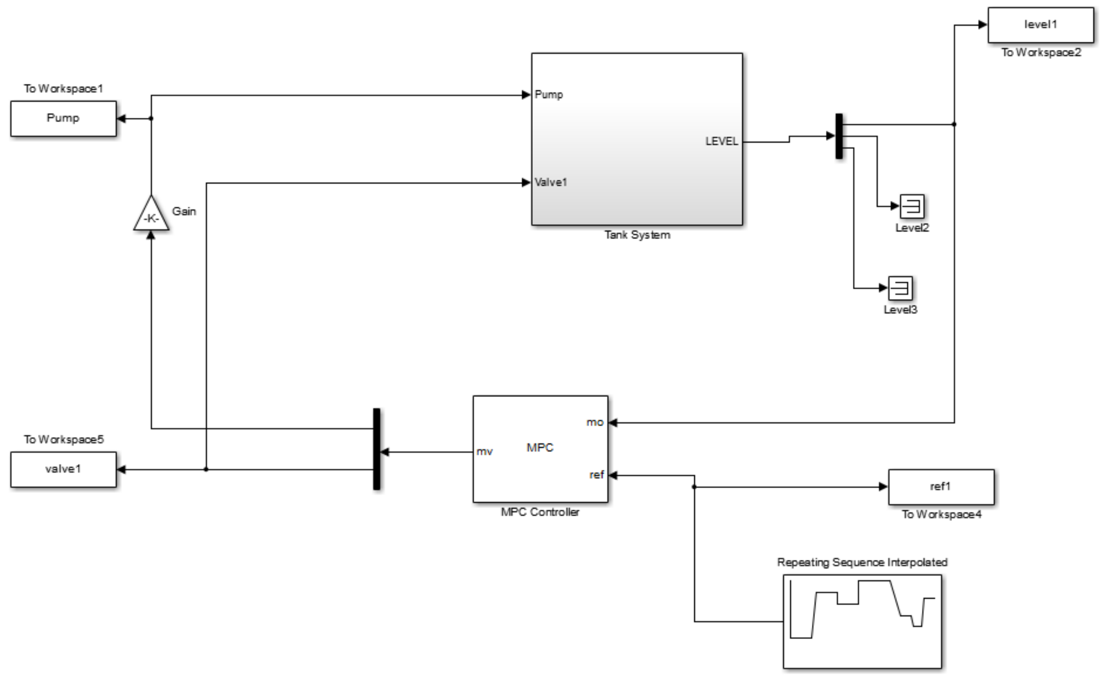
Simulink Block Diagram of the Multi-tank Driver



Simulink Diagram of SISO Configuration



Simulink Diagram of MISO Configuration



Simulink Diagram of MIMO Configuration with External MPC