SLOVAK UNIVERSITY OF TECHNOLOGY
IN BRATISLAVA

FACULTY OF CHEMICAL AND FOOD TECHNOLOGY

Reg. No.: FCHPT-16584-97476

Data-based Input-output System Identification

MASTER THESIS

2022/2023 **Bc. Martina Bujdáková**

SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA

FACULTY OF CHEMICAL AND FOOD TECHNOLOGY

Reg. No.: FCHPT-16584-97476

Data-based Input-output System Identification

MASTER THESIS

Study programme:	Automation and Information Engineering in Chemistry and Food Industry
Study field:	Cybernetics
Workplace:	Department of Information Engineering and Process Control, Slovnaft, a.s.
Thesis supervisor:	doc. Ing. Radoslav Paulen, PhD.

2022/2023 **Bc. Martina Bujdáková**

Slovak University of Technology in Bratislava
Department of Information Engineering and Process Control

Faculty of Chemical and Food Technology
Academic year: 2022/2023
Reg. No.: FCHPT-16584-97476

STU FCHPT

MASTER THESIS TOPIC

Student:	**Bc. Martina Bujdáková**
Student's ID:	97476
Study programme:	Automation and Information Engineering in Chemistry and Food Industry
Study field:	Cybernetics
Thesis supervisor:	doc. Ing. Radoslav Paulen, PhD.
Head of department:	doc. Ing. Martin Klaučo, PhD.
Consultant:	Ing. Karol Ľubušký, Ing. Martin Mojto
Workplace:	Department of Information Engineering and Process Control, Slovnaft, a.s.

Topic: **Data-based Input-output System Identification**

Language of thesis: English

Specification of Assignment:

In the process industry, there is often a situation where it is time- and financially demanding to create a (mechanical, physical, first-principles) mathematical model of a controlled process. In these cases, approximate input-output models are applied as a very effective alternative for such a model. These often have a linear structure for simplicity and are identified from measured data. The aim of this work is to investigate the identification of input-output models and the possibility of using this method in modeling the dynamic behavior of complex processes.

Tasks:
- Mastering the processing and cleaning of measured data
- Mastering the structure selection of input-output models for identification
- Use of identification of input-output models for a selected process of chemical technology (e.g. distillation column)
- Validation of the designed input-output model

Selected bibliography:

1. Graham C. Goodwin, Robert L. Payne, Dynamic System Identification: Experiment Design and Data Analysis. Academic Press, 1977.
2. N. Bhutani, G. P. Rangaiah, A. K. Ray. First-Principles, Data-Based, and Hybrid Modeling and Optimization of an Industrial Hydrocracking Unit. Industrial & Engineering Chemistry Research 2006 45 (23), 7807-7816 DOI: 10.1021/ie060247q

Deadline for submission of Master thesis:	14. 05. 2023
Approval of assignment of Master thesis:	03. 03. 2023
Assignment of Master thesis approved by:	prof. Ing. Miroslav Fikar, DrSc. – study programme supervisor

Honour Declaration

I declare that the submitted diploma thesis was completed on my own, in cooperation with my supervisor and consultants, with the help of professional literature and other information sources, that are cited in the Bibliography section. As the author of my diploma thesis, I declare that I didn't break any third party copyrights.

.....
Signature

Acknowledgment

I would like to express my gratitude to everyone who contributed to the completion of this thesis. First and foremost, I extend my heartfelt appreciation to my supervisor, doc. Ing. Radoslav Paulen, PhD., for his expert guidance, patience, invaluable insights, and unwavering commitment to providing me with extensive support throughout the entire process. I would also like to thank my consultant, Ing. Martin Mojto, for his patient guidance and valuable advice during the preparation of my diploma thesis. Additionally, I am deeply grateful to my consultant, Ing. Karol Ľubušký, for his industry expertise and for providing me with a practical glimpse into the world of industry. Last but not least, I would like to thank my family and my partner for their tremendous support and motivation throughout my entire studies.

Abstract

This thesis aims to identify and model the depropanizer column in the Fluid Catalytic Cracking unit of the Slovnaft refinery. The depropanizer column is a distillation column used for separating propane and lighter gases from the butane fraction. The historical data from online sensors is used to validate a first-principles model using gPROMS ModelBuilder software and a data-based model using the autoregressive model called ARX. A hybrid model is then created by combining the first-principles and data-based models. Three approaches of hybrid modeling are introduced - constant, static, and dynamic correction.

Keywords: first-principles modeling, data-based modeling, hybrid modeling, depropanizer column

Abstrakt

Cieľom tejto práce je identifikovať a modelovať deprozaničnú kolónu (depropanizér) v jednotke fluidného katalytického krakovania rafinérie Slovnaft. Depropanizačná kolóna je destilačná kolóna používaná na separáciu propánu a ľahších plynov z butánovej frakcie. Historické údaje z online senzorov sa používajú na zlepšenie modelu prvého princípu vytvoreného pomocou softvéru gPROMS ModelBuilder a modelu založeného na údajoch pomocou autoregresívneho modelu nazývaného ARX. Hybridný model je potom vytvorený kombináciou mechanistického a dátového modelu. Predstavené sú tri prístupy hybridného modelovania - konštantná, statická a dynamická korekcia.

Kľúčové slová: mechanistické modelovanie, dátové modelovanie, hybridné modelovanie, depropanizačná kolóna

Contents

List of Abbreviations

ACF	Autocorrelation function
AIC	Akaike information criterion
AR	Autoregressive
ARMA	Autoregressive moving-average
ARMAX	Autoregressive moving-acerage with extra input
ARX	Autoregressive with extra input
BIC	Bayesian information criterion
FCC	Fluid catalytic cracking
FPM	First-principles modeling
GMM	Gaussian mixture model
MCD	Minimum covariance determinant
MISO	Multiple-input single-output
MLE	Maximum likelihood estimation
MSE	Mean squared error
P&ID	Piping and instrumentation diagram
PACF	Partial autocorrelation function
PID	Proportional Integral Derivative
PML	Process Modeling Library
RMSE	Root mean squared error
SHV	Smallest half-volume
SISO	Single-input single-output

List of Figures

List of Tables

Chapter 1

Introduction

A model can be referred to as the imitation of reality, an abstraction of a real process or system [33]. A mathematical model is a specific form of representation. During the process of model building, we convert the problem from the real world into an equivalent mathematical problem. This can help us solve and understand the particular problem. However, the model must represent certain characteristics of the actual system, such as the correct response direction of outputs as inputs change, a proper structure that accurately represents the connection between the inputs, outputs and internal variables, and the correct short- and/or long-term behavior of the system [15].

Currently, mathematical modeling plays a very important role in almost every field such as physics, biology, chemistry, engineering [4], economics [13] and many others [3]. We do not even know that we encounter mathematical modeling on a daily basis without realizing it. Almost all of us carry a phone in our pocket that can be unlocked with a fingerprint or facial recognition. These technologies use sophisticated algorithms and models for identification based on mathematical modeling.

In the process industry, it is possible to use mathematical modeling to model complex and difficult processes [35]. Despite the fact that the more complex the system is, the more time and the deeper knowledge of the process it takes, such modeling can have many benefits. Mathematical modeling can serve as a prerequisite for several purposes such as the design and scaling up of processes, process control, optimization, developing mechanistic understanding, planning and evaluating experiments, troubleshooting and diagnostics, determining unmeasurable quantities, conducting simulations in place of costly experiments, and feasibility studies to assess the potential before building prototype equipment or devices [33].

First-principles models, also known as white box or mechanistic models, are developed based on the fundamental laws of conservation, including mass balance, component balance, and energy balance. First-principles modeling approach provides a physical

understanding of the process and describes the process behavior in terms of state and measured variables. The model state variable is the variable whose rate of change is described by the conservation balance. First-principles models can be developed even before the process exists and require dynamic equations supplemented with algebraic equations for heat and mass transfer or kinetics [35]. Since first-principles models are based on physical laws, they are often easier to understand and provide reliable extrapolations. However, developing these models can be a time-consuming and expensive process due to the requirement for specialized knowledge in the relevant field to derive equations from physical laws [8].

Since almost every chemical process is monitored, information about it is obtained in the form of data. Historical data provides process information and can be used to create a model. In this case, data-based modeling comes into play, which can extract information about the process using various techniques to create an accurate, precise, and flexible model. Data-based models, also known as empirical or black-box models, solely rely on input/output data to capture the relationship between the measured variables of a process, without describing the underlying physical phenomena [35]. Data-based modeling is particularly useful when time is limited for model development or when there is insufficient understanding of the process. Therefore, data-based modeling offers the solution to the challenges and disadvantages of first-principles modeling. However, these models do not provide insight into the underlying behavior of the modeled process [8] and have limited extrapolability [37].

Hybrid modeling represents a combination of first-principles and data-based models while using the advantages of both approaches. Compared to the first-principles models alone, hybrid models present more accurate prediction properties and, unlike the data-based models alone, achieve better interpolation and extrapolation properties [35].

The aim of this thesis is the system identification of a part of the Fluid catalytic cracking unit in the Slovnaft refinery called the depropanizer column. Depropanizer is a type of distillation column, that is classified among complex chemical processes. The depropanizer column serves to separate propane and the lighter gases from the butane fraction. There are several operational degrees of freedom that can be adjusted, including feed flow rate F, bottom product flow rate B, distillate flow rate D, reflux flow rate R, heat duty in the reboiler Q_B, and heat duty in the condenser Q_D. Historical measurements from online sensors are available for several of these variables, as can be seen in Figure 1.1. Temperature measurements from sensors located at the top T_{t} and bottom T_{b} of the column, as well as distillate temperature T_D, bottom product temperature T_B, and pressure measurements from sensors located at the top p_D and

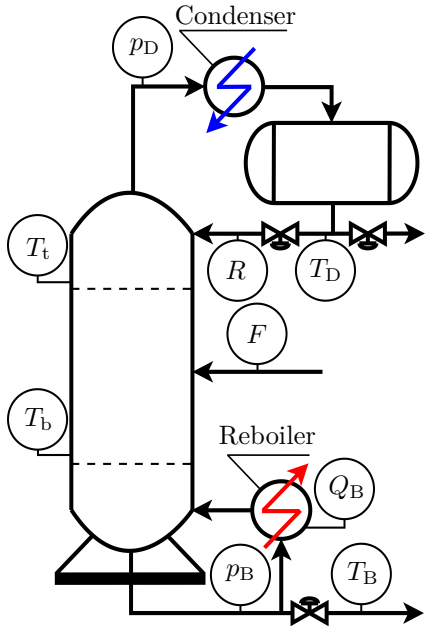

Figure 1.1: Scheme of the depropanizer column.

bottom p_B of the column, are also available as plant measurements. Historical data represents a week of production resulting in 9000 data points.

First-principles model of depropanizer is developed [26] using gPROMS ModelBuilder [1] software. The historical data contains the data from step tests that were performed on the feed flow rate. This data is used to improve performance and validate the gPROMS model against real measurements.

Data-based modeling includes initial data treatment in order to detect the data points that are statistically deviated and can be referred to as outliers. To study the relationships between measurements, correlation analysis is provided. An autoregressive model called ARX is introduced as a data-based model, that is used to predict the output based on past values of inputs and output.

A hybrid model that combines a first-principles gPROMS model and a data-based model is developed. The data-based model takes the role of correcting the error between real measurements and outputs from gPROMS model. Three approaches of hybrid modeling are introduced - constant, static and dynamic correction.

Chapter 2

Mathematical model building

2.1 First-principles modeling

A method for building mathematical models known as first-principles modeling (FPM) gives a quantitative account of the processes of how inputs influence outputs and key performance indicators that are connected to a process technical, economic, safety, and environmental performance. First-principles models differ from the empirical mathematical and statistical correlation that is based on and obtained from plant or other data. In contrast, first-principles models make use of fundamental engineering, physics, and chemistry concepts, such as mass and energy balances and they also involve physics-and/or chemistry-based definitions for the terms that appear in these balances [30].

First-principles models have a number of benefits, including the capacity to extrapolate over a broad range of operating circumstances and represent a process with great spatial and temporal resolution. These models must be created with a thorough understanding of the processes involved, and the modeling assumptions used can impact how well they function. Moreover, complicated first-principles models can be expensive to maintain, as follows it may be required to strike a compromise between sustainability and rigor. Despite these drawbacks, first-principles models are nevertheless helpful in the design and optimization of production processes [17].

The first-principles model requires a specialized modeling and simulation language for efficient continuous and/or discrete uniform simulations. These languages are often developed independently, such as gPROMS, Modelica, AMESim, or combined with simulation platforms such as Matlab/Simulink [8]. Simulators for operational optimization and operator training are just a few possibilities for using simulation technologies in process operations. These simulations, which are based on complex first-principles models, have been successfully used in many industrial applications. They are also often maintained for a long time [30].

An example of the use of this modeling approach, which can potentially be used in the industrial section, is the design and optimization of distillation columns. The purpose of these industrial units is to separate mixtures into individual components. First-principles modeling can be used to understand in detail how a column works or to see how column performance changes as process parameters change.

2.1.1 Model development process

Translating system into a precise and well-defined mathematical model is a challenging task. The mathematical model is created by converting the underlying physical principles and laws that describe a process into a system of mathematical equations. These equations may take the form of algebraic, differential, or a combination of both. The physical principles that underline a process may include mass and energy balances, thermodynamics, fluid mechanics, and chemical reactions, among others. The process of developing a first-principles model involves several steps [8], including:

- **Phenomenological Description**: The modeling object and experimental conditions are combined to form a verbal, pictorial, or other mental description.

- **Identification of Variables and Causality**: A system of causal relationships between specified variables, such as a block diagram, is then created from the description. This process is used to remove mathematically plausible but unnecessary relationships between the object's input and output variables. It may also introduce unmeasured internal variables or random disturbances.

- **Mathematical Modeling**: The specification of known relationships between variables, including parameterization, or the selection of structures for unknowable relationships, including disturbances. It is necessary to develop numerous hypothetical model structures of escalating complexity if it is uncertain how many or what kinds of relationships are required.

- **Calibration**: The simplest models that are not falsified by experimental data are found by fitting to the data and testing significance. The results allow for the evaluation of uncertainty and credibility and may prompt a return to the previous step.

- **Validation**: The model is examined using several sets of data. The calibrated model must be simplified if it is too complex for the intended use.

2.2 Data-based modeling

Data-based modeling is a mathematical model-building method based on experimental data or field operation datasets. Unlike the first-principles models, data-based models do not rely on any prior knowledge of the underlying physical and chemical processes [8]. In [22], these types of models are contrasted to what is commonly referred to as black-box models or statistical models.

Datasets frequently contain a richness of information that cannot always be understood by looking at data visualizations alone. We can gain new knowledge and insight from these datasets by building models based on the observed input and output data. In some circumstances, data-based models can also take the role of more intricate process-based models, particularly where computing speed is an important factor or the connections between inputs and outputs are unclear. Data-based models can take many different shapes, ranging from straightforward regression models to those that are based on biological or evolutionary processes [22].

Primarily due to its affordability, the data-based process modeling approach has gained significant use in the process industry and offers a number of benefits mainly because of its high cost-effectiveness. The majority of chemical processes are currently "data rich and information poor" according to practical engineers, who have benefited from studying the process operating data. Many data-based modeling approaches are being studied, but most of them can be generally grouped into two categories. The first category includes using artificial neural networks for constructing process models. The second category involves conducting statistical data analysis and building a model through regression [32].

In addition to the development of first-principles modeling approaches for distillation columns, numerous research studies have focused on incorporating data-based modeling techniques for controlling and optimizing distillation columns. In [7], a method is proposed that involves using clustering to extract steady-state operational data and selecting relevant input features to train an LSTM model that predicts steady-state operation in a distillation column. Given the complex and often unpredictable nature of distillation column operation, data-based models have emerged as useful tools for predicting unexpected operational conditions. [24] presents a novel strategy for fault detection in distillation columns using multiscale partial least squares. A data-based approach for detecting flooding in distillation columns through the utilization of dynamic principal component analysis and Bayesian inference is proposed in [21].

2.2.1 Data treatment

Measurements are often taken in the process industry for various reasons, including equipment performance monitoring, production process tracking, or process control. Nevertheless, the collected data may be flawed due to incompleteness, inconsistency, or broken sensors, which can adversely affect the subsequent analysis. Collected process data points are often contaminated with abnormal data points that differ from the other observations. These data points are called outliers. Detecting outliers in the multivariate dataset is very challenging and many detecting techniques were proposed [2]. In order to maintain optimal performance, product quality, efficiency, and safety in industrial production, it is essential to detect outliers. Detecting outliers is challenging in complex systems such as distillation columns, where variables interact intricately. Outliers could indicate issues within the system. However, identifying them might be complicated because an outlier in one variable may not necessarily be an outlier in another. Outliers can also skew statistical analyses and machine learning outcomes, potentially leading to biased results. Therefore, accurate outlier detection is important for reliable results in industrial production.

To ensure that the dataset is in a suitable format for future analysis, data pre-processing is needed. The multivariate dataset from industrial measurements, such as temperatures or mass flow rates, contains various data on different scales. The raw data might include features that are not useful for the intended purpose, such as using machine-learning algorithms. Appropriate data pre-processing can lead to a better understanding of correlations in the analyzed dataset.

In the process data, the important step of the pre-processing data is to define the deviation variable as

$$x_{\mathrm{dv},i} = x_i - x_{\mathrm{ss}}, \tag{2.1}$$

where $x_{\mathrm{dv},i}$ is deviation variable, x_i is the actual value and x_{ss} is steady-state value of variable. In complex systems, such as distillation columns, the definition of deviation variables may be difficult due to the presence of numerous steady states.

Minimum covariance determinant method

The Minimum covariance determinant (MCD) method is a commonly used approach for detecting outliers in multivariate data. The method is based on a distance measure called the Mahalanobis distance, which is defined as:

$$d_i = \sqrt{(x_i - \bar{x})^{\intercal}\, S^{-1}\, (x_i - \bar{x})}, \tag{2.2}$$

where x_i is the vector of scores on the set of p variables for subject i, $\bar{x}$ is the vector of

sample means on the set of p variables, and S is the covariance matrix. Large values of Mahalanobis distance suggest that observation is far from the center of the data. The goal of the MCD method is to identify a subset of observations that eliminate the presence of outliers by minimizing the determinant of the covariance matrix. In other words, the identified subset of observations creates the smallest volume of data points. The procedure for searching the subset with the smallest determinant is carried out using an iterative, multiple-step algorithm. The tuning parameter h represents the number of samples in the subset and is chosen according to the condition:

$$\frac{n}{2} < h < n, \tag{2.3}$$

where n represents the number of observations.

The algorithm starts by randomly selecting an initial subset of size h from the data, evaluating the values of the sample mean $\bar{x}$ and covariance matrix. Mahalanobis distance is then calculated for each observation by 2.2 and ordered from smallest to largest. The subset with the smallest values of Mahalanobis distance is retained and considered as the new subset of data h, and the value of the covariance determinant is calculated. In the next step, there are two possible options. If the value of the determinant of the covariance matrix for the new subset is higher than that for the previous subset of observations, the algorithm stops. Otherwise, the new subset of observations is considered as the subset for the next iteration of the algorithm [12]. Given the presence of randomness in the proposed method, it is useful to run the algorithm with various initial subsets of observations.

Smallest half-volume method

In [9], the Smallest half-volume method (SHV) was proposed for detecting outliers in multivariate analytical chemical data. Firstly, the Euclidean distance between each pair of observations is determined, creating a distance matrix D of size n by n with each row sorted in ascending order. Then, it is determined which row has the shortest sum of the first $n/2$ smallest distances. This represents the most stable section of the normal data and consists of the $n/2$ observations that are closest to one another in the multivariate space. Subsequently, the distributions of the Mahalanobis distances are obtained. The outliers are detected by comparing them with the χ^2 distribution at the level of significance $\alpha = 0.95$ [10]. However, the most consistent $n/2$ observations may still contain a significant fraction of the outliers under rare circumstances where the number of outliers is near 50% and the outliers are close to one another. For example, the outliers may be constant values in data from a stuck valve in the typical operational position, resulting in zero distances between the outliers. The SHV method is comparable to the MCD approach as it also identifies a subset of data points that best

represent the normal data while being stable. However, the SHV method is regarded as an improved version of MCD mainly because of its simplicity and computational efficiency [9].

Gaussian mixture model method

In the field of statistics, a Gaussian mixture model (GMM) [40] is a type of probabilistic model that assumes a set of data points follows a mixture of multiple Gaussian distributions with unknown parameters. The Gaussian mixture model is defined as

$$q(x;\theta) = \sum_{\ell=1}^{m} w_\ell n\left(x;\mu_\ell,\Sigma_\ell\right), \tag{2.4}$$

The model consists of a linear combination of m Gaussian models, each weighted according to $\{w_\ell\}_{\ell=1}^{m}$. Each Gaussian model is characterized by a mean vector μ_ℓ and a covariance matrix Σ_ℓ. The weight of each Gaussian model indicates the likelihood that a particular data point belongs to that specific model. The parameters of a Gaussian mixture model are estimated using the maximum likelihood estimation (MLE) algorithm. The likelihood function is maximized with respect to the parameters $\{w_\ell\}_{\ell=1}^{m}$, μ_ℓ, S_ℓ. The optimization problem for MLE involves constraints on the weights w_ℓ. To solve this problem, the weights are reparameterized as

$$w_\ell = \frac{\exp\left(\gamma_\ell\right)}{\sum_{\ell'=1}^{m}\exp\left(\gamma_{\ell'}\right)}, \tag{2.5}$$

where γ_ℓ is a free parameter, and then the likelihood function is maximized with respect to the parameters w_ℓ, μ_ℓ, S_ℓ.

The algorithm aims to fit the data to a combination of Gaussian distributions that have varying means and variances. The user selects the number of Gaussian distributions, which also determines the number of clusters. After fitting the GMM to the data, the algorithm computes the likelihood of each data point. Any data points with a low likelihood of belonging to any of the Gaussian distributions are considered outliers. The user can set the threshold for identifying outliers depending on the required level of sensitivity.

2.2.2 Variable selection

Selecting relevant variables is important in data analysis and modeling, particularly when we have available many variables but with limited understanding of their relationships. Choosing the right variables can improve the interpretability of results and

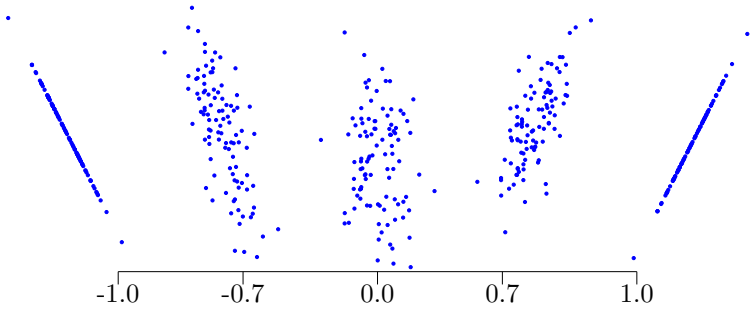

Figure 2.1: Correlation between two variables.

increase model accuracy. Correlation analysis is a widely used technique for variable selection. This statistical technique is specifically called the Pearson correlation method and is used for investigating the statistical relationships between two or more variables [5]. To interpret the correlation analysis numerically the correlation coefficient is used. The Pearson correlation coefficient r assigns a value between -1 and 1 and for the two variables $x_{i,1}$ and $x_{i,2}$ can be derived from:

$$r = \frac{\sum_{i=1}^{n} x_{i,1}x_{i,2} - \frac{\left(\sum_{i=1}^{n} x_{i,1}\right)\left(\sum_{i=1}^{n} x_{i,2}\right)}{n}}{\left[\frac{\sum_{i=1}^{n} x_{i,1}^2 - \left(\sum_{i=1}^{n} x_{i,1}\right)^2}{n}\right]\left[\frac{\sum_{i=1}^{n} x_{i,2}^2 - \left(\sum_{i=1}^{n} x_{i,2}\right)^2}{n}\right]}. \tag{2.6}$$

If the value of the coefficient is 0, it means that there is no correlation present [27]. If the score value is near $\pm$ 1, the correlation between the two variables is high. In other words, this means that as one variable increases, the other variable tends to increase and vice versa. A strong correlation between two variables occurs when the correlation score is between $\pm$ 0.50 and $\pm$ 1. If the score value lies between $\pm$ 0.30 and $\pm$ 0.49, then the correlation tends to be medium. While the score value lies bellow $\pm$ 0.29, we consider that the correlation among the variables is weak. Interpretation of the Pearson correlation method is also considered using the scatter plot seen in Figure 2.1, where values of one variable appear on the horizontal axis and the values of the other variable are on the vertical axis. By analyzing the scatter plot, we can often identify interesting patterns and relationships in the dataset.

2.2.3 Model selection

In the data-based modeling process, choosing the best model structure is crucial because it permits an accurate representation of the underlying system dynamics. The analysis of time series data gathered from sensors is crucial for the creation of efficient

models in the context of process industries, such as distillation columns.

In this thesis, time series data are analyzed using autoregressive models. A useful tool for understanding complex systems is the regression model known as the autoregressive model, which provides information on the relationships between recent and historical observations. The family of autoregressive models is introduced [11].

AR model

In autoregressive (AR) model, the lagged values are used as predictors. This means that the value at a given point in time is a linear function of its past values. The mathematical representation of AR model is following

$$y(t) = -a_1 y(t-1) - \cdots - a_{n_a} y\left(t - n_a\right) + e(t), \tag{2.7}$$

where $y(t)$ is the value of the time series at time step t, $a_1, ..., a_{n_a}$ are the model parameters, and $e(t)$ is the error term.

ARMA model

Let us consider a process $y(t)$ that can be represented as white noise that passes through a linear system described as:

$$y(t) = F(q)e(t), \tag{2.8}$$

where $e(t)$ is the white noise, q is the shift operator and $F(q)$ is represented as

$$F(q) = \frac{C(q)}{A(q)} = \frac{1 + \sum_{i=1}^{n_c} c_i q^{-i}}{1 + \sum_{i=1}^{n_a} a_i q^{-i}}, \tag{2.9}$$

where n_a/n_c is the numerator/denominator order. The ARMA (autoregressive moving-average) model can be expressed as:

$$\begin{aligned} y(t) = &- a_1 y(t-1) - \cdots - a_{n_a} y(t-n_a) + \\ &e(t) + c_1 e(t-1) + \cdots + c_{n_c} e(t-n_c), \end{aligned} \tag{2.10}$$

where $y(t)$ is the value of the time series at time t, $a_1, ..., a_{n_a}$ and $c_1, ..., c_{n_c}$ are the model parameters and $e(t)$ is the error term. While the first autoregressive (AR) part describes the dynamics of the output $y(t)$, the MA component models the influence of the disturbance variable $e(t)$.

ARX model

ARX model is an extension of AR model that includes previous values of input variable as additional predictors. The mathematical representation of ARX model is the following

$$\begin{aligned} y(t) = &- a_1 y(t-1) - \ldots - a_{n_a} y\left(t - n_a\right) + \\ &+ b_1 u(t-1) + \cdots + b_{n_b} u\left(t - n_b\right) + e(t), \end{aligned} \tag{2.11}$$

where $y(t)$ is the value of the time series at time t, $a_1, ..., a_{n_a}$ and $b_1, ..., b_{n_b}$ are the model parameters and $e(t)$ is the error term.

ARMAX model

ARMAX model is a popular variation of the autoregressive model with exogenous variables (ARX) that accounts for the influence of past error terms on the output variable. In other words, ARMAX model extends the ARX model by adding a moving average (MA) component for the error term. This makes ARMAX model suitable for modeling dynamic systems that are affected by both internal and external factors. Like the ARX model, the ARMAX model is a linear model that assumes that the system can be expressed as a difference equation

$$\begin{aligned} y(t) = &- a_1 y(t-1) - \cdots - a_{na} \\ &y(t - na) + b_1 u(t-1) + \cdots + b_{nb} u(t - nb) \\ &+ e(t) + c_1 e(t-1) + \cdots + c_{nc} e(t - nc). \end{aligned} \tag{2.12}$$

2.2.4 Order determination

The choice of the structure of a data-based model comes with the challenging task that involves the determination of the model order. In a family of autoregressive models, the order of the model corresponds with the number of previous values that are used to predict the current value. By choosing too low order of the model, we may end up with a situation where the model does not capture trends in the data and the prediction of such a model will be very poor. This may lead to underfitting the data, which means the model is not effectively representing the data. On the other hand, the choice of a higher order could result in overfitting. Due to overfitting, the model fits badly on the testing data but works perfectly on the training data. This is because an over-fitted model does not capture underlying patterns or relationships in the data, but captures unavoidable noise in the training data [43]. The optimal model order is then a compromise of the model complexity and its performance.

Autocorrelation function

Autocorrelation function (ACF) is a statistical tool for measuring the correlation between a time series and its previous values. The autocovariance at lag h [20] is given by:

$$\gamma_X(h) = \mathrm{Cov}\left(y(t), y(t-h)\right) = E\left[\left(y(t) - \mu_X\right)\left(y(t-h) - \mu_X\right)\right], \tag{2.13}$$

where $y(t)$ is a stationary time series with length T, $y(t-h)$ is the lagged time series by h periods and μ_X is the expected value of $y(t)$. The autocorrelation of $y(t)$ is defined as following

$$\rho_X(h) = \mathrm{Cor}\left(y(t), y(t-h)\right) = \frac{\gamma_X(h)}{\gamma_X(0)} = \frac{E\left[\left(y(t) - \mu_X\right)\left(y(t-h) - \mu_X\right)\right]}{E\left(y(t) - \mu_X\right)^2}. \tag{2.14}$$

Partial autocorrelation function

To measure the degree of association between $y(t)$ and $y(t-h)$ while not considering the effect of the other time lags, the partial autocorrelation function (PACF) is used. The PACF is defined as function $\alpha(\cdot)$ as:

$$\begin{aligned} &\alpha(0) = 1, \\ &\alpha(h) = \phi_{hh}, \quad h \geq 1, \end{aligned} \tag{2.15}$$

where ϕ_{hh} is the last component of

$$\phi_h = \Gamma_h^{-1}\gamma_h,$$

where $\Gamma_h = [\gamma(i-j)]_{i,j=1,\ldots,h}$, and $\gamma_h = [\gamma(1), \ldots, \gamma(h)]'$.

The visualization of ACF and PACF can be helpful to identify patterns in the data that are repeating over time, for example, seasonal effects such as changing temperature over day and night. Both functions take place in the determination of the order of the autoregressive model [41]. The order is determined by the point at which values drop below a certain significance level [28]. For example, if the ACF or PACF quickly decays, it could indicate that the lower order model would be able the capture the characteristics of time series data.

2.2.5 Model evaluation metrics

To evaluate the performance of models, it is important to have objective measures of their accuracy and fit to the data. To compare the accuracy of models based on their

predicted responses, it is common to use measures such as mean squared error (MSE) and root mean squared error (RMSE). On the other hand, criteria such as Akaike and Bayesian information criteria are used to compare and select model.

Mean squared error

Mean squared error [18] measures the average difference between the predicted values $\hat{y}$ and the real values y of the variable given as

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2, \tag{2.16}$$

where n denotes the number of data points.

Root mean squared error

Root mean squared error measures the squared average difference between the predicted values $\hat{y}$ and the real values y of the variable given as

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}. \tag{2.17}$$

Akaike information criterion

A commonly used approach for determining model accuracy is through the use of Akaike information criterion (AIC). AIC is a statistical measure that evaluates the suitability of the model by comparing the actual probability distribution of output real values y to that of predicted values $\hat{y}$. Mathematically, AIC is given by:

$$\mathrm{AIC} = n\log(\hat{\sigma}^2) + 2|M|, \tag{2.18}$$

where N represents the number of data points, $\hat{\sigma}^2$ is the variance of the model prediction error given by the difference between the real values y and the value predicted by the model $\hat{y}$, and M is the total number of model coefficients.

Bayesian information criterion

Another used approach for selecting a model is through Bayesian information criterion (BIC) given by the formula

$$\mathrm{BIC} = n\log(\hat{\sigma}^2) + M\log(N). \tag{2.19}$$

BIC is considered to be more reliable than AIC in many model selection scenarios due to its asymptotic consistency. This is because BIC has a stronger penalty term for the number of model parameters in 2.19, making it more likely to select a less complex model. In practical applications, BIC is commonly used to select the optimal model for subsequent inferences, while AIC is often used to identify the best prediction model [29]. A model with a lower AIC or BIC value is considered to have a better fit which indicates it is more likely to predict new observations accurately [25].

2.3 Hybrid modeling

Hybrid modeling can be thought of as a combination of two modeling principles - first-principles and data-based modeling. The original idea of hybrid modeling was introduced in 1992 by a paper by Psichogios and Ungar [31], where the goal was to obtain a more reliable and easier-to-interpret model by combining a first-principles model with a data-based model. The first-principles model provided prior knowledge about the modeled process of the fed-batch bioreactor, while the data-based model in form of the neural network was used for the estimation of unmeasured process parameters that are challenging to model using first-principles.

Hybrid modeling serves as a modeling enhancement and provides many benefits. Considering the complexity of process model development, first-principles modeling requires a very good knowledge and understanding of the physics and mechanisms of the process, whereas with data-based modeling no prior knowledge of the process is required. In this case, in hybrid modeling, a combination of both types of knowledge is required. The complexity of hybrid models varies depending on how much knowledge and understanding are included in the modeling. Considering the performance of hybrid models, through the proper combination of the strengths of both modeling approaches, more accurate results can be obtained. Hybrid models represent a viable trade-off between model accuracy and computational tractability, albeit with potential increases in computational and data requirements compared to their single-model counterparts, namely data-based or first-principles models.

Nowadays, with the advent of Industry 4.0, hybrid modeling has taken on a new dimension. This combination of primary and data-driven models brings new opportunities in the fields of engineering, energy, and especially the process industry. The article [36] presents many applications of hybrid modeling dating back to 1992 and today. The article mentions areas, where hybrid models are used. One of the mentioned is process control, where hybrid models have been integrated into model predictive control [44, 42] to predict future states of variables and adjust control inputs to

maintain optimal performance. It has been shown to improve prediction accuracy, as demonstrated by the fusion of physics-based and measurement-based models using a particle filter for predictive maintenance [14]. Hybrid models have also been used for process monitoring, fault detection [16], and prediction of flux evolution [19] and reaction kinetics in batch processes and bioprocesses, utilizing both batch-run specific and process specific information.

2.3.1 Model development

To construct a hybrid model, one must first evaluate the requirements of the modeled process. Many variables impact the choice of the modeling approach and the selection of first-principles and data-based models, including the amount of process data available, the studied system complexity, or available software. Creating a hybrid model necessitates a thorough grasp of the process under consideration, the available datasets, and the restrictions imposed by the modeling methodologies used. The purpose of hybrid modeling is to build an efficient model that may be used for future research or practical applications. There are several ways to develop the hybrid model, including the series model, parallel model, series-parallel/combined approach [6]. Scheme of each model is seen in Figure 2.2. We will explain the use of these principles using a practical example. Let us imagine that we have a first-principles model of a distillation column, but its outputs differ from real measurements. The goal is to improve the predictions by creating hybrid model. After a deeper analysis, it was found that the deviations of first-principles model are caused mainly by impurities in the feed, that cannot be included.

In the case of the series hybrid modeling approach, a combination of first-principles and data-based model is included to improve the accuracy of predictions. This approach is used when the physics of the system is sufficiently understood by a first-principles model, but there are parameters that are uncertain and difficult to measure or model. In our case, the undetermined parameters would correspond to the composition of the feed, which is estimated using a data-based model. Subsequently, the estimated parameters would be included as input to the first-principles model.

In the parallel hybrid approach of first-principles modeling, the model is used to capture the behavior of the system. A data-based model, on the other hand, serves to predict and correct residual model errors based on the difference between actual measurements and the output of the first-principles model. While the first-principles model of the distillation column serves to predict outputs based on physical and chemical equations, the data-based model takes into account the first-principles differences caused by uncertainties in the system. The data-based residual model is also referred to error

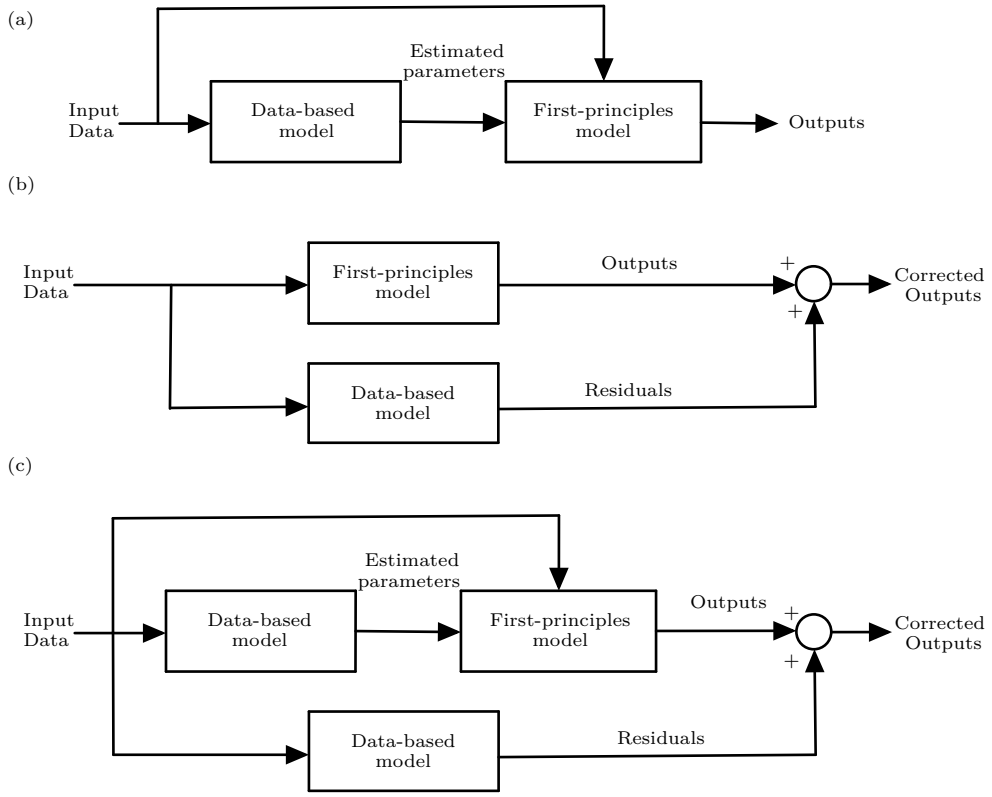

Figure 2.2: Scheme of series (a), parallel (b) and series-parallel (c) model.

correction model [34].

In this combined approach, two data-based models are combined with a first-principles model in a series-parallel connection. One data-based model is trained for the series model to estimate the parameters for the first-principles, while the second data-based model served to correct for the residuals between the plant data and the series model predictions. This approach takes advantage of both series and parallel models and can improve accuracy in predicting behavior in complex systems such as a distillation column.

Chapter 3

Case Study

3.1 Depropanizer column

The focus of the thesis is to model the distillation column in the Slovnaft refinery, which is an essential part of the Fluid catalytic cracking (FCC) unit. The type of distillation column studied in this section is called the depropanizer column. To study detailed column information and comprehensively understand the process, we were provided with a piping and instrumentation diagram (P&ID) [38] that displays the engineering details of the process equipment including the shared devices, instruments, valves, or pumps. However, due to the nondisclosure agreement with Slovnaft, the description of the depropanizer column and its control configuration cannot be fully stated.

The distillation column [39] or distillation tower is the most commonly used separation unit in refineries. The operation of the distillation column is based on a process called distillation, which separates the components of a mixture based on their different boiling points. The depropanizer column, in particular, separates propane and lighter gases from the butane fraction, with the vapor of the propane and lighter gases rising to the top and hydrocarbons with higher boiling points falling to the bottom of the column. Several major parts of the distillation column are the reboiler, condenser and tower called the column.

The measurement of pressure is important for the optimal operation of the distillation column as it is directly linked to changes in the relative volatilities [23] of the components in the mixture. The pressure in the column is measured at the head and the bottom of the column. The differential pressure sensor is used to measure the difference in pressure between the two points to determine the pressure drop across the column.

The feedstock mixture, which is a distillate product from the previous column in the FCC unit seen in Figure 3.1, known as the debutanizer column, is introduced to a feed

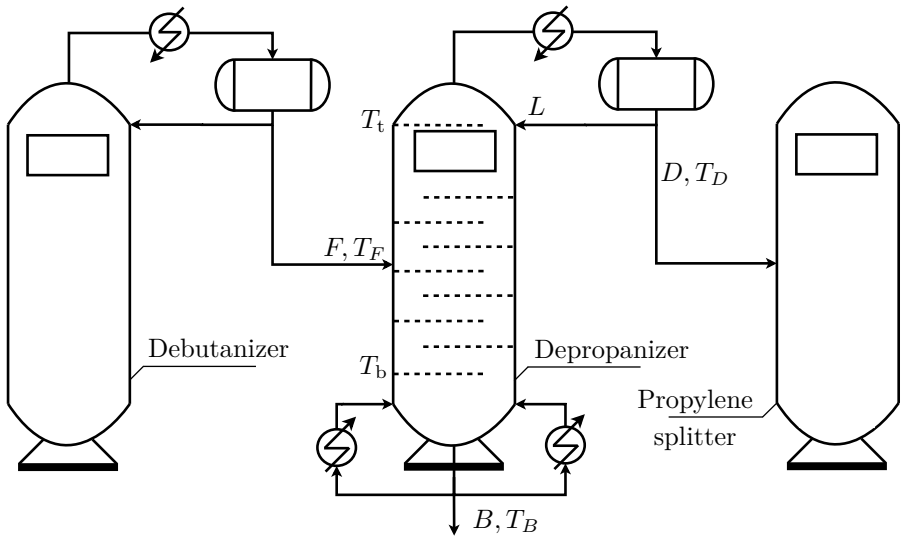

Figure 3.1: Scheme of Fluid catalytic cracking unit in Slovnaft refinery.

tray in the column that divides it into a stripping (bottom) section and an enriching (top) section. The feedstock mixture is measured by a temperature sensor and consists of nine components- propane, propylene, isobutane, n-butane, 1-butene, isobutene, trans-2-buten, cis-2-buten, 1,3-buadiene and isopentane. However, due to the exterior placement of the column, the temperature varies throughout the day and night. The temperature profile of the column in the enriching (top) section is monitored by a temperature sensor located at top of the column. Another temperature sensor located in the stripping (bottom) section of the column represents the temperature of that section, which is typically higher than the feed temperature due to the increasing presence of less volatile components [26].

The vapor leaving the top of the column is condensed and continues to a condenser where it is cooled. A temperature sensor measures the temperature of the condensate that is leaving the condenser. The condensate then enters a reflux drum where it is separated into two streams. One stream is the reflux flow, which is recycled back to the top of the column. The second stream is the distillate flow, which is removed from the system. The distillate product consists of propane and propylene. The distillate product is then used as a feedstock mixture for the third column of the FCC unit called propylene splitter [26]. The temperature of the distillate product is measured by a temperature sensor.

There are two reboilers in the lower part of the column. The naphtha pump around is used as the heating medium in the first reboiler with temperature sensors placed at the inlet and outlet. The second reboiler uses light cycle oil as the heating medium. The heat input into each reboiler is calculated from 3.1

$$Q = mc_p(T_{\text{in}} - T_{\text{out}}), \tag{3.1}$$

where m is the mass flow rate of heating medium, c_p is specific heat capacity of each heating medium, T_{in} is the temperature of heating medium at input and T_{out} at output [26].

The analyzed dataset consists of 9000 historical measurements from a depropanizer column. This dataset represents the production of one week in January 2022 with 1 minute time period. The measurements include 12 process variables. The vector m of these variables is given as:

$$m = (F, L, D, B, Q, P_{\text{b}}, P_{\text{t}}, T_F, T_D, T_B, T_{\text{t}}, T_{\text{b}}), \tag{3.2}$$

where F is the feed flow rate, L is the reflux flow rate, D is the distillate flow rate, B is the bottom flow rate, Q is the united heat duty of two reboilers, P_{b} is the pressure at the bottom, P_{t} is the pressure at the top, T_F is the temperature of the feed, T_D is the temperature of the distillate, T_B is the temperature of the bottom, the temperature at the top of the column T_{t} and temperature at the bottom of the column T_{b}.

3.2 First-principles model of depropanizer

The depropanizer column was modeled [26] using the gPROMS modeling software, which is renowned for its ability to handle complex process modeling and simulation. This software offers several benefits, including a user-friendly interface, robust modeling capabilities, the ability to handle process discontinuities, and versatility in solving a diverse range of mathematical models. gPROMS offers two methods for creating models: one is by defining mathematical equations using the gPROMS modeling language, and the other is by using the Process Modeling Library (PML) to create the models [1]. The depropanizer model was specifically created using PML, a library that contains pre-defined models for process equipment. By using pre-defined models, users can save time since these models have already undergone testing and validation.

The model was developed based on the P&ID diagram and technical documentation of the actual process [26], aiming to achieve the highest possible resemblance to reality. Nonetheless, constructing a model of a real-world process from a plant is a challenging

task, so some approximations and differences were necessary. For instance, the real depropanizer has two reboilers, while the gPROMS model includes only one.

The PML library was used in the model of the depropanizer and included several important objects such as column section, flash drum, source/sink, valve liquid, and pump simple are shown in Figure 3.2. This figure shows a distillation column model without its control configuration due to a non-disclosure agreement. Column section is a model of a section of a distillation column and can be used to model both tray and packing sections. Flash drum serves a dual purpose as it can be used as a reboiler to provide heating to the column, or as a condenser to remove heat from a process and condense the vapor into liquid. Source/sink is a model of a feed or product stream that can be used to represent both inputs and outputs of a process. The valve liquid model is designed to control the flow of liquid in a stream by adjusting a valve. This valve uses the flow coefficient and the position of the valve stem to regulate the flow rate of the liquid, based on the pressure difference. Pump simple is a model of a simple positive displacement pump that can be used to increase the pressure of a liquid stream. To define the physical properties of the mixture, the Multiflash property package was used, which provides a database of chemical components [26]. The measurement device is used to measure the composition of the mixture.

Table 3.1: List of equipment shown in the Figure 3.2.

Equipment	Description
1	Column section
2	Flash drum
3	Source
4	Sink
5	Pump Simple
6	Valve liquid
7	Valve non-return
8	Measurement device

3.2.1 Model validation

The process of designing a simulation model is not complete until the model has been validated against real data from the actual process. For this purpose, we used historical data from step tests that were performed on feed flow rate F in the depropanizer. This data allowed us to gain insight into the dynamic behavior of the process and compare it to the simulation results.

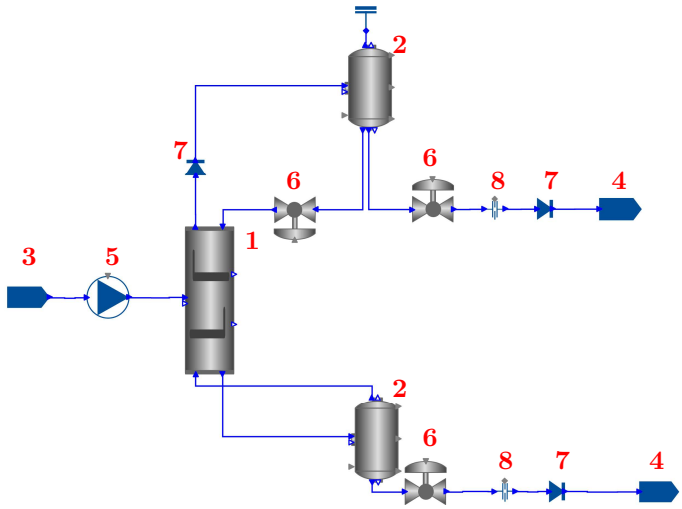

Figure 3.2: Scheme of gPROMS model of depropanizer.

To accurately simulate the real process, the initial conditions of the simulation model were specified to match the real measurements. The initial conditions included the feed temperature T_F, column top P_t and bottom P_b pressure, distillate T_D and bottom product T_B temperature, and the composition of the feed, distillate, and bottom. Although the composition measurements from the lab were not taken directly at the start of the step tests, they were obtained the day before. It is worth noting that composition measurements are not easily obtained continuously using online analyzers, but rather infrequently (once a month or a week).

To ensure consistent input data for the model, specifically the feed flow rate, we used the goRUN library that is available within the gPROMS Modelbuilder. Although the documentation mentions a direct link between gPROMS and MATLAB, we discovered, after consulting with PSE support (the company responsible for gPROMS), that version 7.1.1 of ModelBuilder does not support the goMATLAB feature, which required us to use the link between gPROMS and Excel, known as goRUN. In Appendix B, it is outlined a detailed guide on how to load Matlab data into an Excel file, and how to use the goRUN library to communicate with gPROMS. Additionally, Appendix B also provides instructions on how to automate the loading and saving of simulation data via Excel.

The other settings and initial conditions for the individual device models and the parameters of the PID controllers involved in the control loops were set using a stepwise, iterative procedure. Our objective was to obtain simulation results that closely match the real data. This approach helped us enhance the accuracy of the simulation model and better understand the dynamic behavior of the depropanizer.

Figure 3.3 presents feed flow rate F and a comparison of the temperature at the bottom of the column obtained from simulation $T_{\mathrm{b,sim}}$ and real process data T_b. It should be noted that both temperatures are normalized due to the non-disclosure agreement. However, there is a noticeable difference of 7.8°C between the real temperature measurements and simulation data. Although we made an effort to minimize this difference, it was not entirely possible. This discrepancy may be attributed to the different behavior of the real column or the composition of the mixture. Another possible factor could be the presence of two reboilers in the real process, while the gPROMS model only has one. This could be causing the temperature at the bottom T_b to be lower due to the lesser effect of the reboiler.

Our objective was not to focus on the model error deviation but rather to optimize the gPROMS model parameters to achieve a similar dynamic response of the output variable T_b as the real measurements. We observed in 3.3 that an increase in feed

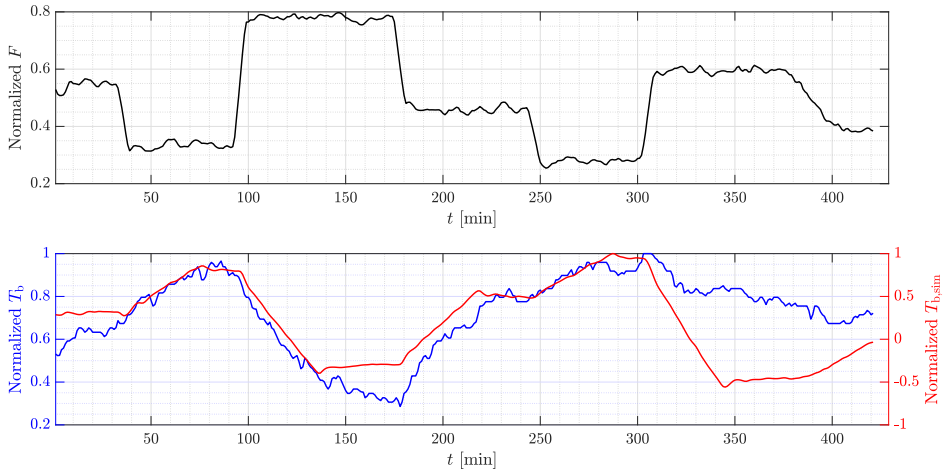

Figure 3.3: Step tests. Feed flow rate F (black), the temperature at the bottom of the column - real T_b (blue), simulated $T_{b,sim}$ (red).

flow rate F resulted in a decrease in temperature at the bottom of the column T_b, and vice versa. A total of five step changes were performed. We observed that the model sufficiently described the dynamics and behavior of the real variable during the first four step changes. However, during the final step change, we noticed that the temperature in the simulation model decreased much faster than in the real case. Due to the distillation column's nonlinear and multivariate nature, this deviation could be caused by the influence of other variables in the column, such as pressure.

3.2.2 Results

Once the model validation was completed, we proceeded to simulate data by initializing the initial conditions with real measurements and feed, distillate, and bottom compositions with the laboratory analysis conducted 3 days prior. The input data in the form of feed flow rate was loaded into the simulation using the method described in the Appendix B. A total of 9000 input data points were loaded, and the simulation took 93 minutes to complete. After conducting several experiments, we discovered that the simulation time is primarily influenced by the size of the dataset and the number of input variables to be configured. Using the go:RUN component to execute simulation results in the creation of an object when input data is loaded. The larger this object, the longer the simulation takes to run. This issue could be resolved by devising a method for streaming data directly into gPROMS ModelBuilder.

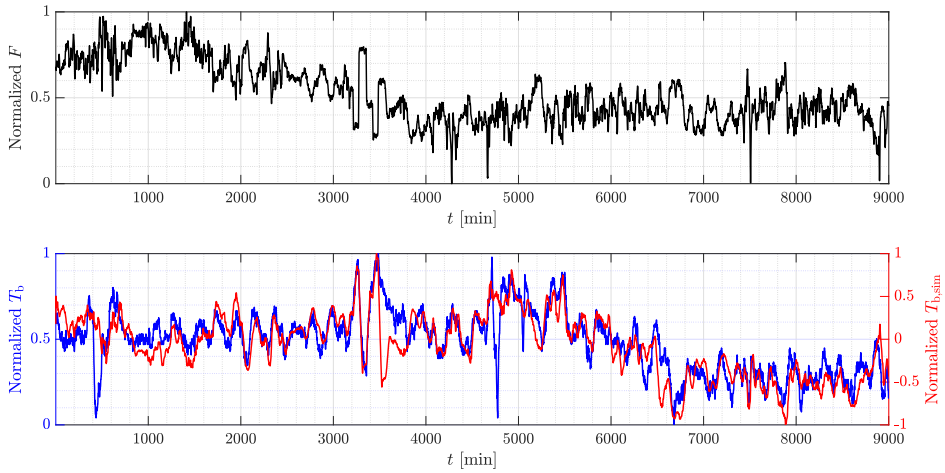

Figure 3.4: Perfomance of gPROMS model. Feed flow rate F (black), the temperature at the bottom of the column - real T_{b} (blue), simulated $T_{\mathrm{b,sim}}$ (red)

.

Figure 3.4 shows the feed flow rate F and a comparison between the simulated $T_{\mathrm{b,sim}}$ and real temperature T_{b} at the bottom of the column. Figure 3.4 also demonstrates how well the gPROMS model performs when using the whole dataset and it was generated using MATLAB `yyaxis` function to compare the prediction of the gPROMS model with real data. The average error is 7.8 °C, but normalized temperature values are presented due to a non-disclosure agreement. We can also observe that the trend of an increase in feed flow rate F resulting in a decrease in temperature at the bottom of the column T_{b}, and vice versa, was consistent with the observations made during step tests discussed in Section 3.2.1.

3.3 Data-based modeling

Data-based modeling involves several steps, including cleaning data from outliers, correlation analysis, and selecting an appropriate model for predicting the output variable. In this thesis, the output predicted variable is the temperature at the bottom of the column T_{b}. We decided to choose an ARX model as our data-based model, which uses past values of temperature T_{b} and input variables values for prediction. However, due to the presence of noise or outliers in the plant data, this can be challenging. We

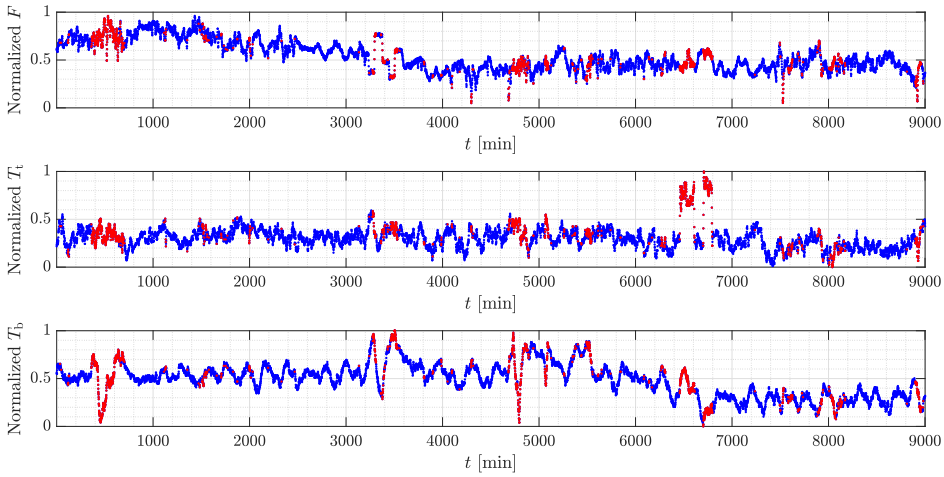

Figure 3.5: The visualization of the detected outliers (red points) and the retained measurements (blue points) in the normalized data points of the temperatures using MCD method.

outline the possible procedure for data-based modeling, including correlation analysis, order determination, and selecting the best model.

3.3.1 Data treatment analysis

A situation often occurs in the industry when a sensor malfunctions or there is a shutdown in plant. We decided to use three methods to detect these values, which may deviate from the other data points.

The application of the MCD method involves controlling the h parameter. In our case, we set this parameter so that it preserves 99% of the data points in the case of a dataset with a normal distribution. However, since we are working with data that comes from industry, we can assume that the percentage composition of detected outliers will be larger. Due to the presence of randomness when using the MCD method we ran the algorithm 30 times and averaged the results from each iteration. The number of detected outliers is 1245, which represents almost 14% of the original 9000 data points. Figure 3.5 shows detected outliers highlighted in red. The first interesting area of detected outliers is located around the time $t = 500$ min, where the temperature in the upper part of the column T_t and feed flow rate F do not change their character. On the other hand, the temperature in the bottom part T_b suddenly decreased and the

MCD method detected this area where the state of the column changed. An interesting area is the implementation of step tests around time $t = 3300$ min, in which the MCD method also identified outliers because there was a change in the system. On the one hand, the MCD method is accurate as it detects significant changes in the nature of the data. On the other hand, it should be noted that data from step tests are desired, and the researcher should be cautious when excluding them. In the region around $t = 6700$ min, we can notice a sudden change in temperature in the upper part of the column, while in the case of feed flow rate F there is no visible change compared to the overall dynamics. These observations confirm the good properties of the MCD method for detecting outliers in a multivariate dataset.

As the second method for detecting outliers, we used the Gaussian mixture model method with 2 Gaussian components. To determine how well each observation fits the estimated mixed distribution, we calculated the log-likelihood of each observation under the identified model. We set a threshold corresponding to the 1^{st} percentile of log-likelihood values. Outliers were detected as values lower than this threshold. Using this approach, we detected 450 values. In Figure 3.6, the areas detected by this method can be observed. Interestingly, this method also detected an area around time $t = 500$ min as a potential change in the system. In cases, where we are looking for areas where the system undergoes rapid changes, it may be problematic that this method did not detect many outliers in the region around time $t = 6500$ min, despite there being noticeable changes in the system. This suggests that the method may not be as sensitive to certain types of changes as we would desire.

When using the SHV method to detect outliers, we followed the procedure provided in Section 2.2.1 Using this method, we identified only 389 outliers. In Figure 3.7, we can notice the detected areas highlighted in red. Using this method, we identified the area around time $t = 500$ min, as well as when using the previous two methods. We also observe detected outliers in the area around time $t = 6700$ min due to the temperature change in the temperature at the top of the column T_{t}.

3.3.2 Correlation analysis

The distillation column, depropanizer, belongs to the group of systems that have strong interactions between the variables. Therefore, it is important to analyze the correlation in time-series data to provide better insight into the dataset. In addition, the presence of control loops in the system can further complicate the analysis of correlations.

Our interest is to study the relationship between the output variable T_{b} and other variables. In this part, we used the remaining data points from data treatment using

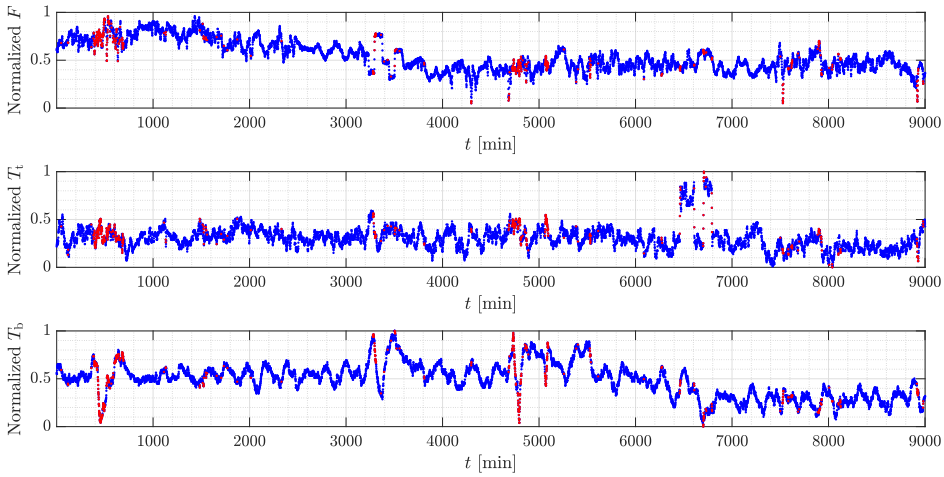

Figure 3.6: The visualization of the detected outliers (red points) and the retained measurements (blue points) in the normalized data points of the temperatures using GMM method.

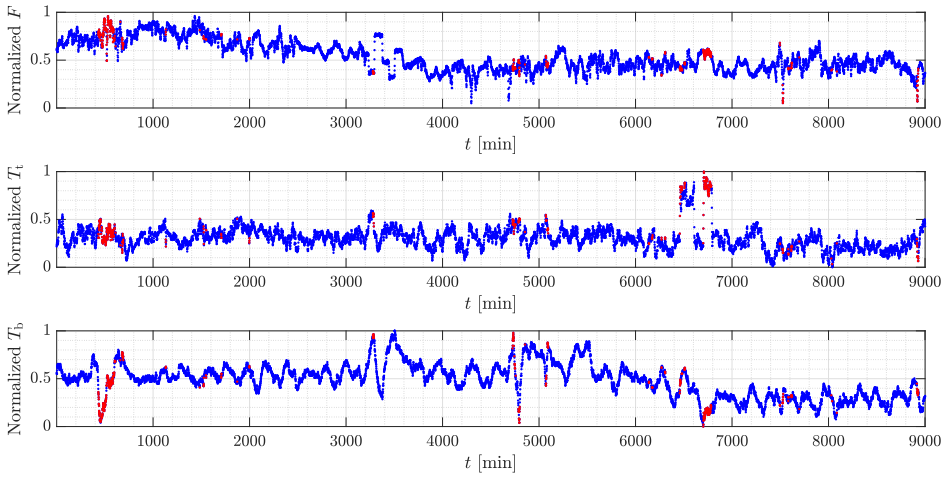

Figure 3.7: The visualization of the detected outliers (red points) and the retained measurements (blue points) in the normalized data points of the temperatures using SHV method.

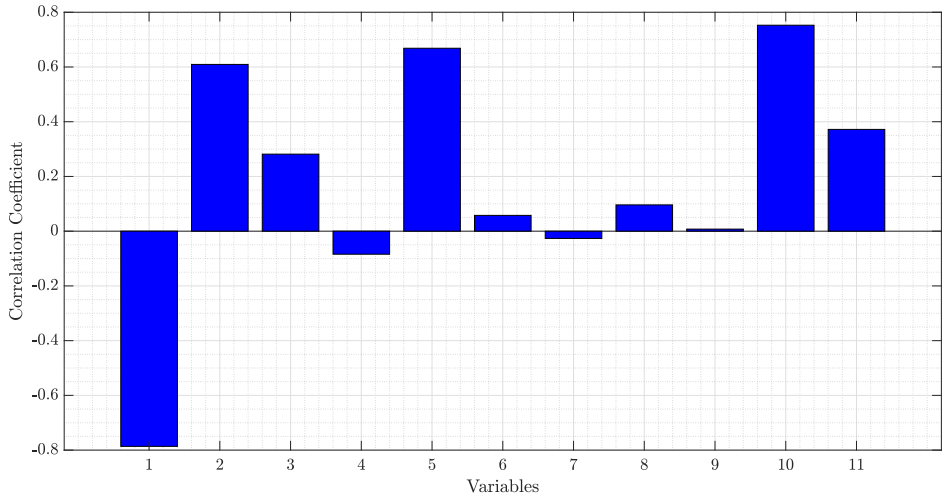

Figure 3.8: Bar graph of correlation coefficients between the variables with output variable T_{b}.

the MCD method. We calculated the values of the Pearson coefficient. The calculated values of the coefficients are summarized in Table 3.2. Figure 3.8 shows a bar graph to visualize the correlation coefficients, where the numbers on the x-axis correspond to the numbers in Table 3.2 representing the variables.

From the results, we can observe that the largest correlation based on the correlation coefficient -0.7863 is with the feed flow rate F. A negative value gives us a negative correlation, which in practice means that increasing the flow leads to a decrease in the temperature T_{b}. Increasing the feed flow rate F may cause the mixture to not be in the column long enough to absorb the sufficient amount of heat required to bring the mixture to the desired temperature. This observation can also be seen in Figure 3.3.

We can also see a medium strong correlation with the bottom temperature T_B with a coefficient value of 0.7527. This correlation is understandable mainly because both temperatures are closely affected by the amount of heat that is supplied or removed from the column, which in this case is represented by the heat duty in the reboiler Q with a correlation coefficient of 0.6685. The purpose of the reboiler is to produce and supply heat to the lower part of the column to provide the energy for the separation of the mixture. If more heat is supplied by the reboiler, a greater amount of vapor is produced with higher temperature than the liquid mixture. Vapor emerges from the reboiler and rises up the column. When it collides with a cooler liquid and vapor in

Table 3.2: Correlation coefficients between the variables with output variable T_{b}.

Variable Number	Input Variable	Correlation Coefficient
1	F	-0.7863
2	L	0.6094
3	D	0.2816
4	B	-0.0842
5	Q	0.6685
6	P_b	0.0577
7	P_t	-0.0265
8	T_F	0.0961
9	T_D	0.0076
10	T_B	0.7527
11	T_{t}	0.3719

the column, it condenses and transfers heat to the column. This heat transfer causes an increase in both T_B and T_{b} temperatures.

We also observe the moderately positive correlation with reflux L with a coefficient equal to 0.6094. This relationship might be causal, as an increase in the temperature at the bottom T_{b} could lead to an activation of more reflux to cool down the column, resulting in an increase in the reflux flow rate. However, this phenomenon may not be valid for all types of columns and depends on the specific process or control configuration of the column. For the other variables, we observe a low correlation due to the values of the correlation coefficient close to zero. The correlations with column pressures p_b and p_t are low, probably because they are controlled variables. The low values of correlation coefficients of distillate flow rate D and T_D indicate, that they are independent of T_{b} as they are related to the cooling setting in the condenser. What is interesting, is the low values of correlation with the temperature of feed T_F and at the top of the column T_{t}. This observation can be caused by the presence of control loops in the system.

3.3.3 Order determination

Determining the order of an autoregressive model is not straightforward and may involve using multiple techniques and methods. When using the ARX model, we must realize that it is not only necessary to determine the order of the output (autoregressive) part, but also the order of the input (exogenuos) part.

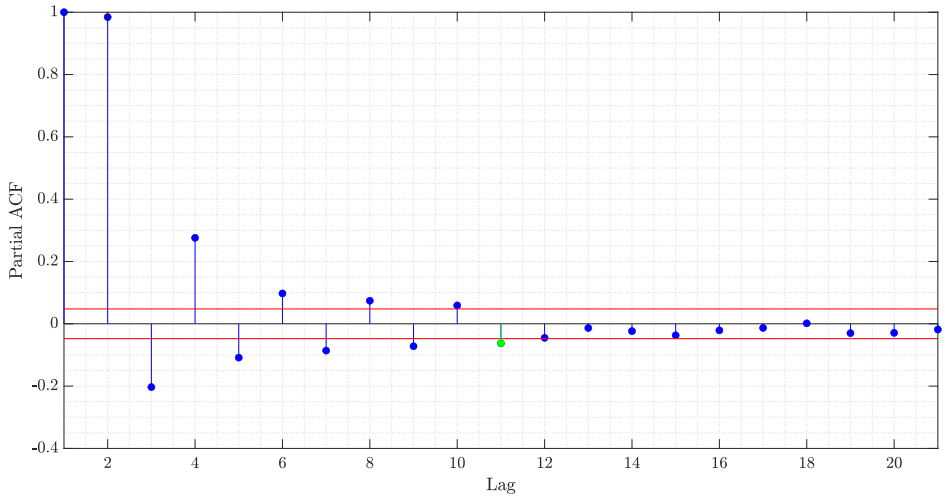

Figure 3.9: Partial autocorrelation function of T_b.

One of the methods to determine the order of the model is to use the partial autocorrelation function. Visualization of this function gives us a closer look at the relationship between the samples and their past values. In our case, the output variable is the temperature T_b, for which we calculated the PACF and displayed its dependence in Figure 3.9. This figure shows PACF values at different PACF values. The PACF value at lag 0 is equal to 1 because it represents the correlation between the same value. Horizontal lines show the 99% confidence interval. The highest possible order of the model is the one that still protrudes from this interval. In our case, this value occurs at a lag of 11. The order of the exogenous part must be lower than or equal to the order of the autoregressive component, however, determining this is often the result of performing several experiments of fitting the ARX model and subsequently selecting the best order based on the chosen criterion.

3.3.4 Model training and validation

To preprocess data for model training, we first adjust our variables to define the deviation variables. It means that we subtract the steady state value from the individual variables. There are many methods for determining the steady state, but in complex processes such as a distillation column, determining the steady state can be very difficult and the system may have multiple steady states. In our case, we determined the steady state in the column corresponding to the value at time

$t = 1000$ min seen in Figure 3.5. Subsequently, we divided our dataset into 3 sub-datasets - training, validation, and testing in a ratio of 50:15:35. In this thesis, we considered only one steady state, but another possibility of data preprocessing could be, for example, the definition of a steady state in each sub-dataset.

Choosing the optimal model is an iterative process. In Section 3.3.3 we determined the maximum order of the output part of ARX model that is 11. We create a loop in which we go through all combinations of the order of the output and input part of the model, with the condition that the order of the input part n_b is less than or equal to the order of the output part n_a is less than or equal to 11. In each iteration of the cycle, we train ARX model with the given values of the orders of the output and input parts. We validate the trained models on the validation data and calculate their AIC values, which represent a measure of how well the model fits the data while taking into account the complexity of the model. After all the iterations are completed, we select the ARX model with the lowest AIC value on the validation data as the best model of the series. This approach allows us to select the model that strikes the best balance between accuracy and simplicity.

In Section 3.3.1, we cleaned the data from outliers. Now the question arises of how to deal with the missing data. One solution would be to replace this data with, for example, interpolation techniques or a median value. In this thesis, we consider two scenarios. In the first scenario, we use only the data left after cleaning by individual methods to train the model. The second scenario includes the original data with outliers.

Scenario 1: Model training without outliers

In this scenario, we assume that we have all three sets - training, validation, and testing - cleaned of outliers. However, the dataset, in this case, contains missing data, which the ARX model cannot handle, as it uses the least squares method to estimate parameters, which requires complete data. We split all sub-datasets into several experiments. We introduced the condition that the experiment must contain more than 100 measurements using data treatment analysis, we identified several outliers that are close to each other and between them, there is not much available data to estimate the model parameters. We could also apply such a procedure, for example, if we received information from the operator that there was a sensor failure at a specific time. In our case, we do not have this information, but the areas of detected outliers using the three methods used could be a suitable approximation. Tables 3.3, 3.4, and 3.5 summarize the number of individual experiments with the corresponding number of samples after data treatment using MCD, GMM and SHV methods used for

training. Overall for model training the number of samples for MCD method is 2453, for the GMM method is 3392 and for the SHV method is 4151. For model validation, the number of samples for the MCD is in Table 3.6 and for the GMM and SHV method in Table 3.7. The overall number used for validation for MCD method is 763 and for GMM and SHV method 1197. As we can notice, the numbers of samples used for training and validation are different in each method. It is common to compare the model performance while using the same number of samples that comes into model development. In this scenario, we want to compare the three different cases and how would it affect the model design.

Table 3.3: Number of samples in individual experiments after data treatment using MCD method used for model training.

Experiment	1	2	3	4	5	6	7	8	9	10	11
Samples	231	408	220	117	122	159	138	136	665	287	370

Table 3.4: Number of samples in individual experiments after data treatment using GMM method used for model training.

Experiment	1	2	3	4	5	6	7	8	9	10	11
Samples	117	239	419	350	101	130	317	947	286	484	202

Table 3.5: Number of samples in individual experiments after data treatment using SHV method used for model training.

Experiment	1	2	3	4	5	6	7	8
Samples	432	438	350	155	162	124	1278	1212

Scenario 2: Model training with original data

In the second scenario, we use all historical data to train and validate the model. The total number of data used for training is 4500 and for validation 1350. The advantage of the ARX model is that it also allows us to create a MISO model, which means that multiple inputs can be used to predict the output. Adding more inputs to the ARX model can lead to several benefits. By adding more relevant inputs that are correlated with the output, we can gain additional information for modeling the relationship between inputs and output. In Section 3.3.2 we obtained information about the correlation of other variables in the dataset with the output variable. We found that the best correlation was observed in the case of feed flow rate F, which we used as input in the SISO ARX model. To improve this model, we decided to add another input - heat duty Q. With this, we created a second model that contains two

Table 3.6: Number of samples in individual experiments after data treatment using MCD method used for model validation.

Experiment	1	2	3	4	5
Samples	182	117	131	101	232

Table 3.7: Number of samples in individual experiments after data treatment using SHV and GMM methods used for model validation.

Experiment	1	2	3
Samples	226	216	755

inputs - feed flow rate F and heat duty Q. In addition, we also identified that reflux L has a medium correlation with the output variable. Therefore, we decided to create a third model, which includes three inputs - feed flow rate F, heat duty Q and reflux L. In this case, we extended the basic ARX model to a MISO model, where we used multiple inputs to predict a single output.

3.3.5 Results

We trained and validated six different models, which are summarized in Table 3.8. In each case, the term ARX is followed by a set of numbers enclosed in parentheses. The first number represents the order of the autoregressive part, that in this case represents the T_b. The other numbers refer to the orders of the inputs. For instance, $\mathrm{ARX(9,8)_{MCD}}$ indicates that the model has an autoregressive order of 8 and uses the feed flow rate F as the only input. The other models use different combinations of input variables, as specified in Table 3.8.

Table 3.8: Summary of ARX models and inputs.

Model	Inputs
$\mathrm{ARX(9,8)_{MCD}}$	F
$\mathrm{ARX(11,7)_{GMM}}$	F
$\mathrm{ARX(8,5)_{SVH}}$	F
ARX(11,9)	F
ARX(9,7,4)	F, Q
ARX(9,3,1,4)	F, Q, L

In the first scenario, we obtained three models - $\mathrm{ARX(9,8)_{MCD}}$, $\mathrm{ARX(11,7)_{GMM}}$, $\mathrm{ARX(8,5)_{SVH}}$ with using the retained data from treatment with the corresponding

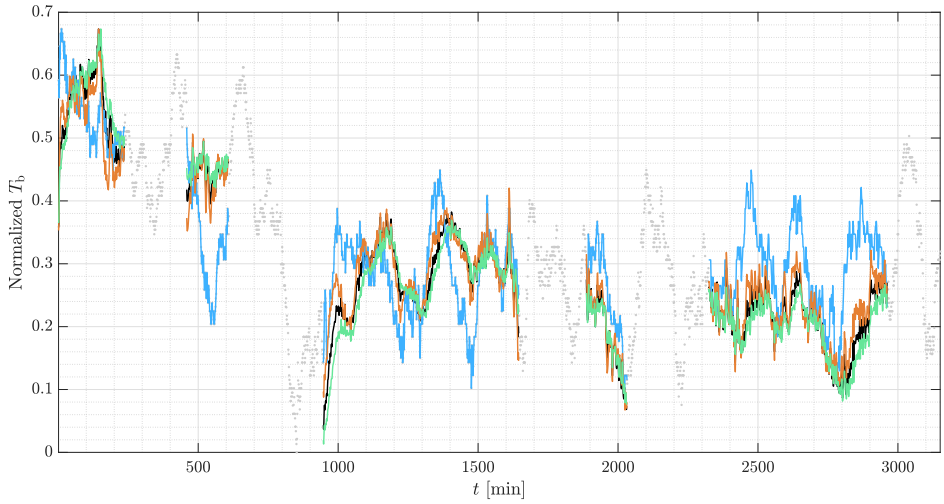

Figure 3.10: Model performance on testing data - ARX(9,8)$_{\text{MCD}}$ (black), ARX(11,7)$_{\text{GMM}}$ (orange), ARX(8,5)$_{\text{SVH}}$ (green) and normalized T_b (light blue).

method. However, to fairly compare these models with each other, we only use the data that is the common intersection of the test data that remained after treatment from all three methods. For the validation dataset, we provide calculated AIC and BIC values of each model in Table 3.9. Based on these results, we can say the lowest value of both criteria suggests selecting ARX(8,5)$_{\text{SHV}}$ as the optimal model. The performance of these models on the testing dataset can be seen in Figure 3.10. Grey dots represent the outliers.

Table 3.9: AIC and BIC values for the validation dataset for the first scenario.

Model	AIC	BIC
ARX(9,8)$_{\text{MCD}}$	3467.0522	3324.5904
ARX(11,7)$_{\text{GMM}}$	3486.4870	3545.0026
ARX(8,5)$_{\text{SHV}}$	2686.0976	2729.1190

The second scenario included the original data for training and validation. The first model ARX(11,9) uses the input variable feed flow rate F as autoregressive models in scenario 1. However, we decided to improve the ARX model by including additional input. The second model ARX(9,7,4) includes, in addition, to the feed flow rate F, the additional input heat duty Q. The last data-based model ARX(9,3,1,4) uses as

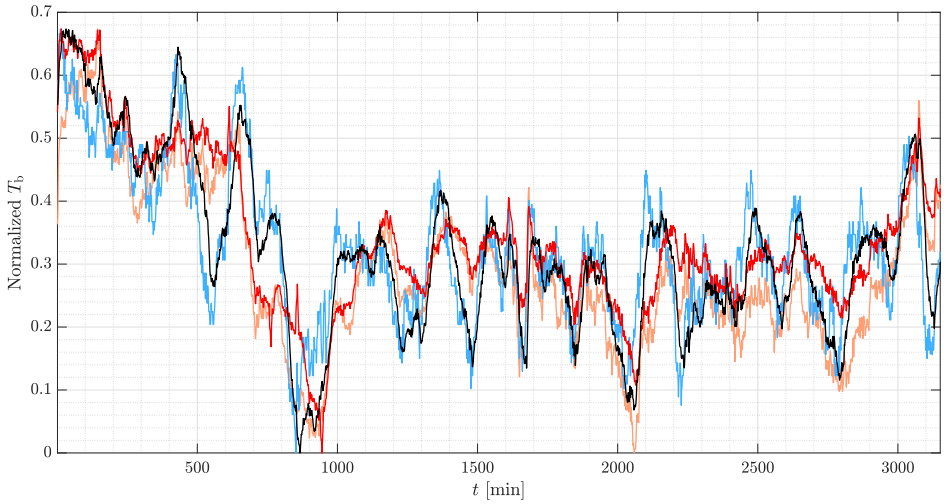

Figure 3.11: Model performance on testing data - ARX(11,9) (orange), ARX(9,7,4) (red), ARX(9,3,1,4) (black) and normalized T_b (light blue).

inputs three variables - feed flow rate F, heat duty Q and reflux flow rate L. The comparison of the performance of these models on the testing dataset can be seen in Figure 3.11. Based on the AIC and BIC values provided for each model in 3.10, we can conclude that the ARX(9,3,1,4) model with the lowest values for both criteria is the optimal choice. It is interesting, that values of AIC and BIC for models ARX(9,7,4) and ARX(9,3,1,4) are significantly lower than for the other trained models. This observation can suggest, that despite the similar number of parameters in all the models, the models ARX(9,7,4) and ARX(9,3,1,4) fit the data better.

Table 3.10: AIC and BIC values for the validation dataset for the second scenario.

Model	AIC	BIC
ARX(11,9)	3499.1347	3556.4349
ARX(9,7,4)	601.5621	647.6975
ARX(9,3,1,4)	403.5048	400.3983

Due to the non-disclosure agreement, we provide the specific parameters of the autoregressive models. To study and analyze the models, we provide the step response of each ARX model seen in Figure 3.12. In other words, Figure 3.12 shows how the ARX model would react to a unit step input change. In the case of $\text{ARX}(9,8)_{\text{MCD}}$,

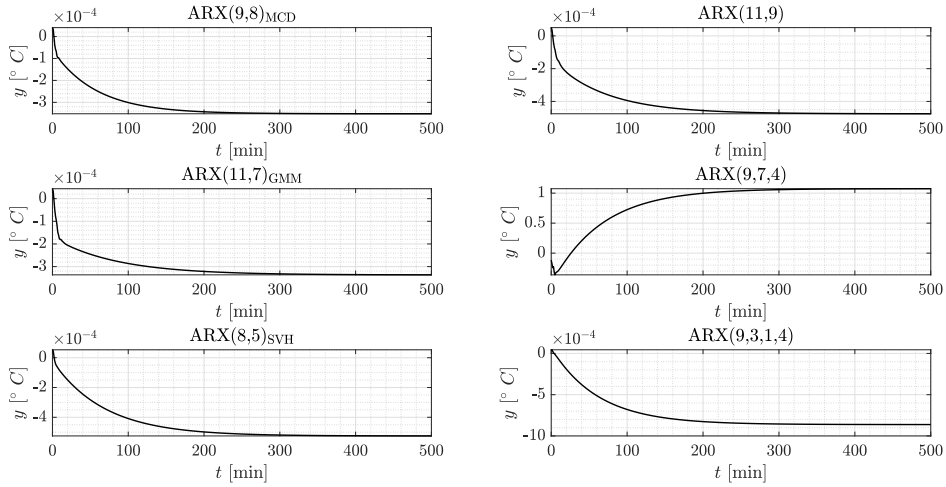

Figure 3.12: Step responses of identified ARX models.

$\text{ARX(11,7)}_{\text{GMM}}$, $\text{ARX(8,5)}_{\text{SHV}}$, and ARX(11,9), the responses show how the step change of 1 kg/h feed flow rate F affects the temperature at the bottom of the column T_{b} in degrees Celsius. We can observe that the temperature decreases in the steady state in the interval of $[-4.7524, -3.5195] \times 10^{-4}$ °C. The response of the ARX(9,7,4) model shows that the temperature changes when both the feed flow rate F and the heat duty Q are changed by 1 kg/h and 1 kJ/h, respectively, simultaneously. After the output response stabilizes, its value will increase by 1 °C. The response of the ARX(9,3,1,4) shows that when the feed flow rate F, heat duty Q, and reflux flow rate L are changed simultaneously by 1 kg/h, 1 kJ/h, and 1 kg/h, respectively, the output decreases in -8.6219×10^{-8} °C. It is worth noting, that all these unit changes are not likely to be present in the process industry. However, these analyses provide us with a better understanding of the dynamics of the system and confirm the stability of the proposed ARX models.

Table 3.11 summarizes the values of the MSE and RMSE criteria for the first scenario. By choosing the lowest values of both criteria, we can state that the best model of the first scenario is $\text{ARX(8,5)}_{\text{SVH}}$. To give an overall comparison between models for the second scenario, the values of calculated criteria are summarized in Table 3.12. The lowest values of all the criteria for ARX(9,3,1,4) represent this model as the best for the second scenario. Considering that both criteria values for this model are significantly lower compared to the models in the first scenario, we can label this model as the best data-based model.

Table 3.11: Comparison of model accuracy on the testing dataset using MSE and RMSE for the first scenario.

Model	MSE	RMSE
$\text{ARX(9,8)}_{\text{MCD}}$	0.1156	0.3401
$\text{ARX(11,7)}_{\text{GMM}}$	0.1166	0.3414
$\text{ARX(8,5)}_{\text{SHV}}$	0.0960	0.3099

Table 3.12: Comparison of model accuracy on the testing dataset using MSE and RMSE for the second scenario.

Model	MSE	RMSE
ARX(11,9)	0.0998	0.3160
ARX(9,7,4)	0.0469	0.2165
ARX(9,3,1,4)	0.0085	0.0923

3.4 Hybrid modeling

The combination of the first-principles model and the data-based model leads to the creation of a hybrid model. This concept takes advantage of both modeling approaches. In this work, we focus on modeling the deviation between the gPROMS first-principles model and real measurements from the plant. Such a modeling approach is also called model error modeling. This approach is similar to the parallel model scheme described in Section 2.3.1, where the data-based model is used for modeling residuals. Our goal is to create an error model that can estimate the deviation between the value measured in the plant y and the value predicted by the gPROMS model y^{FP} as can be seen in Figure 3.13. We assume that if this error model subtracts this estimated error $\hat{e}^{\text{FP}}$ from the real error e^{FP}, the result should be approximately zero. In other words, we want to achieve the correction between the gPROMS model and the real values. We introduce three possible approaches - constant, static, and dynamic correction.

3.4.1 Constant correction

In Section 3.2.2, we discussed that despite the fact that the output from the gPROMS model describes the dynamic trends of real temperature measurements, a deviation is present. In the case of constant correction, a constant value c is added to the outputs from the gPROMS model y^{FP} as can be seen in Figure 3.13. The corrected output $\hat{y}$ can be calculated as:

$$\hat{y} = y^{\text{FP}} + c, \tag{3.3}$$

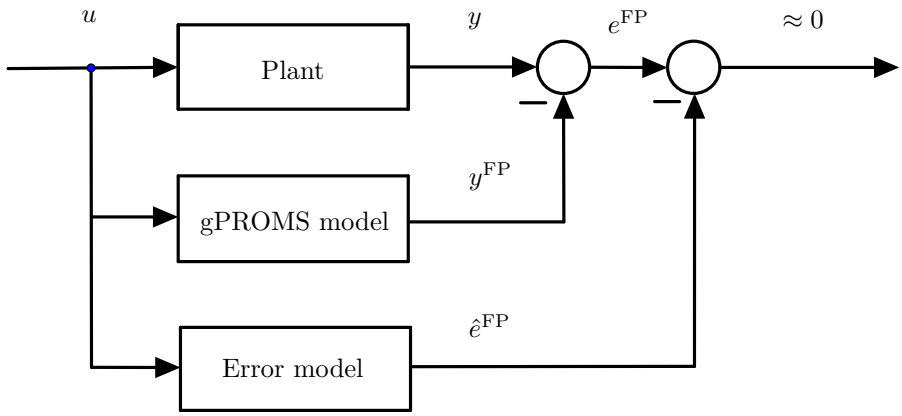

Figure 3.13: Scheme of the hybrid model used for correcting the error.

where c is the constant correction value that is calculated as the mean value of the error values between the real measurements and the values predicted by the gPROMS model from the training dataset given by:

$$c = \frac{1}{N} \sum_{i=1}^{N} \left(y_i - y_i^{\mathrm{FP}} \right), \tag{3.4}$$

where N is the number of data points in the training dataset.

The constant correction of the error is the simplest form of correction, because of its easiest implementation. However, this approach may include disadvantages such as a lack of adaptability and robustness in case of changes in the process dynamics or operating conditions in the depropanizer column.

3.4.2 Static correction

One simple approach in parallel modeling involves using static correction, where the estimated error is represented by the input u multiplied by a constant parameter a. This approach includes solving an optimization problem with the objective of minimizing the error e^{FP} between the gPROMS model output y^{FP} and the real measurements y, by finding the optimal value of a that minimizes the objective function J:

$$J = \sum_{i=1}^{N} \left(a \cdot u_i - e_i^{\mathrm{FP}} \right)^2. \tag{3.5}$$

Corrected outputs are calculated as

$$\hat{y} = y^{\mathrm{FP}} + a \cdot u. \tag{3.6}$$

The static correction approach is a straightforward approach, that does not require complex algorithms. Static correction can lead to improved model predictions, as it can account for information from the input that may not be captured by the gPROMS model. Nevertheless, static correction has limitations and may not be able to capture dynamic changes or include the entire range of operating conditions of the studied process, especially in the case of multiple operating points.

3.4.3 Dynamic correction

Dynamic correction takes into account the error model that can capture the time-varying behavior of the model error e^{FP}. In this approach, the selection of the model is the SISO ARX model, which takes into account previous values of model error as the output and the previous values of input to estimate the current value of the error. To achieve this, we follow the procedure explained in Section 3.3.3 and 3.3.4, where the output is the estimated model error. The advantage of applying this approach involves taking into account the effect of the input u on the model error. In this case, the input variable is feed flow rate F. Dynamic correction approach allows more accurate and adaptive correction than the use of constant or static correction.

3.4.4 Results

The purpose of the hybrid model is to provide better predictions of the temperature at the bottom of the column T_{b} by combining the output of the gPROMS model and correction of its occurred model error. The model error of the gPROMS model can be seen in Figure 3.15.

In constant correction, the model error represents its mean value in the training dataset. The value of this constant is 7.8 °C. By adding this value to the output of the gPROMS model, we obtain a hybrid model with a constant correction. The performance of this model on testing data can be seen in Figure 3.15.

The static correction represents the model error as a static model using multiplying the input by a constant value. The optimal value of the constant a, which minimizes the objective function 3.5 was determined to be 3.2366×10^{-4} using a gradient-based optimization method. The performance of the static correction of the output of the gPROMS model can be seen in Figure 3.15.

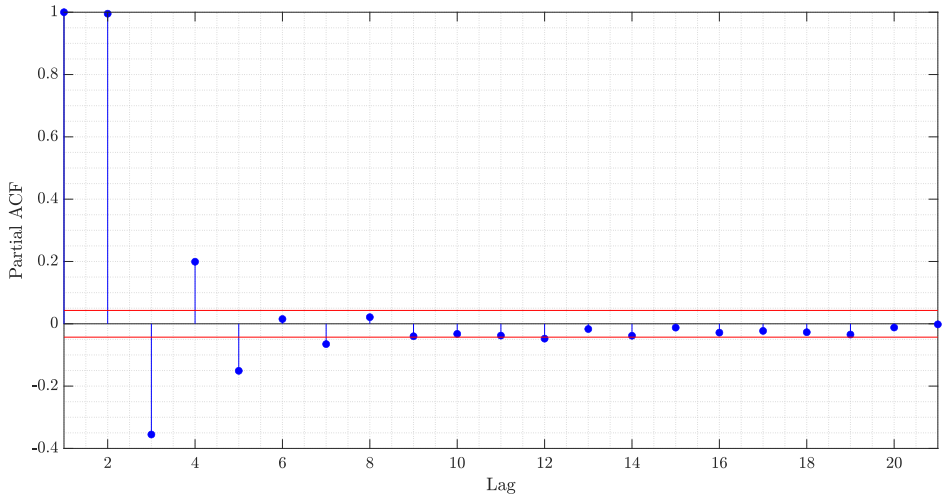

Figure 3.14: Partial autocorrelation function of e^{FP}.

In the dynamic correction, it is needed to predict the model error by the ARX model. Firstly, we determine the maximum order of the autoregressive part, the output part of the model that is the model error. The maximum order is selected from using Figure 3.14 that shows the PACF function of model error. The value is 7, because at this lag still protrudes the 99% confidence interval. After training and validating the different combinations of orders, we selected the best model as ARX(5,2) with input variable feed flow rate F. The prediction of this model error model is shown in Figure 3.15.

Table 3.13: AIC and BIC values for the validation dataset for three correction scenarios.

Correction Scenario	AIC	BIC
Constant	408.0487	410.0241
Static	294.4329	294.4329
Dynamic	955.0727	955.0428

In Table 3.13 we present the calculated AIC and BIC for the validation dataset. The highest values of both criteria are present for the model with dynamic correction. By comparing with the values for the other corrections, we can suppose that this is caused mainly by the presence of more parameters in the dynamic ARX model. While in the case of static and constant correction, there is only 1 parameter, in the case of dynamic correction there are 7 parameters. Table 3.14 summarizes the values of the

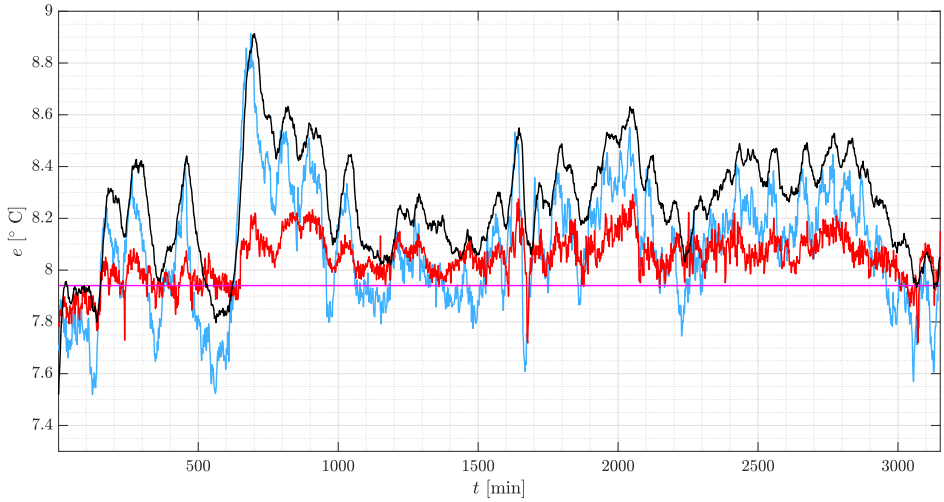

Figure 3.15: Error model (blue) and its estimation by constant (magenta), static (red), and dynamic (black) correction.

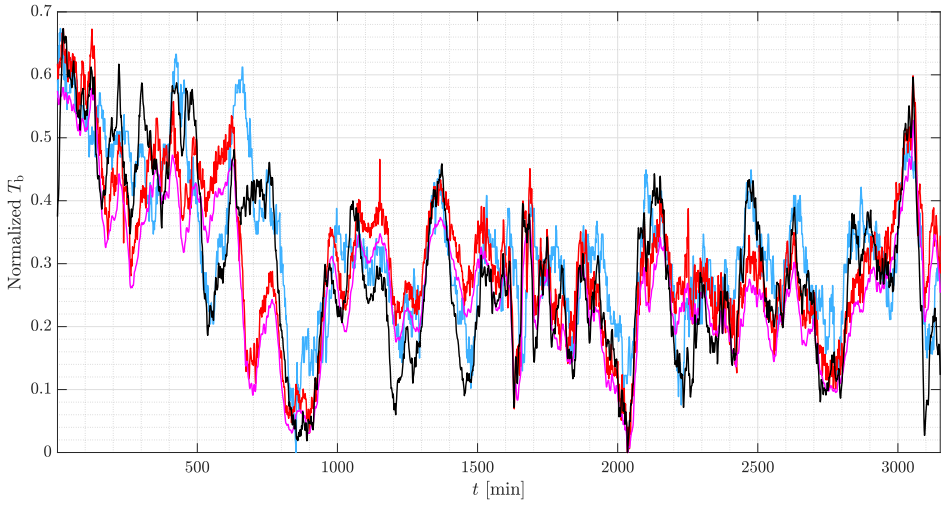

Figure 3.16: Normalized T_b (blue) with corrected outputs using constant (magenta), static (red), and dynamic (black) correction.

Table 3.14: Comparison of model accuracy on the testing dataset using MSE and RMSE for three correction scenarios.

Correction Scenario	MSE	RMSE
Constant	0.1040	0.3021
Static	0.0413	0.2031
Dynamic	0.0343	0.1853

MSE and RMSE criteria considering corrected and real outputs to fairly compare them with the results from the data-based modeling approach discussed in Section 3.3.5. From the results, we can notice that in the case of dynamic correction, the lowest MSE and RMSE values are achieved, which means that this scenario is the most accurate in terms of bias estimation. The values of the criteria for the static correction are somewhere between the values for the constant and dynamic models. The use of static correction could indicate that this approach is a compromise between the simplicity of constant correction and the complexity of dynamic correction. Upon comparing the values of criteria in Tables 3.11, 3.12, and 3.14, it becomes evident that the dynamic and static correction methods outperform all of the data-based models identified in Section 3.3.5. The only exception is the ARX(9,3,1,4) model, which, due to its significantly lower MSE and RMSE values, can be considered the best-performing model. It might be worth considering whether it is practical to use the ARX(9,3,1,4) model with its multiple parameters and three inputs, given that it requires more computation time. Alternatively, using a hybrid model with fewer parameters could be a better compromise.

The Figure 3.16 we can see the corrected outputs using all three correction scenarios. The corrected outputs for constant and static correction scenarios have similar characteristics and represent a solid performance. However, we can observe some areas such as around time $t = 700$ min, that corrected outputs deviate from the real outputs. On the other hand, the corrected outputs using dynamic correction capture the character of the real output data. Furthermore, the dynamic ARX model acts as a filter, resulting in smoother model outputs, as observed around time $t = 1200$-1500 min. This filtering effect can be also seen in Figure 3.15, which means that dynamic correction (ARX model) is smoother than model error.

Chapter 4

Conclusions

This diploma thesis deals with the system identification of the depropanizer, which is part of the Fluid catalytic cracking unit in the refinery Slovnaft, a.s. Three different modeling approaches are being considered, namely first-principles, data-based, and hybrid methods.

First-principles model of the depropanizer column was modeled using gPROMS software, with the aim of achieving the highest possible resemblance to the real-world process. While some approximations and differences were necessary due to the challenging task of constructing a model of a real-world process from a plant, the gPROMS model provides a robust representation of the depropanizer column.

The procedure of obtaining a data-based model included various steps. The first step is data treatment analysis, where three methods are used to detect outliers in the data: Miminum covariance determinant, Gaussian mixture model, and Smallest half-volume method. The Minimum covariance determinant method detected the most outliers, with almost 14% of the original data points being detected. The other two methods detected fewer outliers, but all three methods identified similar areas of change in the system. The next step is correlation analysis, where the correlation between the output variable and other variables is studied. It is found that the temperature at the bottom of the column T_b has a strong positive correlation with the temperature at the top of the column T_t and a negative correlation with the feed flow rate F. The final step is selecting an appropriate model for predicting the output variable. We chose the ARX model to predict the temperature at the bottom of the column T_b. To determine the order of the autoregressive output part, we used PACF analysis. Then we tested all combinations of the order of input and output parts of the ARX model to find the best model based on the AIC criterion. We cleaned the data from outliers and split it into experiments with over 100 measurements in the first scenario. Although we used different numbers of samples for training and validation for each method, we aimed to compare the impact of each data treatment in order to obtaining model.

The last part introduced the hybrid modeling. We used parallel hybrid model, where the data-based model is used to predict and correct the deviations between the first-principles model predictions of temperature at the bottom of the column and its real measured values. Three error correction models were considered - constant, static, and dynamic.

In order to compare the models performance, we calculated the RMSE and MSE criteria. In the first scenario of data-based modeling according to the lowest values of criteria, results show that the best model is $\text{ARX}(8,5)_{\text{SVH}}$. Considering the values of criteria for other models, we can state that the values are similar, which means that in this case, the effect of the different amount of data used for training models is not playing an important role. However, it is worth noting, that all the methods detected outliers at similar positions. Another interesting research could be to try to remove outliers by other methods. The results of the second scenario indicate that the ARX(9,3,1,4) model produces the lowest values of both criteria. This model is characterized by three inputs - feed flow rate F, heat duty in the reboiler Q, and reflux flow rate L - which are strongly correlated with the output. In hybrid modeling, the lowest values of RMSE and MSE mean that the model with dynamic correction has the best fit for the testing data. Interestingly, constant and static corrections also demonstrate sufficient performance. Therefore, it is worth considering whether a model with fewer inputs and parameters would be more sustainable in the long term. Despite the success of data-based models, hybrid models offer the benefit of having fewer parameters, making them simpler and more practical for recursive identification. Therefore, in conclusion, while the ARX(9,3,1,4) model was considered as the best model, it may be beneficial to explore the potential of error correction models for future applications.

This thesis demonstrated how various modeling approaches can be applied to the system identification of a complex chemical system - a distillation column. In future work, it would be interesting to explore the use of different data-based models, such as neural networks. One interesting possibility for future work could be exploring new hybrid modeling approaches, not just for predicting the temperature at the bottom of the column, but also for other areas of the distillation column.

Appendix A

Resumé

V procesnom priemysle často nastáva situácia, kedy je potrebné použiť matematický model procesu. Matematický model predstavuje abstrakciu procesu vo forme matematických vzťahov. Tvorba modelu pre komplexné procesy je často časovo náročná a vyžaduje si dostatočnú znalosť a vedomosti o študovanom procese. Pre získanie matematického modelu je možné použiť niekoľko prístupov.

Prvý prístup sa nazýva mechanistické modelovanie. Tento typ modelovania zohľadňuje fyzikálne a chemické vzťahy, ktoré opisujú daný systém. Mechanistické modelovanie sa využíva aj na tvorbu digitálnych dvojčiat v simulačných programoch. Z toho dôvodu takéto modely našli uplatnenie aj v priemysle, napríklad na tvorbu programov na školenia operátorov v prevádzke. Jednou z nevýhod tohto princípu modelovania je najmä časová náročnosť tvorby modelu.

Ak sú dostupné nameraná dáta z procesu, na tvorbu modelu môže byť použitý princíp dátového modelovania. Z historických dát je možné získať nové poznatky a porozovania o danom procese, čo vedie k tvorbe modelov založených na dátach. Základné rozdelenie dátových modelov sa delí na umelé neurónové siete a regresné modely.

Kombináciou dátového modelu a mechanistikého modelu vzniká hybridný model, ktorý môže mať viacero podôb. Zatiaľ čo model založený na prvých princípoch opisuje základné fyzikálne vlastnosti systému, dátový model môže byť použitý na odhad neurčitých parametrov alebo na korekciu odchýlok medzi reálnymi meraniami a výstupom z mechanistické modelu.

V tejto práci je skúmaným procesom rektifikačná kolóna, konkrétne depropanizér, ktorý slúži na separáciu butánovej zmesi. Depropanizér je časťou jednotky fluidného katalytického krakovania v refinérii Slovnaft, a.s. v Bratislave. Produkt, resp. destilát z prvej časti tejto jednotky, debutanizéra, je nástrekom do depropanizéra. Depropanizér sa skladá z kolóny, kondenzátora a dvoch varákov. Na pochopenie a študovanie tejto

kolóny je poskytovaná technická dokumentácia a P&ID diagram. Na meranie dát sú poskytované online senzory. K dispozícií sú historické dáta z prevádzky z januára 2022 predstavujúce týždeň fungovania prevázky. Dáta obsahujú 9000 meraní s minútovou periódou vzorkovania.

Simulačný model depropanizéra, ktorý predstavuje mechanistický model, bol vytvorený v prostredí gPROMS ModelBuilder. Na validáciu modelu sú použité historické dáta z testov pomocou skokových zmien, ktoré boli uskutočňované na prietoku nástreku do kolóny. Na základe týchto dát je možné pozorovať dynamiku simulačného modelu, pričom v prípade skúmanej teploty v dolnej časti kolóny je prítomná odchýlka 7,8 °C od reálnych meraní.

Dátové modelovanie zahŕňalo viacero krokov. Prvým krokom je detekcia odľahlých hodnôt v údajoch pomocou troch metód: determinant minimálnej kovariancie (MCD), model Gaussovej zmesi (GMM) a metóda najmenšieho polovičného objemu (SHV). Metóda MCD zistila najviac odľahlých hodnôt, pričom sa zistilo takmer 14 % pôvodných údajových bodov. Ďalšie dve metódy odhalili menej odľahlých hodnôt, ale všetky tri metódy identifikovali podobné oblasti zmien v systéme. Na lepšie pochopenie vzťahov medzi premennými v rektifikačnej kolóne je vykonaná korelačná analýza. Hodnoty Pearsonovho koeficienta, ktoré predstavujú numerickú hodnotu korelácie, sú uvedené v tabuľke 3.2. Zistilo sa, že teplota v spodnej časti kolóny T_{b} má silnú pozitívnu koreláciu s teplotou v hornej časti kolóny T_{t} a negatívnu koreláciu s prietokom nástreku F.

Na predpovedanie teploty v spodnej časti kolóny sme zvolili autoregresný ARX dátový model. Tento typ modelu využíva na predpovedanie súčasnej hodnoty predošlé hodnoty vstupov a výstupov. Výstupom a sledovanou veličinou je teplota na dne kolóny. Na vytvorenie ARX modelu je nutné určiť rád modelu. Na určenie poradia autoregresnej výstupnej časti je použitá analýza pomocou parciálnej autokorelačnej funkcie. Trénovanie dátových modelov je rozdelené do dvoch scenárov. Prvý scenár zahŕňa iba merania, ktoré ostali po čistení dát od outlierov tromi metódami. Druhý scenár využíva všetky dostupné dáta. Na návrh ARX modelu, a teda na predikciu výstupu je možné použiť aj viacero vstupov. Výber vstupov bol realizovaný na základe korelačnej analýzy vzťahov výstupu s ostatnými premennými. Zatiaľ čo prvý model druhého scenára zahŕňa ako vstup do ARX modelu prietok nástreku, v prípade druhého modelu je k nemu pridaný aj tepelný výkon varáka. Tretí model zahŕňa okrem týchto dvoch vstupov aj prietok refluxu, teda prietoku destilátu, ktorý sa vracia naspäť do kolóny.

Hybridné modelovanie využíva dátový model na korekciu odchýlky výstupu z gPROMS modelu a reálnych meraní teploty na spodku kolóny. Sú predstavené tri druhy korekcie.

Prvou korekciou je konštantá korekcia, pri ktorej je k výstupu z gPROMS modelu pripočítaná priemerná hodnota odchýlky v trénovacích dátach. Statická korekcia zohľadňuje možný lineárny vzťah prietoku nástreku od odchýlky. V prípade dynamickej korekcie je predstavený ARX model, ktorého vstupom je prietok nástreku a výstupom hodnota odchýlky meniaca sa v čase.

Appendix B

gPROMS data exchange

Preparation of .xls file

1. Copy the `Batch_Reactor.xls` file from the gProms Modelibuilder installation located in the "`examples/gO Product examples/gORUN`" folder and paste it into the desired folder.

2. Rename the `Reactor Schematic` sheet to `InputData`.

3. Delete the content of the `InputData` sheet.

4. Add the variable names and their values to the `InputData` sheet. For example, to load the feed rate values F into the simulation, write the name "Feed" into cell A1 and add the values below it.

5. If the data is already loaded in MATLAB, save it using the provided code.

```
filename = 'path to Excel file';
sheetname = 'InpuData';
% Define input variable, e.g. feed flow rate
F = [1, 2 ,3];
% Define the starting and ending cells of the range
startcell = 'A2';
endcell = ['A' num2str(length(u)+1)];
% Define the range of cells where you want to write
    the data
cellrange = [startcell ':' endcell];
% Write the data to the specified range of cells
xlswrite(filename, F, sheetname, cellrange);
```

Loading data

1. Load data into gPROMS using the `gFO-initial data` sheet.
   (a) In the gFO-initial data sheet, specify the initial data in the Name and Cell X-Ref fields.
   (b) For example, enter `"Feed"` in the Name field and specify the data location in the Cell X-Ref field as `=InputData!A2:A4`.
2. In the ModelBuilder, select the desired process to simulate in the Processes section.
3. In the PARAMETER and SET section, add the following:

```
PARAMETER
RxnData AS FOREIGN_OBJECT "ExcelFO"
F AS ARRAY OF REAL

SET
RxnData := "ExcelFO::path to Excel file";
F := RxnData.Feed;
```

   Note that `RxnData.Feed` must match the variable name in the gFO-initial data sheet (in this example, `Feed`). If the variable name is different, change `[variable name]` to `RxnData.[variable name]`.
4. Click on the "Simulate Process" button in the ModelBuilder environment to execute the simulation.
5. gO:RUN is automatically started and the values are loaded into Process.

Saving data

1. Add the following line to the end of the Process section in ModelBuilder:

```
FPI := "ExcelFP::path to Excel file";
```

2. Use the SEND statement to send data during the simulation. For example, to send the `Temperature` data, use the following code:

```
CONTINUE FOR sigma(v_time)
    SEND
        "Temperature" := bottom_temperature ;
    END
END
```

3. In the gFPI sheet, define the `Tag name` in a predefined table. In this example, the Tag name is Temperature.

4. In the Cell X-ref field in the `gFPI` sheet, define the range in the Results sheet where you want to store the data from gPROMS. For example, =Results!A2 means that the data storage starts from cell A2 in the Results sheet.

5. In the predefined table in the `gFPI` sheet, specify C for column tabulation, and R for row tabulation.

6. Data is stored in real-time during the simulation.

7. After the simulation, load the data into MATLAB using the following code:

```
% Load gPROMS simulation data
filename = 'path to Excel file';
sheetname = 'Results';
data_range = 'A2:A4'; % Change this to match the range
    of your data
sim_data = xlsread(filename,sheetname,data_range);
```

Bibliography

[1] gproms introductory guide. https://usermanual.wiki/Document/gPROMS20ModelBuilder20Guide.1984951121/view.

[2] Martial Amovin-Assagba, Irène Gannaz, and Julien Jacques. Outlier detection in multivariate functional data through a contaminated mixture model. *Computational Statistics & Data Analysis*, 174:107496, 2022.

[3] Keng Cheng Ang. Mathematical modelling and real life problem solving. In *Mathematical Problem Solving*, pages 159–182. World Scientific, 2009.

[4] Tahmineh Azizi, Bacim Alali, and Gabriel Kerr. *Mathematical Modeling: With Applications in Physics, Biology, Chemistry, and Engineering, Edition-2.* B P International, 06 2021.

[5] Michael Barrow. *Correlation and Regression Analysis*, chapter 6. Pearson Education Limited, 4th edition, 2019.

[6] N. Bhutani, G.P. Rangaiah, and A.K. Ray. First-principles, data-based, and hybrid modeling and optimization of an industrial hydrocracking unit. *Industrial & Engineering Chemistry Research*, 51(43):14047–14062, 2012.

[7] Yeongryeol Choi, Bhavana Bhadriaju, Hyungtae Cho, Jongkoo Lim, In-Su Han, Il Moon, Joseph Sang-Il Kwon, and Junghwan Kim. Data-driven modeling of multimode chemical process: Validation with a real-world distillation column. *Chemical Engineering Journal*, 457:141025, 2023.

[8] Piotr Czop, Gabriel Kost, Damian Slawik, and Grzegorz Wszolek. Formulation and identification of first- principle data-driven models. *Journal of achievements in materials and manufacturing engineering*, 44:183, 2011.

[9] William J Egan and Stephen L Morgan. Outlier detection in multivariate analytical chemical data. *Analytical Chemistry*, 70(23):2372–2379, 1998.

[10] J.A Fernández Pierna, F Wahl, O.E de Noord, and D.L Massart. Methods for outlier detection in prediction. *Chemometrics and Intelligent Laboratory Systems*, 63(1):27–39, 2002. Chemometrics 2002 S.I.

[11] M. Fikar and J. Mikleš. *Identifikácia systémov*. Vydavateľstvo STU, Bratislava, Slovakia, 1th edition, 1999.

[12] Holmes Finch. Distribution of variables by method of outlier detection. *Frontiers in Psychology*, 3, 2012.

[13] Angel de la Fuente. *Mathematical Methods and Models for Economists*. Cambridge University Press, 2000.

[14] H. Hanachi, Yu, Kim, Liu, and C.K. Mechefske. Hybrid data-driven physics-based model fusion framework for tool wear prediction. *International Journal of Advanced Manufacturing Technology*, 101(9-12):2861–2872, 2019.

[15] K.M. Hangos and I.T. Cameron. *Process modelling and model analysis*. Academic Press, 2001.

[16] Masoud Hassanpour, Prashant Mhaskar, James House, and Tim I. Salsbury. A hybrid modeling approach integrating first-principles knowledge with statistical methods for fault detection in hvac systems. *Computers & Chemical Engineering*, 142:107022, 2020.

[17] Andreas Himmel, Janine Matschek, Rudolph Kok, Bruno Morabito, Hoang Hai Nguyen, and Rolf Findeisen. Machine learning for process control of (bio)chemical processes, 01 2023.

[18] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. Statistical learning. In *An Introduction to Statistical Learning with Applications in R*, chapter 2. Springer, 2013.

[19] M. Krippl, A. Dürauer, and M. Duerkop. Hybrid modeling of cross-flow filtration: predicting the flux evolution and duration of ultrafiltration processes. *Separation and Purification Technology*, 248:117064, 2020.

[20] Shen Liu, James McGree, Zongyuan Ge, and Yang Xie. 8 - big data from mobile devices. In Shen Liu, James McGree, Zongyuan Ge, and Yang Xie, editors, *Computational and Statistical Methods for Analysing Big Data with Applications*, pages 162–163. Academic Press, San Diego, 2016.

[21] Yi Liu, Yu Liang, Zengliang Gao, and Yuan Yao. Online flooding supervision in packed towers: An integrated data-driven statistical monitoring method. *Chemical Engineering & Technology*, 40(11):436–446, 2017. Citations: 12.

[22] Pete Loucks, Eelco Beek, Jery Stedinger, Jozef Dijkman, and Monique Villars. *Water Resources Systems Planning and Management: An Introduction to Methods, Models And Applications*. Springer Cham, 01 2005.

[23] William L. Luyben. Control of a two-pressure distillation column. *Journal of Process Control*, 92:288–295, 2020.

[24] Muddu Madakyaru, Fouzi Harrou, and Ying Sun. Improved data-based fault detection strategy and application to distillation columns. *Process Safety and Environmental Protection*, 107:22–34, 2017.

[25] Emad A. Mohammed, Christopher Naugler, and Behrouz H. Far. Chapter 32 - emerging business intelligence framework for a clinical laboratory through big data analytics. In Quoc Nam Tran and Hamid Arabnia, editors, *Emerging Trends in Computational Biology, Bioinformatics, and Systems Biology*, Emerging Trends in Computer Science and Applied Computing, pages 577–602. Morgan Kaufmann, Boston, 2015.

[26] M. Mojto. Advanced process control of a depropanizer column. Master's thesis, ÚIAM FCHPT STU v Bratislave, Radlinského 9, 812 37 Bratislava, 27. 05. 2019 2019.

[27] David Nettleton. *Commercial Data Mining: Processing, Analysis and Modeling for Predictive Analytics Projects*. Morgan Kaufmann, 03 2014.

[28] Ventsislav Nikolov. Autoregressive model order determination, 09 2018.

[29] Jeong Hoon Oh, Junyong Lee, Jong-Hyeon Jeong, and Jae Kwang Kim. Bayesian information criterion accounting for the number of covariance parameters in mixed effects models. *Communications for Statistical Applications and Methods*, 27(3):301–311, 2020.

[30] C.C. Pantelides and J.G. Renfro. The online use of first-principles models in process operations: Review, current status and future needs. *Computers & Chemical Engineering*, 51:136–148, 2013. CPC VIII.

[31] Dimitris C Psichogios and Lyle H Ungar. A hybrid neural network-first principles approach to process modeling. *AIChE Journal*, 38(10):1499–1511, 1992.

[32] SJ Qin and TJ McAvoy. A data-based process modeling approach and its applications. In *Dynamics and Control of Chemical Reactors, Distillation Columns and Batch Processes*, pages 93–98. Elsevier, 1993.

[33] Anders Rasmuson, Bengt Andersson, Louise Olsson, and Ronnie Andersson. *Mathematical Modeling in Chemical Engineering*. Cambridge University Press, 2014.

[34] Carlos Rodriguez, Prashant Mhaskar, and Vladimir Mahalec. Linear hybrid models of distillation towers. *Computers & Chemical Engineering*, 171:108160, 2023.

[35] Brian Roffel and Ben Betlem. *Process Dynamics and Control: Modeling for Control and Prediction*. John Wiley & Sons, 2007.

[36] Joel Sansana, Mark N. Joswiak, Ivan Castillo, Zhenyu Wang, Ricardo Rendall, Leo H. Chiang, and Marco S. Reis. Recent trends on hybrid modeling for industry 4.0. *Computers & Chemical Engineering*, 151:107365, 2021.

[37] Artur M. Schweidtmann, Jana M. Weber, Christian Wende, Linus Netze, and Alexander Mitsos. Obey validity limits of data-driven models through topological data analysis and one-class classification. *Optimization and Engineering*, 23:855–876, 2022.

[38] Ray Sinnott and Gavin Towler. Chapter 5 - piping and instrumentation. In Ray Sinnott and Gavin Towler, editors, *Chemical Engineering Design (Sixth Edition)*, Chemical Engineering Series, pages 215–273. Butterworth-Heinemann, sixth edition edition, 2020.

[39] James G. Speight. 4 - distillation. In James G. Speight, editor, *The Refinery of the Future (Second Edition)*, pages 130–132. Gulf Professional Publishing, second edition edition, 2020.

[40] Masashi Sugiyama. Chapter 15 - maximum likelihood estimation for gaussian mixture model. In Masashi Sugiyama, editor, *Introduction to Statistical Machine Learning*, pages 157–168. Morgan Kaufmann, Boston, 2016.

[41] Tarno Tarno, Suhartono Suhartono, Subanar Seno Saleh, and Dedi Rosadi. New procedure for determining order of subset autoregressive integrated moving average (arima) based on over-fitting concept, 09 2012.

[42] A.Y.-D. Tsen, Jang, Wong, and B. Joseph. Predictive control of quality in batch polymerization using hybrid ann models. *AIChE Journal*, 42(2):455–465, 1996.

[43] Xue Ying. An overview of overfitting and its solutions. *Journal of Physics: Conference Series*, 1168:022022, 02 2019.

[44] Hong Zhao, John Guiver, Ramesh Neelakantan, and Lorenz.T. Biegler. A nonlinear industrial model predictive controller using integrated pls and neural net state-space model. *Control Engineering Practice*, 9(2):125–133, 2001.