# Master Thesis

# DEMONSTRATION OF A HOT MELT EXTRUSION PROCESS WITH SIPAT

Daniel Markl

---

Institute for Technical Informatics
Graz University of Technology


Supervisor:
Ao. Univ.-Prof. Dipl.-Ing. Dr. techn. Eugen Brenner
Institute for Technical Informatics

Technical Advisor:
MMag. Dr. Daniel M. Koller
Research Center Pharmaceutical Engineering GmbH

Technical Advisor:
Dipl.-Ing. Barbara Kavsek
Siemens AG, Corporate Technology

Graz, 5 March 2012

## Abstract

Full quality control of pharmaceutical materials and manufacturing processes using *Process Analytical Technology* (PAT) tools guarantees a final target product profile. Continuous real-time process verification can be accomplished by monitoring univariate (temperature, pressure, etc.) and multivariate (spectrum, image, etc.) process and performance attributes. Following the growing industry activity concerning PAT, Siemens developed the PAT software solution SIPAT. SIPAT allows to monitor and identify *Critical to Quality Attributes* (CQA) to control and finally to optimise the manufacturing process. The data is collected and timely aligned in real-time. This functionality paves the way for innovative continuous manufacturing processes in the pharmaceutical industry, where different unit operations e.g., continuous granulation, *Hot Melt Extrusion* (HME), tableting, etc. can be aligned.

In this context, this PAT approach based on a SIPAT installation is applied on a HME process (Coperion ZSK18) at the *Research Center Pharmaceutical Engineering* (RCPE), where data from generic sensors and from a *near-infrared* (NIR) spectrometer (Sentronic SentroPAT FO with Dynisco probe) were monitored in real-time. SIPAT aligns the spectrum of the extrudate (at the die) and scalar measurements (screw speed, barrel temperatures, material pressure, etc.) and provides a common database. The user or external software (i.e. Matlab, Simca-Q) can access the measured data from the SIPAT database, which enables process control out of SIPAT.

A discussion of the extrusion process in general and the processing unit in detail as well as process parameters is essential to analyse the process. A good process understanding is required in order to develop a model of the HME process. Various different models are discussed and several validation methods were applied to finally choose one model for simulation and one for predictions. Furthermore, a real-time predictor which combines the extruder, the spectrometer, SIPAT, Matlab and Simca-Q, is developed. Moreover, an automatic *Design of Experiments* (DoE) to obtain a design space for several formulations manifests the advantages of SIPAT. Besides the determination of a design space, trajectory and process optimisation can be carried out.

## Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

| Place | Date | Signature |
| --- | --- | --- |

## Acknowledgement

When I started working on my thesis at the RCPE, I was not familiar with the area of pharmaceutical engineering. I was lucky to have great collegues and a brilliant supervisor who introduced me to the field of manufacturing drugs as well as the concept of PAT.

I want to thank Dr. D. Koller for his advice, supervision, crucial contribution and for many useful comments to this thesis. Furthermore, I would like to express my gratitude to Dipl.-Ing. P. Wahl for our dynamic exchange of ideas and for teaching me how to develop a chemometric model, providing me with the essential data. Special thanks also go to my collegues from the PAT group, Dipl.-Ing. M. Besenhard, Dr. Nicolas Heigl, Dipl.-Ing. R. Hohl and Dipl.-Ing. O. Scheibelhofer. I appreciated their help and guidance, and the most important one, the wonderful friendship that has developed from this project.

Special thanks also goes to Dipl.-Ing. B. Kavsek from Siemens supporting and guiding me thoughout the entire project.

Furthermore, I sincerly thank my supervisor Professor E. Brenner from the Institute for Technical Informatics for his guidance and advice.

I also want to thank the entire extruder team under guidance of Dr. Gerold Koscher who always offered me a helpful hand and supported me when working at the extruder.

I also would like to express my gratitude to Dipl.-Ing. A. Eitzlmayr and Dipl.-Ing. D. Treffer from the Institute for Process and Particle Engineering for supporting and providing me with some figures for my thesis.

Finally, I would like to express my thanks to BA M. Grösslich for many useful comments and proofreading. I am especially grateful to my parents and Margot, who supported me throughout my studies. Without them all this would not have been possible.

Graz, 12 December 2011 Daniel Markl

# Contents

# List of Figures

# List of Abbreviations

AIC ..... Akaike's Information Criterion
API ..... Active Pharmaceutical Ingredient
ARMAX ..... Autoregressive Moving Average with External input
CC-Link IE ..... Control and Communication Link Industrial Ethernet
cGMP ..... Current Good Manufacturing Practice
COM ..... Component Object Model
CQA ..... Critical to Quality Attributes
CSMA/CD ..... Carrier Sense Multiple Access with Collission Detection
DCOM ..... Distributed Component Object Model
DoE ..... Design of Experiments
DS ..... Design Space
FDA ..... US Food and Drug Agency
GUI ..... Graphical User Interface
HME ..... Hot Melt Extrusion
HMI ..... Human Machine Interface
HPLC ..... High-performance liquid chromatography
IP ..... Internet Protocol
ISO ..... International Organization for Standardization
LAN ..... Local Area Network
MAC ..... Media Access Control
MIMO ..... Multi-Input Multi-Output
MLR ..... Multiple Linear Regression
MPI ..... Multi Point Interface
NIR ..... Near-Infrared
OAE ..... Overall Assett Effectiveness
OLE ..... Object Linking and Embedding
OPC ..... OLE for process control
OPC DA ..... OPC Data Access
OSI ..... Open System Interconnection model
PAT ..... Process Analytical Technology
PCA ..... Principal Component Analysis
PID ..... Proportional-Integral-Derivative

PLC ..... Programmable Logic Controller
PLS-R ..... Partial Least Square Regression
PRBS ..... Pseudo Random Binary Sequence
Profibus ..... Process Field Bus
QbD ..... Quality by Design
RCPE ..... Research Center Pharmaceutical Engineering
SCADA ..... Supervisory Control and Data Acquisition
SISO ..... Single-Input Single-Output
SME ..... Specific Mechanic Energy
SNR ..... Signal-to-Noise Ratio
SNV ..... Standard Normal Variate
SQL ..... Structured Query Language
SWAXS ..... Small and Wide Angle X-Ray Scattering
TCP ..... Transmission Control Protocol
TPP ..... Target Product Profile
TPT ..... Throughput Time
UNF ..... Unified Fine Thread
UV-Vis ..... Ultraviolet-Visible spectroscopy
VAF ..... Variance-Accounted-For
WinCC ..... Windows Control Center

# 1 Introduction

## 1.1 Process Analytical Technology

> *"The pharmaceutical industry has a little secret: Even as it invents futuristic new drugs, its manufacturing techniques lag far behind those of potato-chip and laundry-soap makers."* [4]

A *Wall Street Journal* article described the manufacturing techniques of potato chips as more advanced than those of pharmaceuticals. Using new technology, silicon chip makers have increased their manufacturing efficiency by more than 10,000% in the past few decades, whereas the pharmaceutical industry has been using the same basic manufacturing processes. The technological progress has been made in the equipment but not in the process itself.

The US *Food and Drug Agency* (FDA) published *Process Analytical Technology* (PAT) guidelines for pharmaceutical processes to catch up with other industries concerning process development [5].

### 1.1.1 Background

In August 2002, the FDA announced a 2-year initiative, titled *"Pharmaceutical cGMPs for the 21st Century: A Risk-Based Approach"*. The aim of the FDA is to enhance the regulation of pharmaceutical manufacturing and product quality. In September 2004 the PAT guidance, *"PAT - A Framework for Innovative Pharmaceutical Development, Manufacturing, and Quality Assurance"*, was released [6].

FDA proposed PAT to anticipate technical and regulatory issues. The goal of PAT is to understand a manufacturing process as well as process automation and control. Furthermore, it includes designing and analysing of pharmaceutical manufacturing processes, chemical engineering and knowledge as well as risk management. Significant regulatory barriers have suppressed pharmaceutical manufacturers from adopting state-of-the art manufacturing practices. The FDA wants to stimulate the pharmaceutical industry to focus on the science of manufacturing [6, 7].

Process understanding contains the identification and explanation of all critical sources of variability and accurate and reliable predictions of product quality attributes [6]. Moreover, it includes a mechanistic understanding of formulations and process factors.

### 1.1.2 PAT Measurement Principles

Full quality control of pharmaceutical materials and manufacturing processes using PAT tools guarantees a final *target product profile* (TPP). Continuous real-time process verification can be accomplished by monitoring univariate (temperature, pressure, etc.) and multivariate (spectrum, image, etc.) process and performance attributes. Univariate process parameter and off-line material product characterisation are used in traditional approaches. PAT implies in-line measurements of product quality attributes and additionally, process regulation to minimise product loss. There are different methods to obtain accurate measurements of product quality attributes. Direct measurements imply a method to directly get the information about the parameter of interest. Contrary, an indirect method needs combinations of different indirect measurements to extract the necessary information. The following listing classifies PAT methods:

- Direct Measurements:
  - *Near-Infrared* (NIR) Spectroscopy
  - Chemical Imaging
  - Raman Spectroscopy
  - Chemical Mapping
  - UV-Vis Spectroscopy
  - Mass Spectroscopy
  - Gas Chromatography
  - *Small and Wide Angle X-Ray Scattering* (SWAXS)
  - Terahertz Spectroscopy
  - Optical Coherence Tomography
  - *High-performance liquid chromatography* (HPLC)
  - etc.
- Indirect measurements:
  - Generic sensors
    - Physical Attributes (size, shape, temperature, pressure, etc.)
    - Chemical Attributes (pH value, etc.)

PAT is applied to the *Hot Melt Extrusion* (HME) process and uses one direct method (NIR spectroscopy) and some generic sensors, e.g. temperatur, pressure, torque, to analyse the process. Critical process parameters affecting the product quality attributes are obtained by periodical measurements to design, analyse and control the HME process [6].

## 1.2 Simatic SIPAT as a PAT Software Solution

The growing industry activity concerning PAT encouraged Siemens to develop a PAT software solution, namely SIPAT (Siemens AG, Brussels, Belgium). Characterised by its inherent modularity, SIPAT allows the user to monitor and identify *Critical to Quality Attributes* (CQA), to control and finally to optimise the manufacturing process. The data is collected in real-time and SIPAT aligns the measured data from different collectors. One collector is chosen by the user during the method configuration procedure as the reference collector. Thus, SIPAT carries out an aggregation function on the data of all the other collectors to guarantee time alignment. Furthermore, SIPAT brings in-line, at-line, on-line and off-line data in relation and provides real-time predictions on product and process quality.

On the one hand, SIPAT offers common communication interfaces, e.g. OPC technology, to use sensors from various different manufacturers. On the other hand, it allows the developer to easily import PAT tools, e.g. Spectrometer, by providing customised communication interfaces.

Summarised, SIPAT allows the user:

- to enhance process understanding.
- to improve continuous processes.
- to develop processes based on *Quality by Design* (QbD) principles.

The tools to support PAT principles are:

- Process Analysers,
- Process Control tools,
- Data analysis and mining tools (Multivariate Data Analysis, etc.) ,
- Data collection, storage and retrieval tools,
- Reporting tools,
- Continuous improvement and knowledge management tools [7, 8].

All the PAT tools mentioned are linked to SIPAT. Therefore, the integration of PAT in an existing manufacturing process is carried out by SIPAT.

## 1.3 HME as a Pharmaceutical Manufacturing Process

Solid dosage forms (tablets and capsules) are by far the most popular dosage forms in use today, because of their enhanced stability and easy use. However, these dosage forms also induce bioavailability, stability and manufacturing challenges [9]. To enhance and modernise formulation processes and increase the efficiency of manufacturing operations whilst improving therapeutic efficacy, the FDA guidance facilitates the introduction of manufacturing processes to the pharmaceutical industry [10]. Particularly in the area of solid dispersion the HME

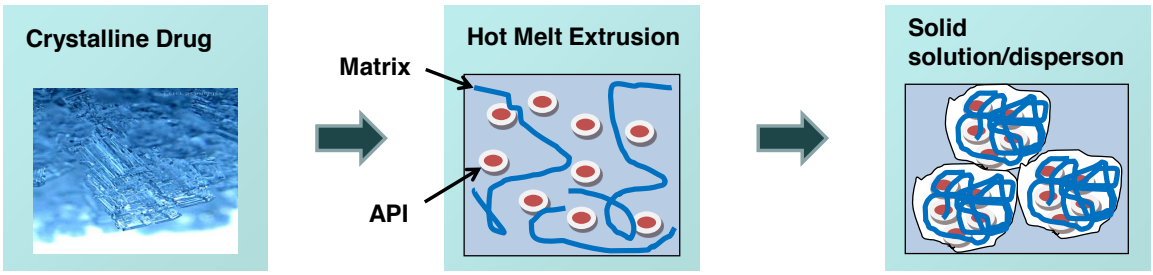



Figure 1.1: HME as a pharmaceutical drug processing unit.

technology has captured the interest of the pharmaceutical industry. Compared to traditional pharmaceutical production processes, the HME process has a lot of benefits.

In the ideal case the HME as a pharmaceutical drug processing unit molecularly dissolves *active pharmaceutical ingredients* (API) in the polymer, as illustrated in Figure 1.1. The matrix acts as a thermal binder, drug stabiliser, drug solubiliser and/or drug release controlling excipient with no compressibility requirements. Therefore, the selection of an appropriate carrier compound mainly depends on drug-polymer miscibility, polymer stability and function of the final dosage form [10]. The result is an enhanced rate of dissolution and solubility of drug molecules compared with the crystalline form. Extrudate solid solutions offer greater thermodynamic stability than those prepared by alternative processes such as spray drying, solvent evaporation and other hot melt methods [9]. Additionally, solid dispersions of drugs increase the active agent bioavailability and the duration of the drug action in the body and reduce side effects.

The HME process allows effective integration in a continuous manufacturing environment (Figure 1.2). The extruder is fed with a (preprocessed) matrix and an API. Directly after exiting the die, the hot, still molten strand is cut with a rapidly rotating cutting knife into pellets of uniform size, shape and density. In this process configuration it is advantageous that there is no need for a subsequent spheronisation step and it does not require a further melting of the product. As a consequence, it lowers equipment cost and reduces energy demand. The further downstream processes might be a tablet press followed by a coating process to produce tablets or an encapsulation process to obtain capsules [11].

The transition from batch to continuous processing is the intention of the pharmaceutical research community. The major advantages of continuous manufacturing are i.a. the reduction of process volume and thereby the integration of quality to the process. Adjustments and controlling of continuous processes are more effective in real-time than for batch processes. Depending on a single large-scale batch process causes high risk to success in delivering a high product quality. In contrast, continuous processing spreads out risk over time. For the preparation of drug delivery systems, the HME process has been gaining more and more attention in the pharmaceutical industry. The HME process is a continuous manufacturing process and combines multiple batch unit operations in one single process [6, 12].

The manufacture of drugs is still changing from the traditional batch processing via lean manufacturing to continuous processing. *Quality by Design* (QbD) approaches improved the

understanding of process steps and their impacts compared to traditional manufacturing treating disconnected process steps. The throughput rates are still increasing while the throughput times are decreasing, as illustrated in Figure 1.3a and 1.3b.

The way of continuous processing offers several advantages:

- Reduced throughput time.
- High overall asset effectiveness.
- Reduction of the systems' foot print.

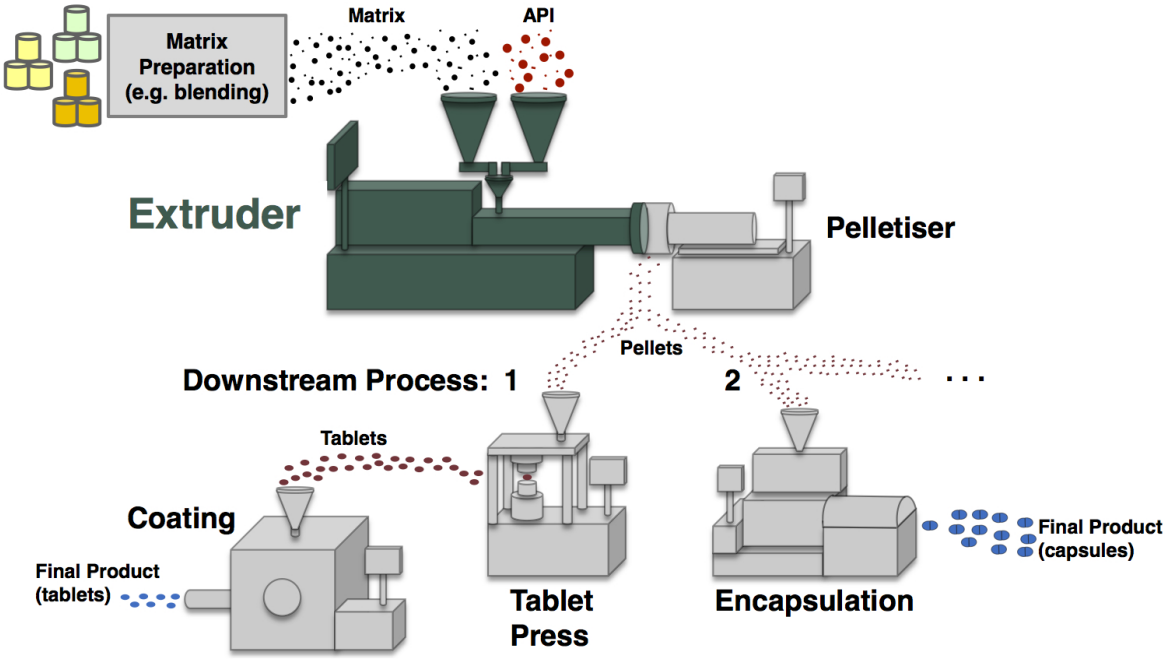
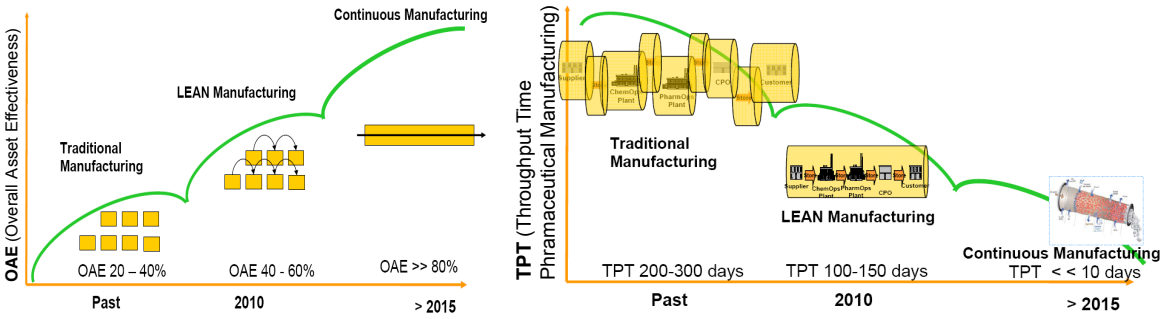



Figure 1.2: The integration of the HME process to the plant.




(a) The throughput rates increase.

(b) The throughput times decrease.

Figure 1.3: Evolution of the type of manufacture over time [1].

- Reduction of capital and operational costs.
- Reduction of raw material and intermediate inventories.
- Integration of compliance/quality within the process.
- Reduced time to market.

Contrary, there are some challenges/disadvantages of continuous manufacturing:

- Can not be applied on every process.
- Dedicated equipment and facilities required.
- Process technology under-developed.
- Advanced and robust PAT and control approaches required.
- Many sensors do not exist (e.g., powder flow rate, impurities, etc.).
- Single point of failure can bring down the whole plant.
- Exceptional events management required.
- Regulatory framework is under-developed.

To sum up, the benefits of continuous manufacturing speak for themselves and continuous processing will determine the development of the manufacture of drugs of the future. Apart from the advantages of continuous manufacturing, the hot melt extrusion as a pharmaceutical manufacturing process has several additional advantages:

- Mixing, melting and extrusion in one step.
- Increased bioavailability due to formation of solid dispersions, which is relevant for novel drugs with poor solubility.
- Processing in absence of solvents and water.
- Economical process with reduced production time and fewer processing steps.
- Short residence times and therefore, suited for temperature-sensitive drugs.
- High drug loadings enables the production of small pills.
- Extruded material can be further processed into a variation of dosage forms, including capsules, tablets, and transmucosal systems.

Yet on the other hand there are some disadvantages and challenges of the HME process:

- Lack of detailed design/simulation tools.
- Configuration of extruder (e.g. screw, barrel, die geometry).
- Prediction of performance still rudimentary.

Hot melt extrusion technology represents an efficient pathway for the manufacture of drug delivery systems. Different process steps like mixing, melting, homogenising, and shaping can be carried out by the HME process and offers many opportunities for automation of the manufacturing plant to limit material loss, to increase the throughput, to decrease energy input, and to yield a product with high quality. Significant impact on the degradation of drugs and excipients has the design of the screw assemblies and extruder dies, which are still areas of development [2].

## 1.4 Scope of this Work

This chapter introduced hot melt extrusion as a pharmaceutical manufacturing process, which is discussed in more detail in chapter 2 including an overview of the individual processing tasks and general functionalities. Additionally, process parameters and characteristic attributes of the extrusion process are also explained in this chapter.

The PAT approach based on a SIPAT installation (version 3.1.1) is applied on the HME process at *Research Center Pharmaceutical Engineering* (RCPE) GmbH, Graz, Austria. Chapter 3 describes the network architecture of the current SIPAT configuration including the HME process and the NIR spectrometer. SIPAT alignes the spectrum of the extrudate (at the die) and scalar measurements (screw speed, barrel temperatures, material pressure, etc.) and provides a common database. The user or external software can access all the measured data from the SIPAT database in real-time or off-line, as illustrated in Figure 1.4. Chapter 3.4 and 3.5 present the communication between the different components including the used interfaces, the functionalities of the system and its subparts and the data flow of measurements and configuration data.

In the first stage SIPAT is used for monitoring of the output parameters of the HME process. Furthermore, establishing a direct connection from SIPAT to the extruder in order to set the input parameters by the SIPAT user or predefined sequences opens up new opportunities to exhaust the capability of SIPAT. Figure 1.5 illustrates the basic concept and also shows the

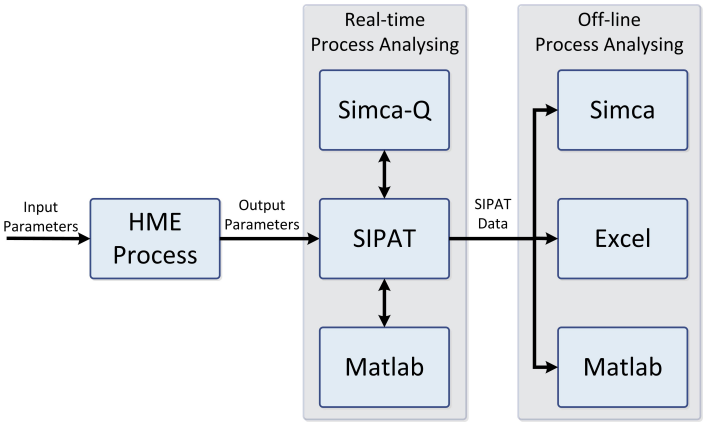



Figure 1.4: Real-time and off-line process analysing are used in this work.

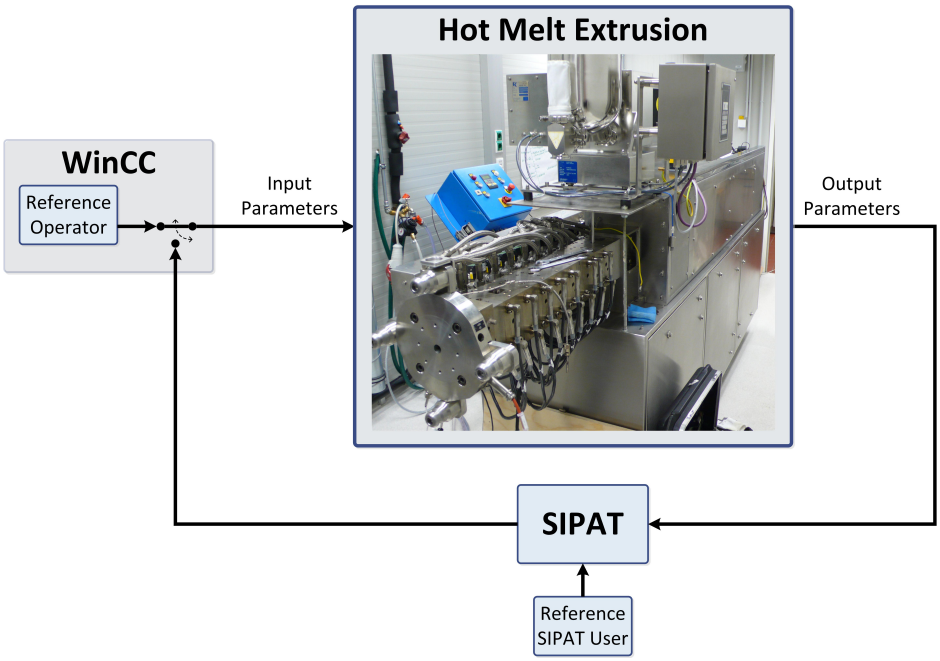



Figure 1.5: The HME process with its input and output parameters, which can be manipulated and monitored by SIPAT.

usual way of manipulating input parameters by the operator via WinCC. In chapter 4 a use case is described demonstrating the integration and interaction of the HME process (generic sensors, actuators), the spectrometer, SIPAT, Umetrics Simca, and Mathworks Matlab. This use case shows the strength of SIPAT by performing an automatic *Design of Experiments* (DoE).

Finally, a black-box model of the extruder presented in chapter 5 is developed and used for simulations and real-time predictions. The application of real-time predictions of the API concentration combines real-time and off-line analysing of the HME process. The last part of this work includes a conclusion and an outlook.

# 2 Hot Melt Extrusion Process

## 2.1 Introduction to the Hot Melt Extrusion Process

The first section introduces various different types of extruders and points out the advantages of the extruder at the RCPE. Section 2.2 gives an overview of the individual processing tasks and general functionalities. Additionally, process parameters and characteristic attributes of the extrusion process are explained in this section.

### 2.1.1 Overview and Classification of Extruder

Industrial applications of extruder date back to the 1930's. This is why the HME process is a well elaborated manufacturing process technology in the plastic- as well as the food-industry [2]. The HME process applications now range from reactor processing over chocolate kneading to the pharmaceutical sector. As for pharmaceuticals, various dosage forms can be manufactured, ranging from pellets, over granules to tablets, and transdermal drug delivery systems [11]. The extrusion process converts a raw material into a product of uniform shape and density by forcing it through die under controlled conditions [2].

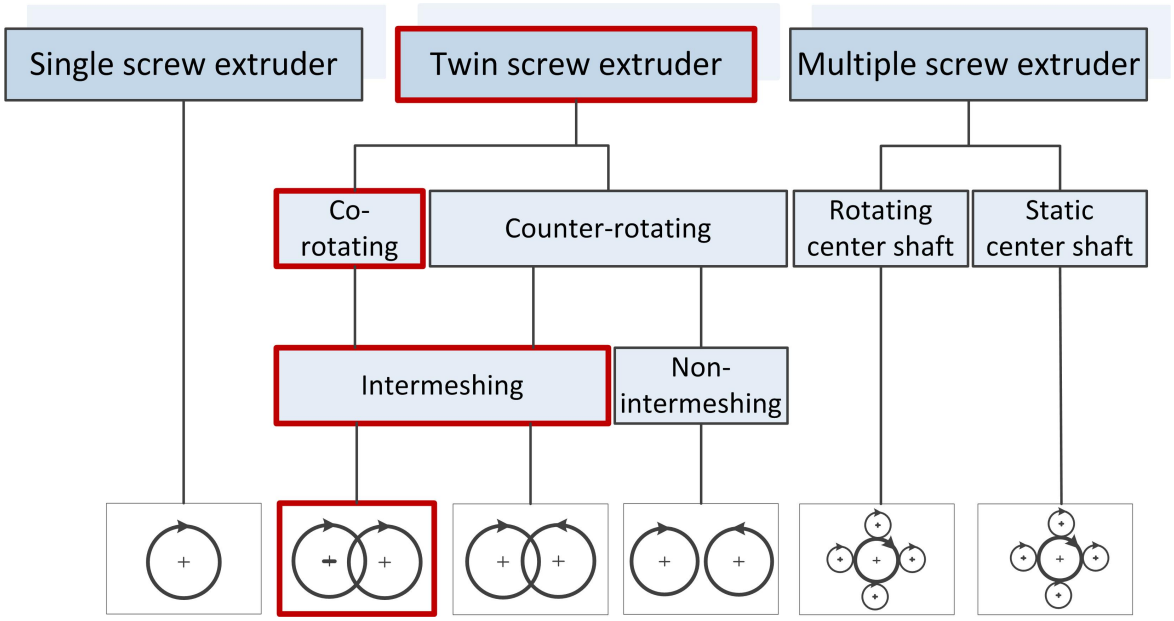



Figure 2.1: Classification of extruder.

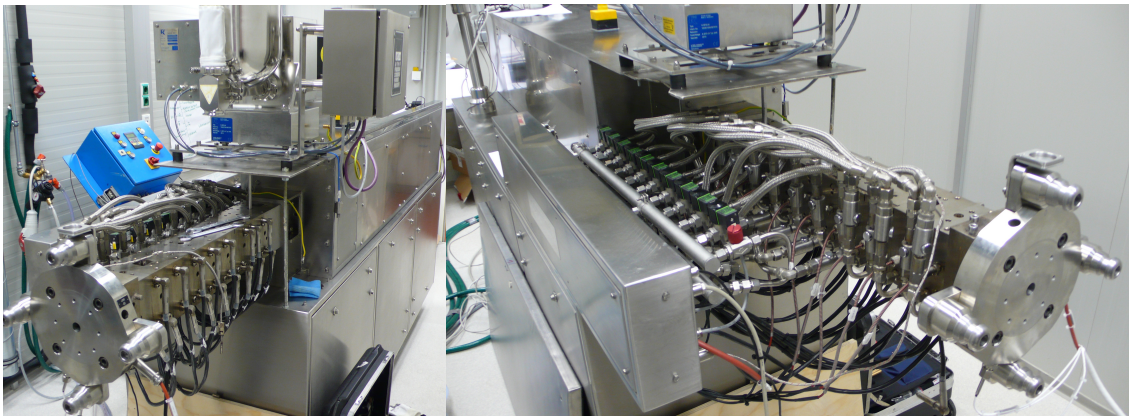
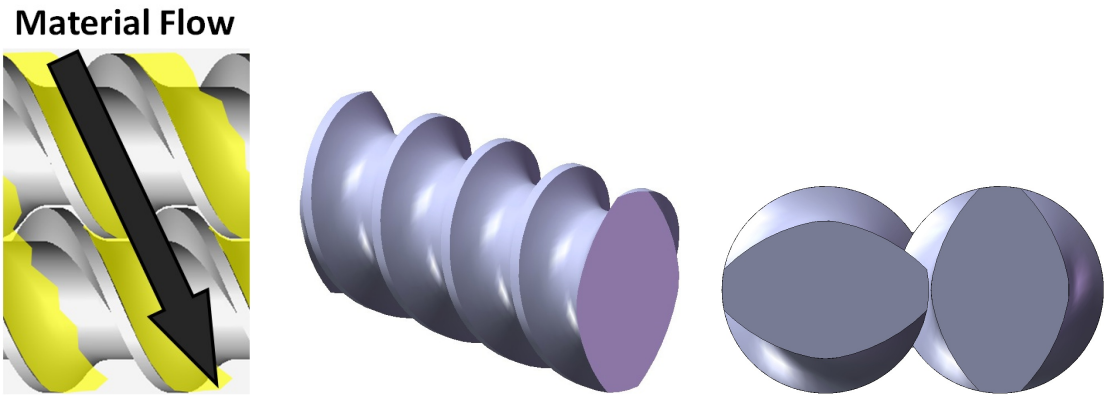


Figure 2.2: Hot Melt Extrusion process.

Extruder can be classified according to their number of shafts, as illustrated in Figure 2.1. Single screw, in other words single shaft extruders, are applied in plastic processing mainly for melting and pressure build-up, allowing extrusion through a die or injection into a mould. The disadvantage of limited mixing ability of single screw extruder delimitates the utilisation of this type of machine as a manufacturing process of pharmaceutical products. The twin screw extruder has two parallel shafts rotating together in the same direction (co-rotating) or in the opposite direction (counter-rotating). Counter-rotating screws can either rotate towards the center, or rotate away from the center (seen from the above). Consequently, the material is squeezed through the gap between the two screws as they come together. Co-rotating twin screw extruders can be operated at higher screw speeds than extruder with counter-rotating screws, due of the pressure that develops from the outward pushing effect as the screws come together in rotation.

Twin screw extruder can be separated into intermeshing and non-intermeshing machines. Advantage of intermeshing extruder are that the flights mesh tightly except for the necessary clearance, and therefore, the machine is designated as kinematically "self-cleaning" [3, 13]. Multiple screw extruder are only used for special purposes and are not applied in the pharmaceutical industry.

An intermeshing co-rotating twin screw extruder (ZSK 18, Coperion GmbH, Stuttgart, Germany) with 18mm screw diameter is used at *Research Center Pharmaceutical Engineering* (RCPE) for the research of manufacturing of pharmaceuticals, as illustrated in Figures 2.2. Furthermore, a twin-screw gravimetric feeder (KT20, K-Tron, Pitman, USA) is used to establish a certain throughput. For split feeding experiments, a second feeder of the same type is applied. Therefore, in the rest of the work some explanations refer only to these specific types.




(a) Axially opened flights result in the denoted material flow.

(b) Double flighted profile of the screws.

(c) Closely intermeshing screw profiles.

Figure 2.3: Illustration of some properties of the screws of the HME process.

### 2.1.2 Introduction to the Co-Rotating Twin Screw Extruder

The main components of an extruder are a feeding hopper, a barrel, screws, a die, a screw driving unit and a heating/cooling device. Furthermore, the process can be divided into four sections: feeding of the extruder, conveying of mass, mixing of the material and entering the die. To fulfill the requirements of the TPP, the extrusion process requires process monitoring of all sections and a good understanding of process and material parameters, e.g. viscosity, variation of viscosity with shear rate and temperature, elasticity [2].

Co-rotating twin screws are based on a modular design and facilitate an easy adaptation of the process to handle a variety of processing requirements and product characteristics. Extruder with two closely intermeshing (Figure 2.3c) co-rotating screws have the following properties:

- The flights of the screws clean each other. This self wiping screw profile ensures nearly complete emptying of the equipment and minimises product loss at shutdown.
- It ensures a narrow and well-defined residence time distribution. The residence time in the twin screw extruder is about 2 minutes.
- The flights of the two screws are opened axially resulting in a material flow as depicted in Figure 2.3a.

Double flighted twin screw extruders (Figure 2.3b) are used for the extrusion of temperature and shear sensitive material by applying low shear forces. Furthermore, double flighted screws in comparison to single flighted have the following advantages:

- Double flighted screws enable a high conveyer capability.
- The extruder can be operated at a higher feed rate.
- The product is treated more carefully [14].

## 2.2 Process Technology

A twin screw extruder has various processing zones, which are arranged in series. Every processing step is linked to the next. Thus they can not be considered independently. The barrels and the screws are based on a modular design and the screw must be assembled according to the formulation. Figure 2.4 illustrates an example of a certain screw configuration considering only one feeder. Analysing the unit operations in the extruder to a satisfactory degree can only be achieved if sufficient information about the rheological and thermodynamic properties of the polymers, polymer compounds and API(s) is available. Obtaining this data is difficult and partially not feasible and therefore, it is necessary to refer to model investigations and simulations. Especially the screw configuration depends on these properties and is still a hot research topic.

The following section is a brief introduction to the geometry of the screw. Section 2.2.2 gives an introduction to each zone and task of the extruder in respect to Figure 2.4. HME process parameters and characteristic parameters are presented in section 2.2.3.

### 2.2.1 Geometry

The closely intermeshing co-rotating twin screw extruder has two adjacent screw elements with identical geometry which are symmetrical and rotate at the same speed. Figure 2.5 illustrates the geometry of a conveying screw element. All screw elements have the same screw diameter, whereas this screw diameter is smaller than the barrel diameter to establish a specified clearance between the external wall and the screw tip. Additionally, the condition of a clearance between the screws (cf. Figure 2.3c) has to be kept. Screw elements can vary in pitch, channel depth or root diameter and helix angle. The screw elements of the RCPE

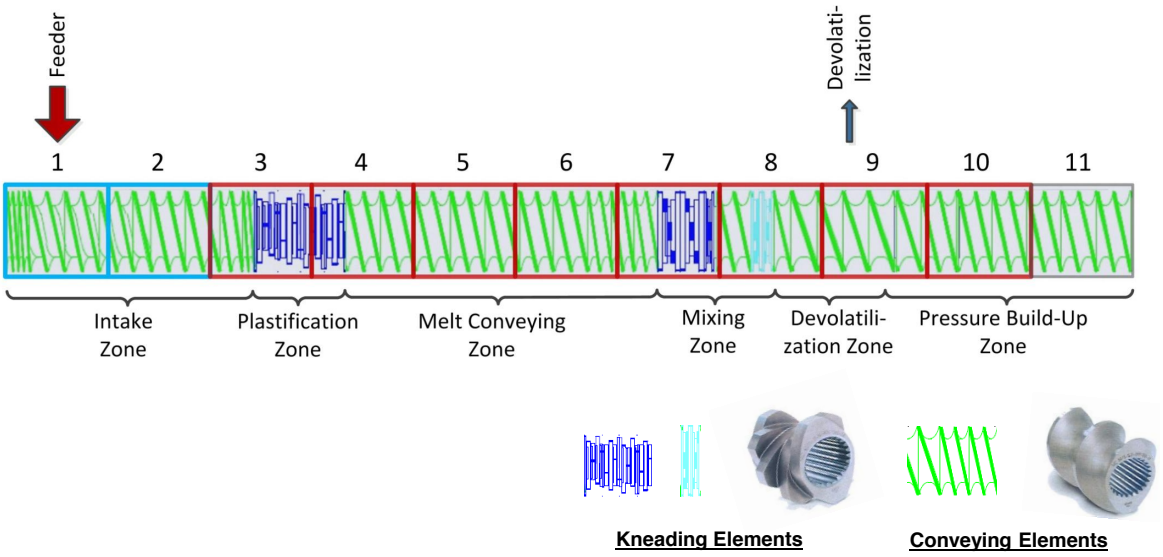



Figure 2.4: An example of a specific screw configuration.

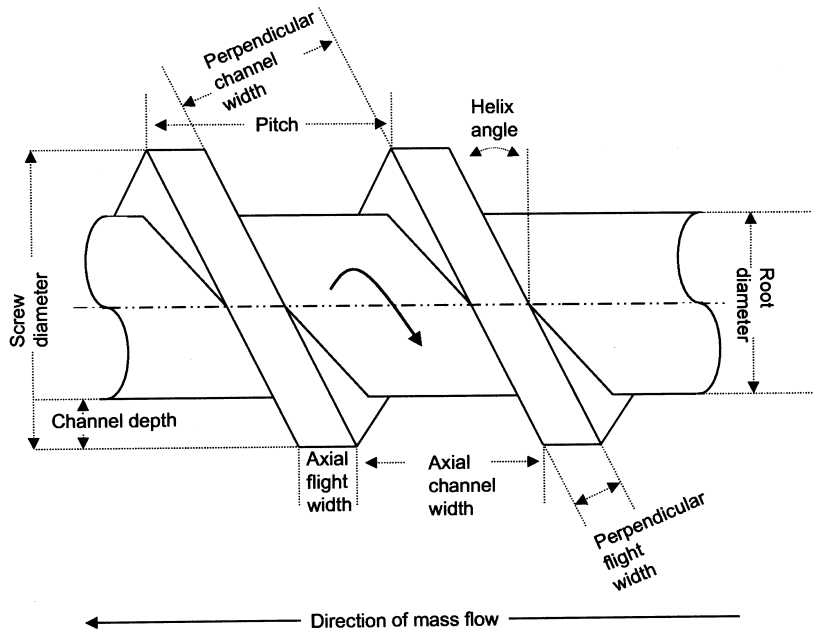
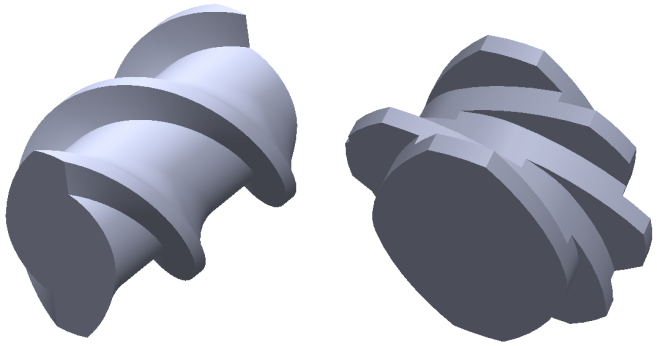



Figure 2.5: Illustration of the screw geometry of a conveying element [2].



(a) Conveying element. (b) Kneading element.

Figure 2.6: Illustration of different screw elements.

extruder only differ in pitch and helix angle. This geometry facilitates the feed material to fall easily into the screw for conveying along the barrel.

Depending on the section of the extruder, a certain screw element has to be chosen. Figure 2.6a presents a conveying element while Figure 2.6b illustrates a kneading block. The explanations of the compounding processes state correlations between the geometry parameters of conveying elements and kneading elements and process characteristics [3, 13].

### 2.2.2 General Overview of the Compounding Processes

This section gives an overview and introduction in the various tasks of an extruder. Figure 2.4 illustrates the locations and the processing sequence of the tasks. The following descriptions of the individual processing zones is based on this example, but it can be adapted to other screw configurations.

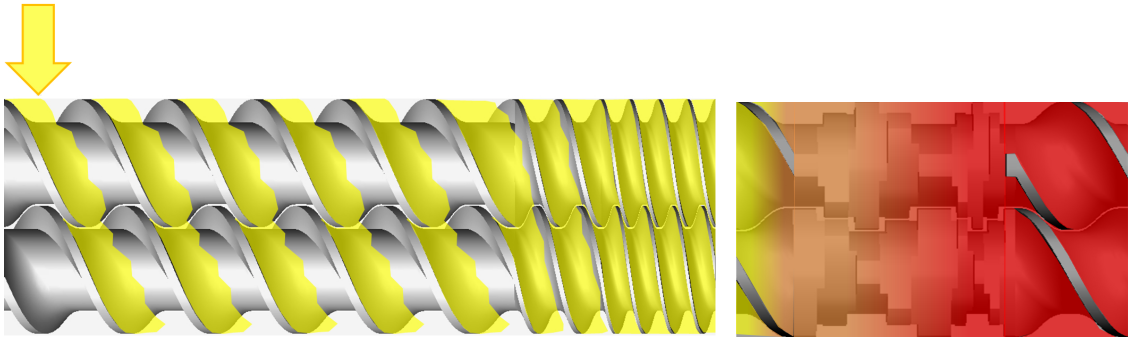
#### Intake Zone

The key tasks of the intake zone are conveying and compressing solids. The free screw volume, the pitch, the screw speed, and the bulk density of the material limits the feed capacity. Decreasing the thread pitch but maintaining a constant channel depth or decreasing channel depth while maintaining a constant thread pitch results in increased pressure as the material moves along the barrel [2]. As illustrated in Figure 2.7a, the pitch is reduced towards to the plastification zone in order to compress the matrix and to achieve a higher degree of filling at the beginning of the plastification zone. This causes high local pressure at the end of the intake zone, which depends on the polymer and may lead to a greater or lesser degree of wear [3]. Consequently, the melting characteristics will be improved.

The intake barrels (barrel sections 1 and 2) are cooled in order to prevent low melting-point components from adhering to the barrel wall since this could cause a blockage of the material directly below the feed hopper.

#### Plastification Zone

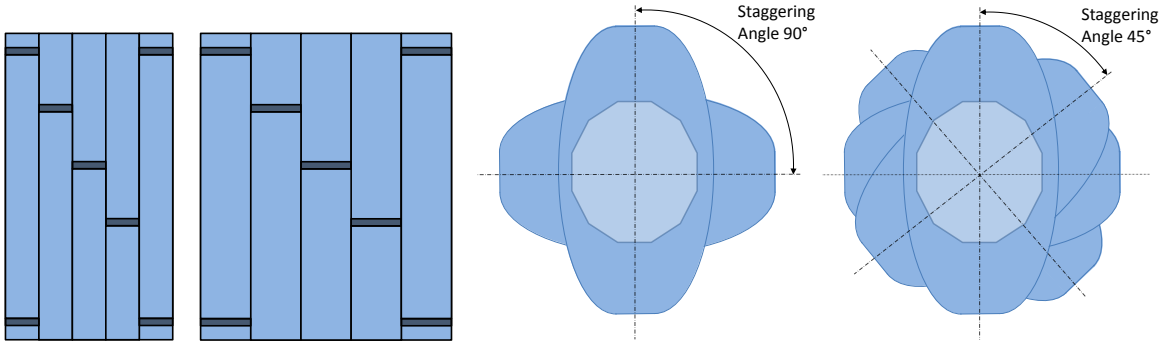
The melting process of the material is carried out in the plastification zone. Moreover, in this zone pre-dispersion of the filler is conducted. The ratio of melt viscosities $r = \mu_{dispers}/\mu_{contin}$ of the blend partners has a major influence on the melting process. $\mu_{dispers}$ and $\mu_{contin}$ are



(a) Reduction of the pitch to compress the matrix.

(b) Melting process in the plastification zone.

Figure 2.7: Characteristics of the intake and plastification zone are illustrated.




(a) Two kneading elements with different disc widths.

(b) Staggering angle of 45° and 90°.

Figure 2.8: Parameters of kneading elements are illustrated.

the viscosities of the blend partner present in the dispersed and continuous phase, respectively. The smaller the ratio $r$, the longer the melting process takes, but the sooner it begins [3].

The screw shafts transmit most of the energy required to melt the matrix, whereas the energy input from the heating of the barrels represent only a small fraction. The heat flow through the barrel is important to create an adhesion of the polymer on the wall. Thus, generation of a shear gradient can only be achieved if the temperature of the barrel wall is higher than the softening point of the polymer [3]. In other words, the barrel temperatures 3 and 4 have to be chosen carefully to guarantee the melting film on the wall. Furthermore, due to the transition from cooling to heating at the barrels 2 and 3, the energy required to heat the barrel walls has its maximum at barrel 3.

It is beneficial to completely fill the plastification zone to accelerate the melting process. Therefore, the screw designer should establish a pressure build-up at the end of the intake zone (reduction of the pitch) and back-pressure at the end of the plastification zone (e.g. re-conveying element).

Figure 2.7b shows the melting process in the plastification zone, where the color denotes the temperature of the material and, at the same time, it illustrates the phase transition from solid to liquid (yellow - solid and red - liquid). Solid particles might be still present after the plastification zone, but it may not be mandatory to achieve 100% melting at the end of this zone. These solid particles can be melted in subsequent zones, e.g. in the mixing zone [3].

Kneading discs are used to achieve the aim of the plastification zone. There are different parameters (Figure 2.8a and 2.8b) to choose by the screw designer which influence the behaviour and characteristic parameters of the process. Among other things, the staggering angle and the kneading disc width can be chosen:

A larger staggering angle

- reduces the downstream conveying action of the melt and thus decreases the residence time, and

- improves the mixing performance because more leakage streams are generated [3].

A narrower kneading disc

- reduces shear in the gap,
- increases downstream conveying action (decrease residence time) and
- mixes more efficiently due to the generation of more leakage streams [3].

### Melt Conveying Zone

Apparently, this zone conveys the melt from one processing zone to the next (e.g. from the plastification zone to the mixing zone). Usually the conveying zone is partially filled, except for the part needed for back-pressure or pressure build-up for pressure consuming elements. The melt conveying zone should dissipate as little energy as possible. Due to leakage flows over the screw tip and the intermeshing zone, it causes some slight back-mixing of the material [2, 3].

If a second feeder is applied on the process (split feeding), e.g. using the first one to feed the matrix and the second one for the API, then the melt conveying zone might be used additionally as an intake zone. Since there can raise a lot of problems attaching a second feeder (e.g. blockage of the material directly below the second feed hopper), the pitch, the feed rates, the screw speed, and the barrel temperature characteristic have to be chosen carefully to guarantee full functionality of the process. The use case discussed in chapter 5 requires split feeding in order to monitor process dynamics in respect to the API concentration. Due to enhanced process stability, both feeders were attached at the intake zone.

### Mixing Zone

The aim of the mixing process is to accomplish homogenous distribution of solids or liquids within the melt with the shortest mixing length and minimum energy input. The mixing zone contains distributive and dispersive mixing. The distribution process depends entirely on the intensity and direction of the ratio of drag and pressure flows and not on the viscosity of the melt. The dispersive mixing depends also on viscosity and is therefore affected by the shear stress in the screw channel.

The screw designer uses the same kind of screw elements, kneading discs, as in the plastification zone. The maximum shear stress and thus, the most effective dispersion, is generated by wide kneading discs, whereas this leads to high energy input, increases the melt temperature and reduces the viscosity of the melt. Reduction of the viscosity of the melt results in a reduction of shear stress. Therefore, dispersive mixing is an optimisation problem [2, 3].

#### Devolatilisation Zone

The key tasks of the devolatilisation zone is to remove water, residual monomers and solvents. It is important to guarantee completely filled zones on either side of the devolatilisation zone due to the axially open screw channels property of the co-rotating twin screw extruder [3].

#### Pressure Build-Up Zone

The pressure build-up zone is located in the extruder in the discharge zone. The pressure required to force the material through the die has to be generated by the pressure build-up zone. The aim of the design of the pressure build-up zone is to optimise the pressure generation in respect to consume as little energy as possible [3].

### 2.2.3 HME Process Parameters

In order to guarantee a final TPP, monitoring of dedicated process parameters has to be carried out. The HME process, the final product and the raw material imply a number of various parameters. Identification of critical process parameters is difficult and presupposes an excellent understanding of the process. This section presents a number of setting and process parameters, whereby only measurable parameters can be considered for additional process analysis or control. Thus, the definition of input and output parameters is also given in this section.

Figure 2.9 illustrates the extruder emphasising the input (red) and output (green) parameters. Furthermore, one might retrieve the location of the sensors from this figure, because it is significant for process analysing.

#### Setting and Input Parameters

A given formulation denotes the concentrations of the raw materials in the final product. In respect to the formulation the screw configuration, barrel temperatures and die geometry have to be designed and adjusted. According to an appropriate screw and die geometry a particular throughput and screw speed can be chosen. The throughput of a compounding process is limited by the available volume, torque, and the maximum tolerable product temperature. Advantageous compared to a single screw extruder, the throughput can be chosen independently of the screw speed [2, 3].

The setting parameters may be summarised as follows:

- Formulation (Concentrations)
- Extruder configuration (screw, die geometry, etc.)
- Barrel temperature profile
- Throughput

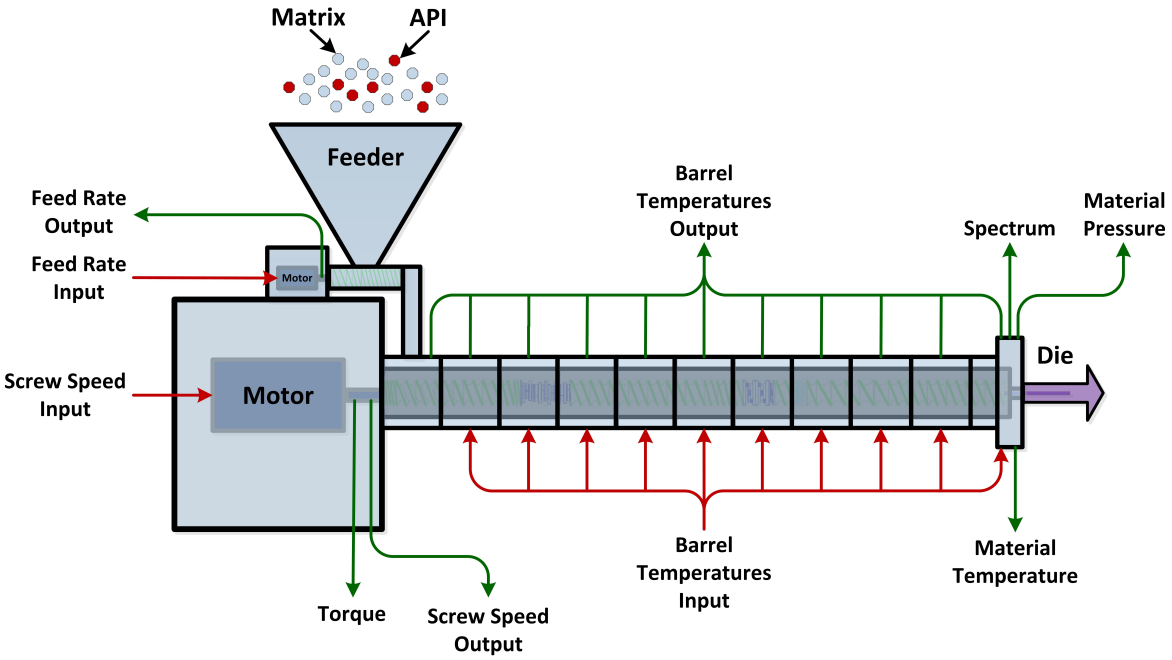



Figure 2.9: Schematic illustration of the extrusion process including the input and output parameters.

- Screw speed

During runtime, or in other words, after the configuration of the screw and the die geometry, the screw speed, feed rate and the barrel temperatures of the barrel sections 2 to 10 can be manipulated. Adjusting these input parameters the process characteristics can be influenced and the final product quality can be improved.

The following input parameters are specified for the operator:

- Screw speed $r_{ss}$
- Feed rate $r_f$
- Barrel temperatures $r_{t_i}$ with $i = 2, 3, ..., 10$

### Process and Output Parameters

Process parameters contain process characteristic information and enable a description of the process. Some important process parameters are:

- Intake characteristic
- Torque
- Pressure profile

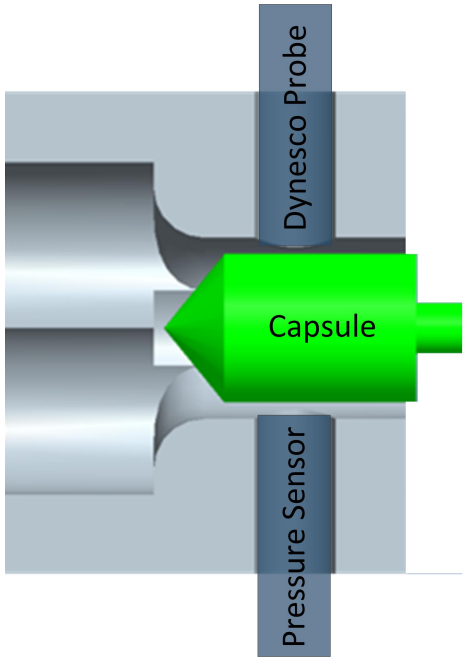
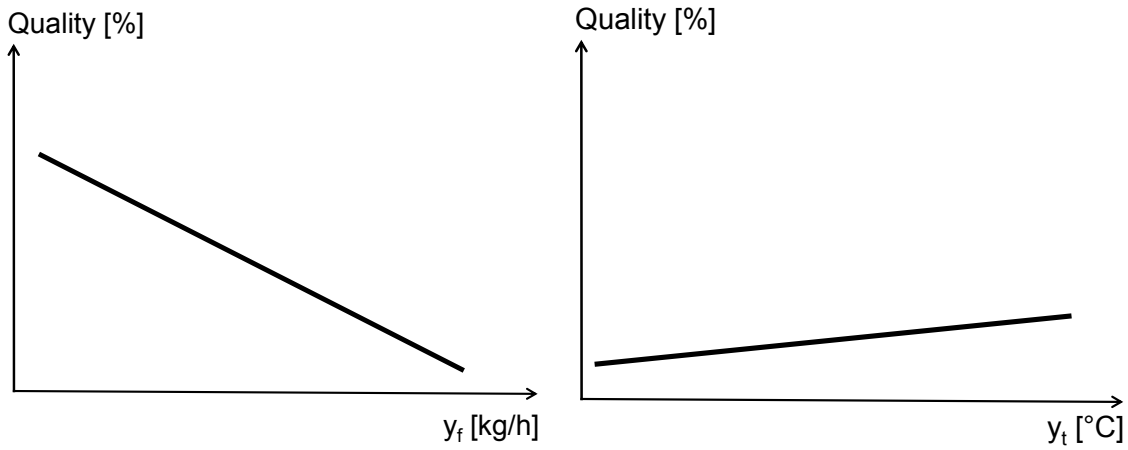



Figure 2.10: Mounted capsule to guarantee reliable measurements.

- Temperature profile
- Residence time distribution
- Dispersion quality
- etc.

Unfortunately, only a few process parameters can be measured directly. Thus, process characteristics have to be extracted from the measurable attributes defined as output parameters:

1. Spectrum $\mathbf{y_s}$. The NIR spectrometer is used to develop information about the quality of the product. The data type of a spectra is multi-value [8].
2. Sensors for measuring scalars:
   - Temperature of the barrel $y_{t_i}$ with $i = 1, 2, ..., 10$. Each barrel has attached one temperature sensor.
   - Feed rate $y_f$. Three sensors are used to measure the metering stream at the feeder.
   - Material temperature $y_{tm}$. There are three sensors which measure the temperature of the material in degree celsius.
   - Material pressure $y_{pm}$. The pressure of the material is measured in bar.
   - Screw speed $y_{ss}$ in revolutions per minute and
   - Torque $y_{md}$. The rpm and the torque of the main drive is measured [15].




(a) Quality in relation to the throughput. (b) Quality in relation to the barrel temperatures.

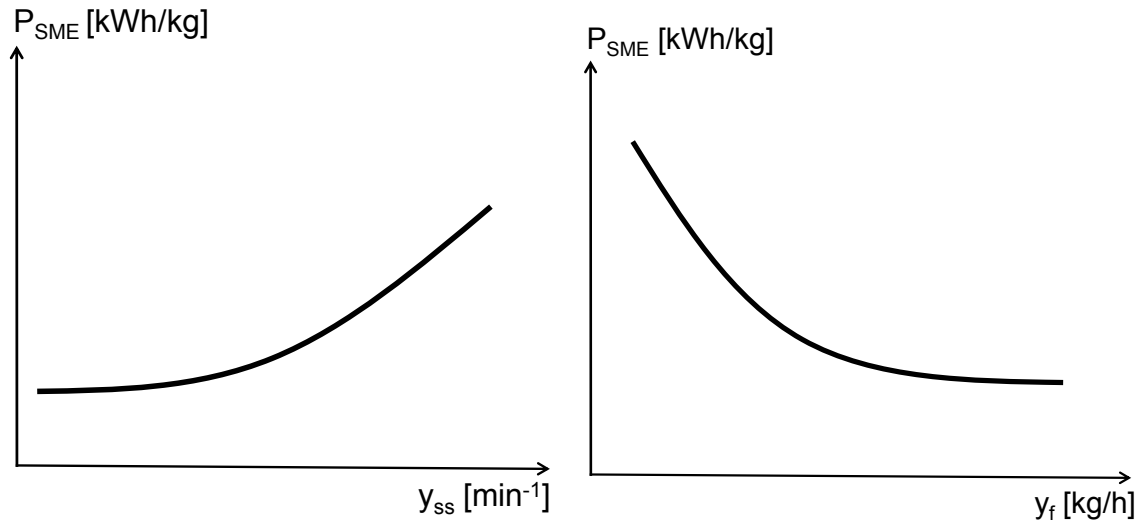
Figure 2.11: Influence of the screw speed and the throughput on the quality [3].

**NIR Spectrometer:** In the field of pharmaceutical engineering the NIR spectroscopy is one of the most popular instruments for raw material testing, product quality control and process monitoring [16]. In contrast to other analytical techniques, NIRS has its major advantages in:

- No or limited sample preparation needed.
- Prediction of chemical and physical sample parameters from one single spectrum after calibration.
- Separating of the sensor head and the spectrometer by use of fiber optic probes.
- Supporting a high measurement rate [6].

The used SentroPAT FO (Sentronic GmbH, Dresden, Germany) with a fibre-optic Dynisco NIR probe is a diode array based near-infrared spectrometer. The Dynisco probe is a special probe for 1/2" UNF thread as used for extruders. It is a fast and nondestructive technique and covers the wavelength range of 1100nm to 2200nm with a resolution of 2nm. NIR spectra are collected in transflexion mode. 120 spectra are averaged and an integration time of 0.014s per spectrum is used. The Dynisco probe should be attached on the process at an appropriate location. In order to guarantee reliable measurements a capsule as illustrated in Figure 2.10 is mounted. This construction induces an annular gap which avoids the extrudate forming a channel and thereby the observed spectrum can be used to predict the current chemical composition of the extrudate.

The properties of NIR spectra, e.g. low intensity, wide and overlapped bands, cause the usage of data filtering and multivariate data analysis techniques to develop analytical methods [6]. Relating two sets of data, the spectra and the reference values/concentrations, by regression results in a model. After a successful calibration of the model, concentrations of the API(s) in the extrudate can be predicted using new measured spectra. This procedure is discussed in




(a) Specific mechanical energy input in relation to the screw speed.

(b) Specific mechanical energy input in relation to the throughput.

Figure 2.12: Influence of the screw speed and the throughput on the specific mechanical energy input [3].

more detail in the use case to model the HME process in chapter 5. Based on these predictions, correlations between the quality and output parameters can be investigated. From a general point of view, Figures 2.11a and 2.11b state the relationship between the quality and the throughput and barrel temperatures. Figure 2.11b shows that the product quality is rather independent of the barrel temperatures, whereas Figure 2.11a represents the influence of the throughput on the quality. In order to develop a controller, these investigations, among other things, have to be elaborated.

### Characteristic Process Parameters

A characteristic process parameter offers significant information about the process behaviour and thereby monitoring and investigation of these parameters is mandatory to improve process understanding. The *specific mechanical energy* (SME) input and the residence time characteristics are identified as such characteristic parameters in several references [3, 17, 18].

**Specific Mechanical Energy Input:** The specific mechanical energy input is one of the most important characteristic values. Regardless of the size of the extruder it is characteristic to each product. The specific mechanical energy input ($P_{SME}$) via the screw shafts states the measured screw speed $y_{ss}$, torque $y_{md}$ and the throughput/feed rate $y_f$ in relationship, as denoted in Equation 2.1. Due to the limited heat transfer through the barrel walls, most of the energy input in the process is reflected by the $P_{SME}$.

$$P_{SME} = \frac{2 \cdot \pi \cdot y_{ss} \cdot y_{md}}{y_f} \tag{2.1}$$

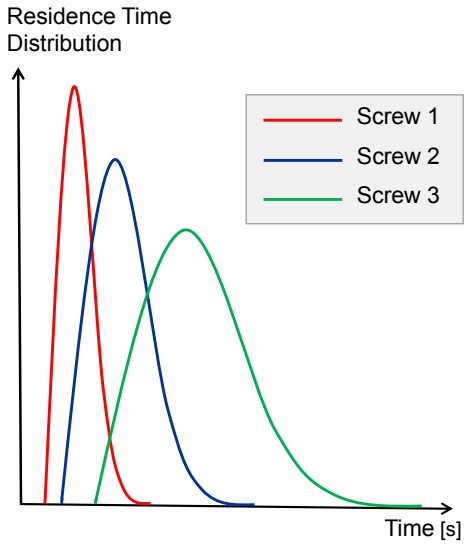



Figure 2.13: Influence of screw configurations on the residence time distribution.

Figures 2.12a and 2.12b illustrate the relationship between the specific mechanical energy input and the throughput and screw speed. Apparently, $P_{SME}$ increases as screw speed increases with constant torque. Furthermore, the throughput decreases as the specific mechanical energy input increases [3, 17].

**Residence Time Characteristics:** Especially for heat-sensitive products, the residence time characteristics in an extruder has a great influence on the quality of the product. As it is a continuous process, the residence time cannot be specified precisely and therefore, the residence time distribution characterices a process [3].

In partially-filled screw sections the average residence time only depends on the pitch and the screw speed, whereas in fully-filled sections only the throughput influences the average residence time. Figure 2.13 compared the residence time distribution for different screw configurations. A simple conveying screw (Figure 2.13 - screw 1) has a narrow residence time distribution, while a screw with a small section of kneading elements (Figure 2.13 - screw 2) and a long section with kneading elements (Figure 2.13 - screw 3) results in a wider residence time distribution. The screw configuration as depicted in Figure 2.4 might have a residence time distribution as illustrated in Figure 2.13 for screw 3.

Apart from the residence time depending on the screw configuration, process parameters equally influence this characteristic parameter. Thus, throughput and screw speed influence the residence time as well.

# 3 Network Architecture

## 3.1 Overview

The network architecture is the logical and structured layout of a network. It illustrates infrastructure as well as connectivity between components. Figure 3.1 illustrates the network architecture according to the SIPAT installation at the RCPE. It is a bus network in which a set of network components are connected via a shared communication line. The communication between the SIPAT components is based on *Ethernet* technology. One of the network interfaces of the controller of the HME process is named *Profinet*, which is based

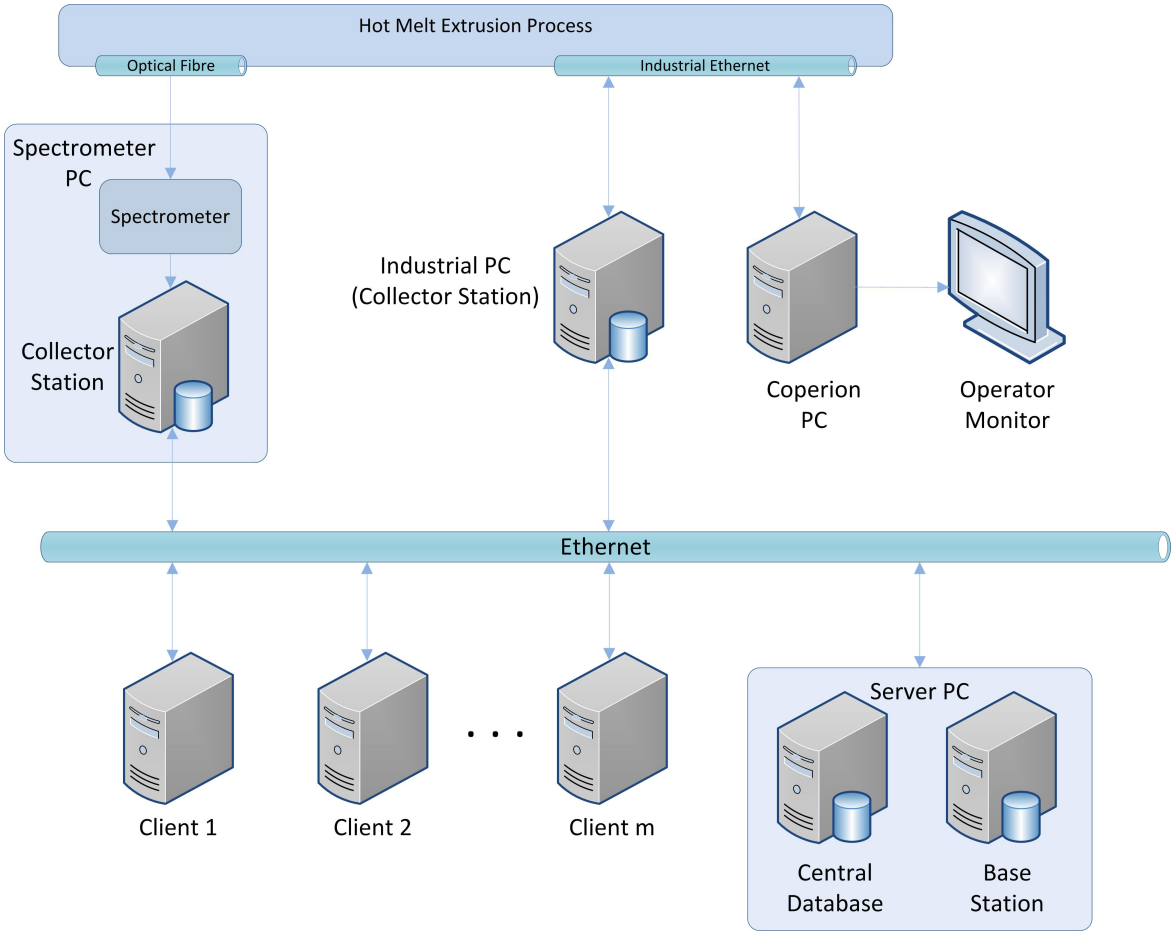



Figure 3.1: Network architecture of the Hot Melt Extrusion process with Simatic SIPAT.

on *Industrial Ethernet* [19]. The other interface to the extrusion process is an *Optical Fibre*. This is only introduced for convenience to have a defined interface which can be used in further explanations. A bus network and digital/analog signals are used for data transmission inside the HME process [15]. All the interfaces and communication principles mentioned are described in section 3.4.

The main components are

- a Spectrometer PC as a collector station,
- an Industrial PC as a collector station,
- the Coperion PC with the operator monitor,
- a Hot Melt Extrusion process,
- a Server PC including the central database and the base station and
- the clients.

Detailed explanations of these components are given in the sections 3.2 to 3.3.3.

## 3.2 Schematic Circuit Diagram of the Hot Melt Extrusion Process

The HME contains a few different electromechanic components, as illustrated in Figure 3.2. The brain of the process is the Simatic S7-300, a *Programmable Logic Controller* (PLC). A PLC is used for automation in industries and machines and can be programmed easily for certain tasks.

The PLC from Siemens has a Profinet, a *Profibus* (Process Field Bus) and a *Multi Point Interface* (MPI). The Profinet interface is used to connect the PLC to the Industrial Ethernet to enable the communication to the Industrial PC and to the Coperion Service Box. The Profibus connects the Weight Feeder, the Frequency Converter Main Drive and the PLC. Moreover the Main Drive, Processing Section and Die, Vacuum Unit, Heating/Cooling Unit, and additionally, the Weight Feeder and the Frequency Converter Main Drive are communicating with the PLC via digital and/or analog signals [15].

The NIR probe is integrated in the HME process and directly connected to the the Spectrometer PC via an optical fibre.

## 3.3 Principles of SIPAT Components

Basically, various different SIPAT components establish the functionalities of SIPAT. Apart from the SIPAT client, the major components are the base station, collector stations and the central database. This section introduces these three components in respect to the SIPAT installation at the RCPE.

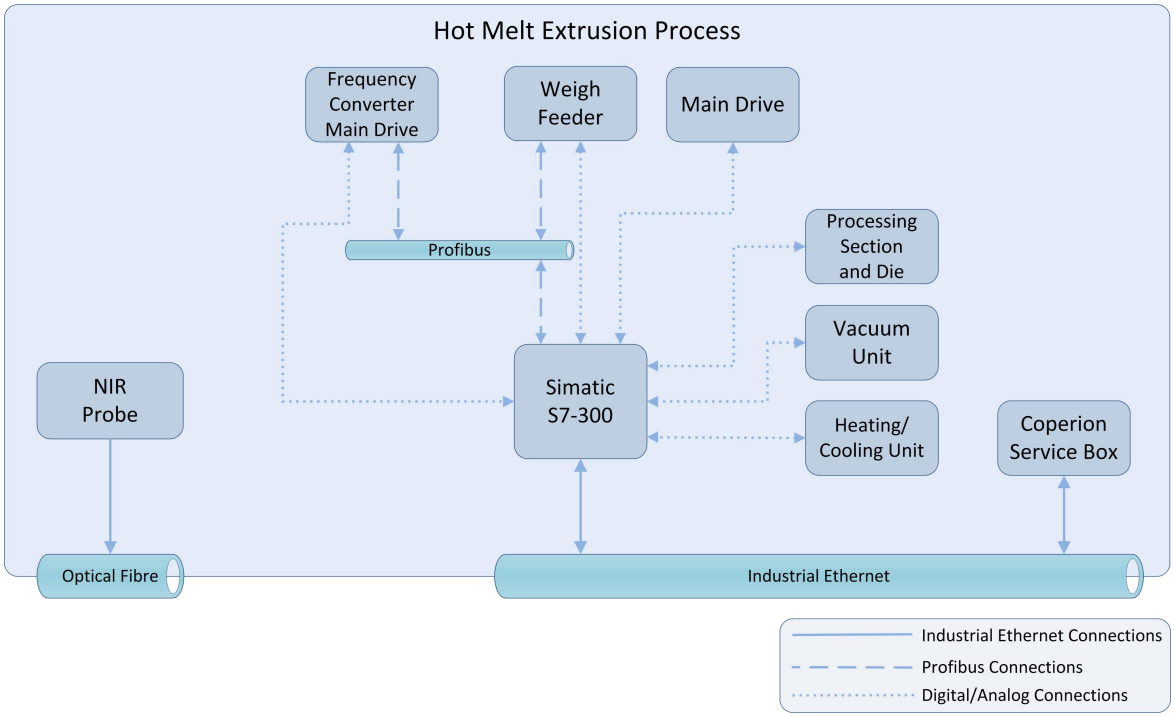



Figure 3.2: Schematic circuit diagram of the Hot Melt Extrusion process.

### 3.3.1 SIPAT Base Station

A base station focuses on unit operations, e.g. blending, tabletting, coating. It uses the Ethernet interface to communicate with other stations, e.g. collector stations. Preconfigured methods can only be executed on a base station and are assigned to a default base station. The archiving of data and calculations based on the input are done by methods. Collectors provide input data for methods, and user functions can write outputs to associated parameters which have to be defined as OPC tags with writing permission [8].

**SIPAT Services:** Windows services are processes that carry out a particular function and do not require a user intervention. The services can be started at boot time or manually and may run in the background as long as Windows is running. Some SIPAT services depend on other SIPAT services. Therefore, the starting sequence of services must be kept in the right order. However, SIPAT Station, SIPAT Watchdog and SIPAT Database Logger Service have to run on every station to enable basic SIPAT functionalities.

The following SIPAT services run on the base station at RCPE and are illustrated in Figure 3.3:

- SIPAT Watchdog: A watchdog timer is used to monitor certain services. Due to some faulty condition the watchdog timer can trigger an action to move the system into a safety state. Consequently the watchdog service verifies the “health” of the running SIPAT services.

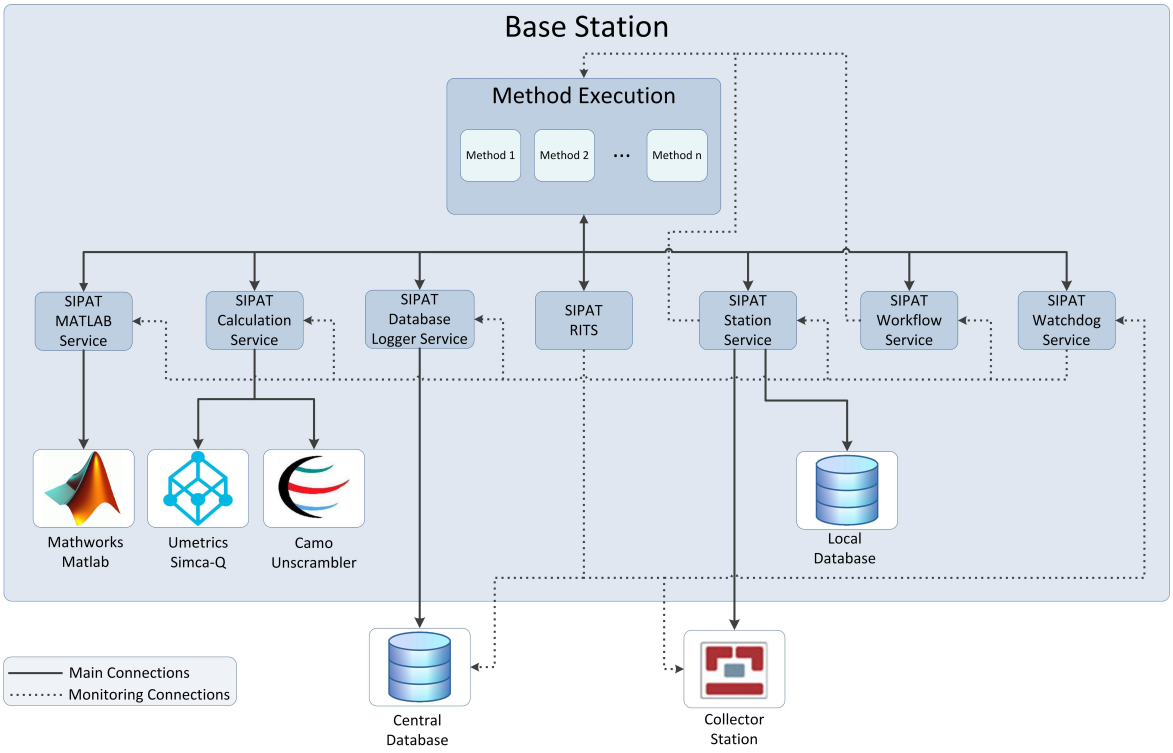



Figure 3.3: SIPAT services running on the base station. Main connections are denoted as the actual transfer of data contrary to the monitoring connections

- SIPAT Workflow: The workflow service is responsible for the execution of a collector intervention or a method workflow. Thus, the service has to be installed on the collector station for an implemented collector intervention and on the base station if a method uses the workflow functionality.
- SIPAT Station: This service runs methods, collectors and writes the data to the local database. Furthermore, it reads values from an OPC server and distributes the data to methods. This station has to be defined as a base station.
- SIPAT Database Logger: This service stores runtime data to the central database.
- SIPAT Calculations: The Calculation Service is needed to use the calculation engines:
  - Umetrics Simca-Q 12.0.1 and
  - Camo The Unscrambler 9.8.
- SIPAT Matlab: The Matlab Service is responsible for Mathworks Matlab (R2008b) model calculations.
- SIPAT RITS: Basically, the SIPAT Runtime Information Service (RITS) is responsible for getting the data from the central database [8, 20].

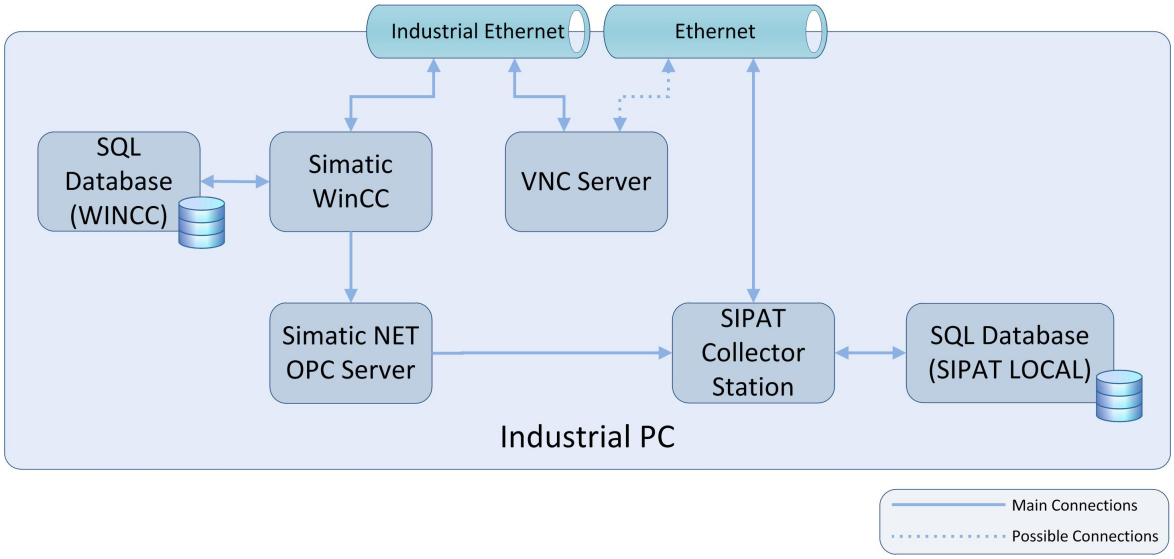



Figure 3.4: Internal structure of the Industrial PC.

### 3.3.2 SIPAT Collector Stations

Collector Stations are responsible for collecting data from sensors, archiving the latest data in a local database and providing data for methods running on a base station [8]. The HME system consists of two collector stations. One collector station corresponds to the spectrometer and the other one uses OPC technology to distribute data from different sensors from various manufacturers to client PCs.

#### Spectrometer PC as Collector Station

The spectrometer is connected to the collector station and the Spectrometer PC represents the combination of these two components into one single PC. Different spectrometers can be integrated into SIPAT, as long as appropriate drivers exist. More details about the communicatin between the spectrometer and SIPAT is given in section 3.4.5.

#### Industrial PC as Collector Station

The industrial PC contains the following components, as illustrated in Figure 3.4: (1) Simatic WinCC, (2) Simatic NET OPC Server, (3) SIPAT Collector Station, (4) VNC Server and (7) two different SQL databases. The communication of automation components which includes amongst others, the PLC, PC and HMI, are summarised from Siemens by the concept of Simatic NET. Simatic NET connects Siemens automation systems and systems from other manufacturers. Hence the communication inside the Industrial PC and to the PLC is based on Simatic NET.

The SIPAT Collector Station block determines the running SIPAT services on this PC. Simatic WinCC and the Simatic NET OPC Server are described in the next paragraph. Database blocks are used for the archiving of runtime data. The *Virtual Network Computing* (VNC) principle enables a platform independent remote control of other computers. After establishing a connection from a VNC viewer to a VNC server, the viewer can remotely control the computer which runs the VNC server. Thus, the industrial PC runs a VNC server and the Coperion PC uses the VNC connection to realise operator access to the industrial PC and display the WinCC Runtime engine on the operator screen.

**Simatic WinCC:** Simatic WinCC is a *Supervisory Control and Data Acquisition* (SCADA) system that contains i.a. a *Human-Machine Interface* (HMI). SCADA generally refers to industrial control systems which monitor and control manufacturing processes.

WinCC is written for the operating system Windows. Further significant characteristics of WinCC are to archive measuring data, to detect and handle certain events as well as to establish connections to external systems. WinCC runtime displays measurements and process status messages on the operator screen. Hence the operator can monitor all the process parameters in real-time [21]. The operator screen is shown in Figure 3.5.

WinCC archives runtime data in its own local database on a Microsoft SQL server. Historical data can be retrieved by sending a query to the database and can be displayed in WinCC Runtime.

**Simatic NET OPC Server:** *OLE for process control* (OPC) stands for *Object Linking and Embedding* and specifies a standard for the communication of real-time process data between the control devices and sensors from different manufacturers. The OPC server offers certain tags for the OPC client. The client may retrieve data by accessing the associated tag. The OPC server is specified by the *OPC Data Access* (OPC DA) specifications. It provides specifications for real-time communications between data acquisition devices and HMIs or databases. Therefore, OPC DA deals only with real-time data and not with historical data [22].

**SIPAT Services:** Collector stations require only a few services in order to guarantee SIPAT functionalities. The basic explanation of the following services was given in section 3.3.1:

- SIPAT Watchdog
- SIPAT Workflow
- SIPAT Database Logger
- SIPAT Station: This station has to be defined as a collector station.
- SIPAT OPC Writer: The OPC Writer Service is responsible for writing values on a specified OPC tag.

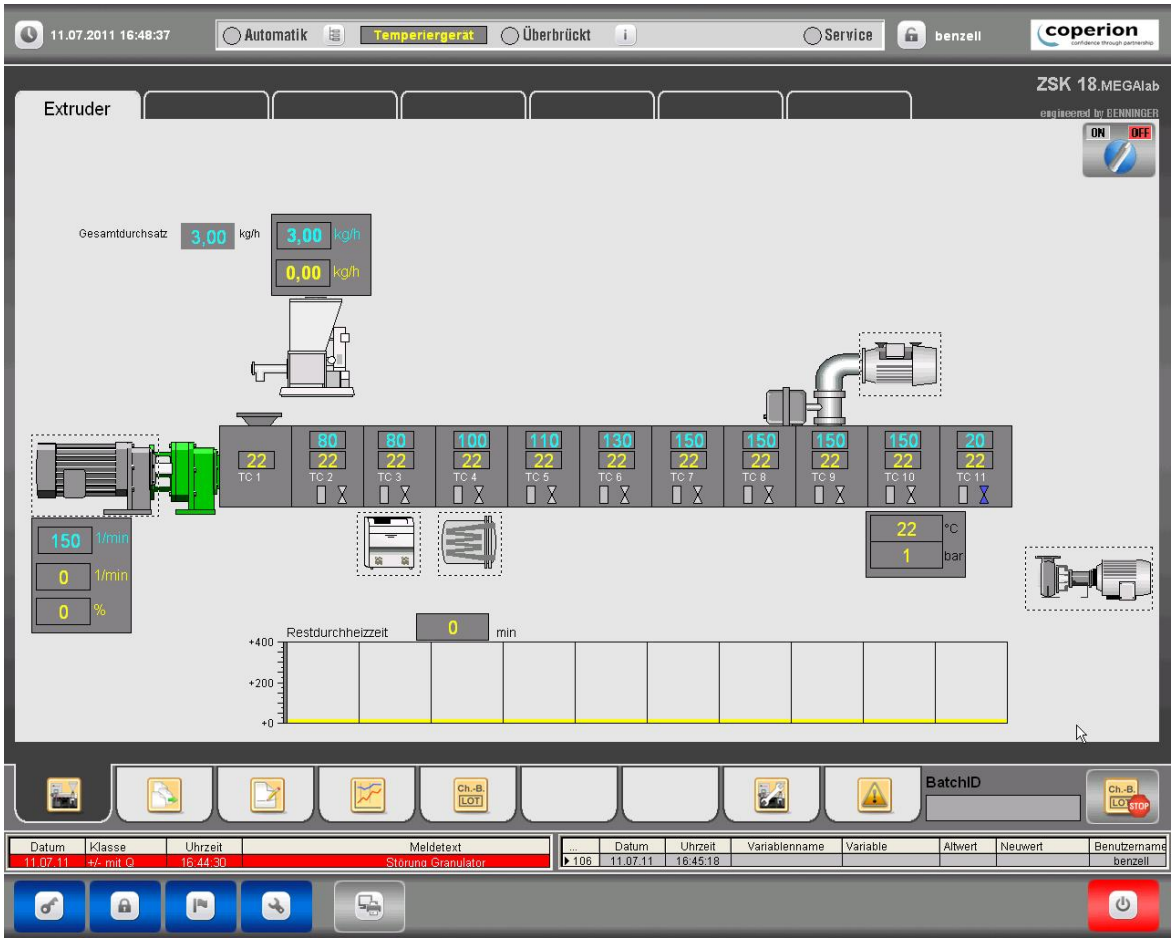



Figure 3.5: WinCC Runtime at the operator screen of the Hot Melt Extrusion process

### 3.3.3 SIPAT Central Database

The central database and the base station are executed on the Server PC. Microsoft SQL Server is used as a relational model database server. Due to the overhead for storing data from SIPAT to the database and the storing of spectra, the size limitation of the database has to be large enough to avoid data loss.

The central database is the brain of the SIPAT system. Apart from the measured and calculated data, most of the configurations of clients, collector stations, the base station, and users are stored in the database. The SIPAT system cannot be used without the central database.

## 3.4 Communication and Interfaces

The interfaces such as ethernet, industrial ethernet, profibus, and optical fibre are presented in this section. Furthermore, the communication between internal components of the extruder and the Industrial PC, and within the Industrial PC are introduced.

### 3.4.1 Interfaces

#### Ethernet

Ethernet is a *Local Area Network* (LAN) technology and is used to connect workstations. It can run over both optical fibre and twisted-pair cables and provides additional performance and network intelligence for the communication system. The communication between clients, collector stations, the base station, and the Microsoft SQL Server is based on Ethernet [21].

#### Industrial Ethernet

The industry organisations ported the traditional field bus architecture to Industrial Ethernet, as the Ethernet standard is widely used and offers many advantages. Ethernet was expanded to Industrial Ethernet from data communications between PCs to manufacturing control networks. In the layers up to and including the network layer, the same protocols are used as for Ethernet. From the transport layer to the application layer, the protocols for Industrial Ethernet are appropriate field bus specifications [22, 23].

**Profinet:** Profinet is an open industrial Ethernet standard and is based on the Ethernet TCP/IP standard [22]. It extends Profibus technology to establish fast reliable data communication based on this standard and to fulfill industrial requirements [19].

#### Profibus

Profibus is a field bus communication standard for automation systems and is universally applicable. It can be used for the communication between different devices without any particular adjustments. Profibus is a multi master system and therefore enables shared operations of multiple automation and visualisation systems with decentralised peripheral devices using the same bus [24].

#### Optical Fibre

The optical fibre transmits light between the NIR probe and the spectrometer and therefore functions as an optical waveguide. The main advantage of this set-up is the opportunity of directly measuring in the product stream.

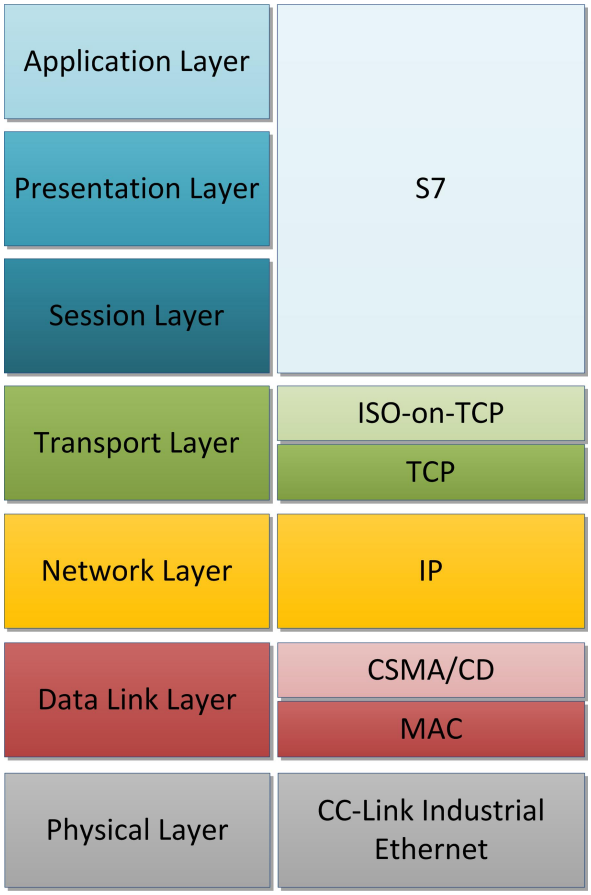



Figure 3.6: OSI Model of the WinCC - PLC Communication.

### 3.4.2 Communication from Simatic WinCC to the PLC

Profinet is used as the interface at the PLC to the Industrial Ethernet. The industrial PC, which executes the WinCC services and applications, has an Industrial Ethernet interface. The communication between WinCC and the PLC is therefore based on Ethernet technology. The *Open System Interconnection model* (OSI model) of the communication system is explained in the next section.

#### OSI Model of the Communication between WinCC and the PLC

The *Open System Interconnection model* (OSI model) subdivides a communication system into layers. It specifies protocols for every layer. There are seven layers where a lower layer provides services for a upper layer.

The layer architecture is illustrated in Figure 3.6 with the following explanations:

**Physical Layer:** This layer specifies the electrical and mechanical properties of the transmission medium. The *Control and Communication Link Industrial Ethernet* (CC-Link IE) enables the communication between devices from numerous manufacturers.

**Data Link Layer:** The data link layer implements the protocol to transfer data between adjacent network nodes. This layer uses the *Media Access Control* (MAC) data communication sublayer which provides channel access and addressing mechanism. The channel access mechanism is specified by the *Carrier Sense Multiple Access with Collision Detection* (CSMA/CD) protocol.

**Network Layer:** The *Internet Protocol* (IP) is responsible for the routing of packages.

**Transport Layer:** This layer provides an end-to-end error detection and correction. *Transmission Control Protocol* (TCP) is used as the transport protocol to guarantee reliable and ordered delivery of a stream of bytes from one network node to another one. ISO-on-TCP is used to guarantee a packet oriented data transfer from ISO and routing functionality from TCP/IP [25]. Thus, the communication is block oriented and not stream oriented like TCP.

**Session/Presentation/Application Layer:** These layers are combined and specified by the Siemens S7 protocol. It provides the applications with services to establish connections.

**S7 Protocol** S7 is a communication protocol developed by Siemens. It is based on Ethernet communication and is used to communicate with the controller and the sensors. Simatic S7 and C7 controllers support S7 communication applications [26].

The usage of the S7 protocol based on the hardware and software set up has the following advantages:

- Caused by the optimisation of the protocol for Simatic communications, the processor and bus utilisation for transferring large amount of data is less.
- Does not depend on the bus medium. As a consequence, the S7 protocol can be used for Industrial Ethernet, Profibus or MPI as high-level communication protocol.
- The S7 protocol is located at layer 7, the application layer; therefore, it is able to carry out an acknowledgment of data packages.

### 3.4.3 Communication from Simatic WinCC to Simatic NET OPC Server

WinCC gets the measuring data in real-time and provides this data to the OPC Server. Every measurement is linked to an OPC tag which can be accessed from an OPC client. The mapping from the data in WinCC to the OPC tag is unidirectional. Therefore, in order to write reference values on an appropriate input parameter, a certain tag associated to this input parameter has to be defined. The configured OPC tags to enable monitoring sensor data and manipulating input parameters are accessible by every OPC client and are listed in section 3.6.

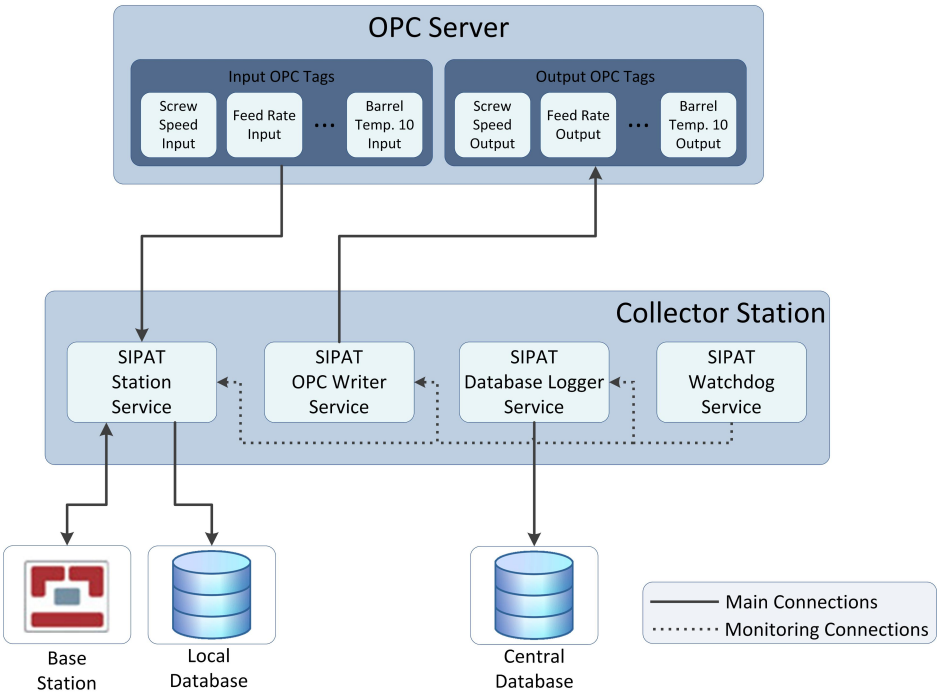



Figure 3.7: Communication between OPC Server and the services of the collector station.

### 3.4.4 Communication from Simatic NET OPC Server to SIPAT Central Database

Figure 3.1 and 3.2 illustrate that the Industrial PC with the OPC Server, the PLC, the Coperion Operator PC and the Coperion Service Box are connected to the same subnet. The Industrial PC is both connected to the subnet as well as to the RPCE network. As a consequence, the Industrial PC has access to the base station. This connection is a requirement for the SIPAT system, because the Industrial PC represents a collector station.

The SIPAT collector station (on the industrial PC) is basically an OPC client to be able to communicate with the Simatic NET OPC Server. The station service is responsible for reading from a defined OPC tag, whereas the OPC Writer service carries out the writing of values on a specified OPC tag, as illustrated in Figure 3.7. The OPC technology would allow to establish a connection from an OPC client running on another PC (e.g. base station) using DCOM. This procedure should be avoided, because DCOM may cause some troubles (e.g. firewall configurations, etc.). Since the collector station acting as a OPC client is located at the Industrial PC, the communication between SIPAT and the Simatic OPC Server is based on COM which excludes DCOM troubles.

The storing of data to the central database depends on the execution of a method on the base station. A method is used to gather data from defined collectors, to perform basic calculations or start advanced model calculations using external calculation engines. The industrial PC

collector station is imported in a method as a *Process Collector*. Consequently, the transfer of the data from the collector station to the central database is performed by SIPAT.

### 3.4.5 Communication from Spectrometer PC to SIPAT Central Database

SIPAT has its own collector interface for a number of different collectors. Hence, for the Sentronic spectrometer a driver for the integration in SIPAT already exists. This collector has to be imported in a SIPAT method as an *Instrument Collector*. One can either use the SIPAT collector interface or the spectrometer software (SentroSuite) in order to access the spectra from the spectrometer. The configuration of the spectrometer (e.g. definition of the integration time, the number of spectra which are averaged, etc.) can be defined by the software SentroSuite and is loaded from the SIPAT collector interface.

Another way to establish a communication between SIPAT and this spectrometer would be the use of OPC technology with the advantage of avoiding the requirement of an appropriate driver. Though the advantages of the collector interface are the easy import of the spectrometer in SIPAT as a collector and to carry out the calibration of the spectrometer out of SIPAT.

## 3.5 System Functionalities and Data Flow Diagrams

The major functionalities of the system are to measure process parameters, to forward data from the process to the central database, and to perform some actions in respect to the data. The path of a data package from the actual sensor reading to the central database is described by means of data flow diagrams. The described interfaces and communication protocols, as explained in section 3.4, are used for the transfer of data packages. The data flow independent of SIPAT components is treated in section 3.5.1 and the data flow with and in the SIPAT system is explained in section 3.5.2. Moreover, this section covers the data flow for the configuration of methods and the central database and the data flow during the execution state of a method.

### 3.5.1 Data Flow Diagram of External Components

Figure 3.8 shows the data flow from the sensors to the interface of the SIPAT components. In other words, this data flow diagram shows the part of the system which does not include a SIPAT component but prepares the data for further processing using SIPAT functionalities.

There are two different data flows from the HME to the base station:

1. The NIR spectrometer to get information about the quality of the product, as described in section 3.3.2. The data type of a spectra is multi-value [8].
2. The sensors for measuring scalars. The observable parameters are discussed in section 3.6.

Details to the communication between each blocks are given in section 3.4.

### 3.5.2 Data Flow Diagram of SIPAT Components

This section is divided into a configuration part and an execution part. This means, the data flow of the configuration is separated from the execution of a method. The main components are the two collector stations, the client and the server PC including the central database and the base station.

#### Method Configuration

In the HME system every client is assigned as PAT Administrator. Therefore, the configuration of a method is done by a client, whereas the configuration of the SQL database server has to be carried out on the Server PC without using SIPAT. Different collectors can be assigned to one method, but a method belongs to only one base station. Among others, basic and advanced calculations, model calculations using external software, runtime visualisations or workflows can be configured from a client. These method specifications are stored in the central database on the server PC. The same PC runs the SIPAT services of the base station. The configuration data flow is depicted in Figure 3.9.

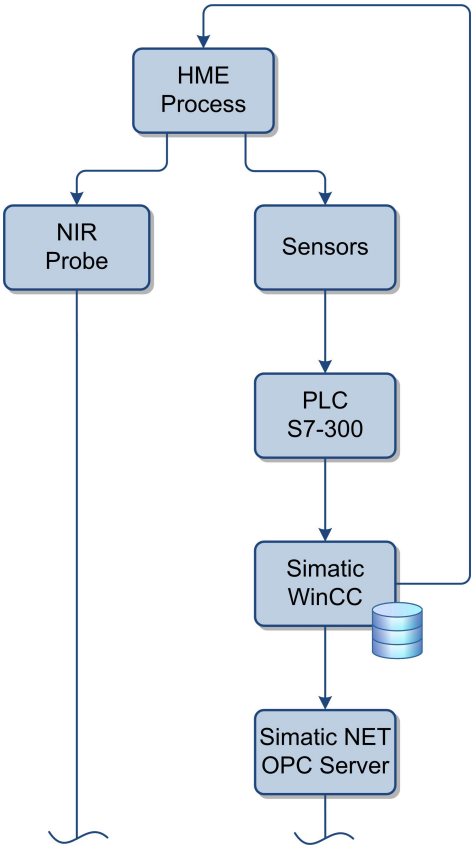



Figure 3.8: Data flow diagram from the sensors to the SIPAT interface.

After the execution of a method, selected data is stored in the central database and can be retrieved by sending a SQL query to the database. This data can be exported to a file, e.g. Excel-file, or visualised by SIPAT. To conclude, method specifications and data selected by a method are stored in the central database.

### Method Execution

Figure 3.10 displays the data flow from and to SIPAT components. All the measurements can be accessed by an OPC tag in SIPAT. Thus, a Process Collector, defined as an *OPC Async Collector*, has to be used in a method to access the measurements. The spectrometer has to be imported in SIPAT as an Instrument Collector [8].

After a successful method configuration, a client can create method instances on the base station. The base station loads the dedicated method specifications from the central database and waits for the client to press the start button before it executes the method. The method runs on the base station and can use external software as calculation engines. The following external model calculation tools can be employed:

- Mathworks Matlab R2008b
- Umetrics Simca-Q 12.0.1
- Camo The Unscrambler 9.8

Run-time data and results of calculations can be visualised in real-time. Hence, graphs are adapting to the latest available data.

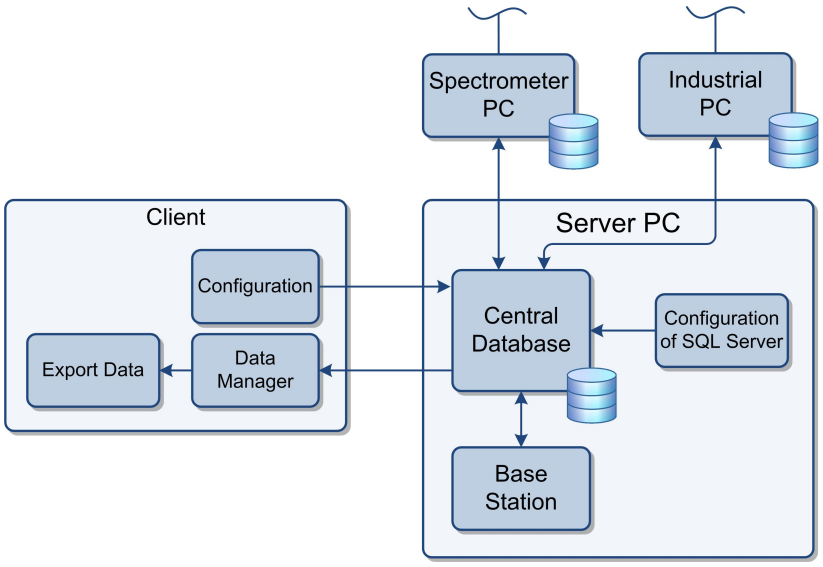



Figure 3.9: Data flow diagram for the configuration of SIPAT methods.

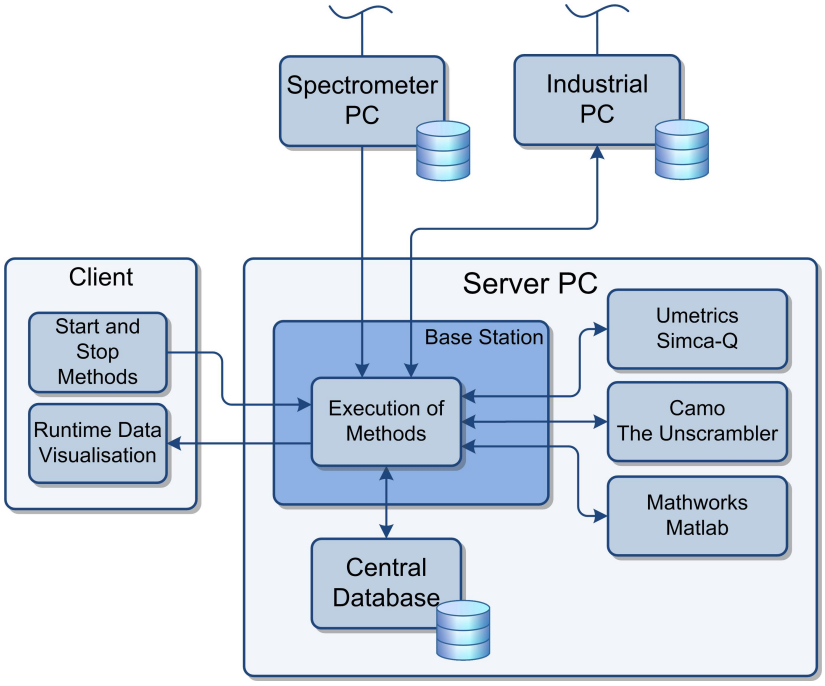



Figure 3.10: Data flow diagram for method execution.

## 3.6 Closed Loop via SIPAT

The use case as discussed in the chapters 4 presupposes a connection from Matlab via SIPAT to the process. This means, the use case requires a feedback connection via SIPAT to be able to control the process, as illustrated in Figure 3.11. The system contains a block representing WinCC Runtime, an internal controller, the HME process, SIPAT and the operator/user. The operator working directly at the HME process uses WinCC to modify input parameters, whereas the SIPAT user can manipulate the input parameters from every SIPAT Client PC. The intervention of SIPAT into the established system is based on OPC technology and has to be activated by a switch (in the same time it deactivates the control via WinCC Runtime).

Scalar and multi-value measurements (cf. Figure 3.11) can be merged to vectors depending on their purpose and availability for SIPAT or WinCC. The input parameters $\mathbf{r}$ may be defined as

$$\mathbf{r} := \begin{bmatrix} r_{ss} \\ \mathbf{r_c} \end{bmatrix} \text{ ... Input parameters,} \qquad \mathbf{r_x} := \begin{bmatrix} \mathbf{r_{xp}} \\ \mathbf{r_{xc}} \end{bmatrix} \text{ ... Configuration parameters,}$$

$$\mathbf{r_c} := \begin{bmatrix} r_{t2} \\ r_{t3} \\ \vdots \\ r_{t10} \\ r_f \end{bmatrix} \text{ ... Controller input parameters,} \tag{3.1}$$

where $\mathbf{r_{xp}}$ and $\mathbf{r_{xc}}$ denote general configuration parameters and the configuration parameters for the *proportional-integral-derivative* (PID) controllers for the barrel temperatures as illustrated in Figure 3.12. The inner control loop including the HME process and the internal controller is separated into one control loop for the barrel temperatures and one for the feed rate. The *programmable logic controller* (PLC) using PID controllers is responsible for controlling the barrel temperatures and also forwards the reference feed rate value to the feeder. The feedback connection for feed rate controlling is established within the feeder and does not need external hardware or software (e.g. PLC). Furthermore, the feeder also uses a PID controller.

The control variables for the barrel temperatures and the feed rate can be merged to

$$\mathbf{u} := \begin{bmatrix} \mathbf{u_t} \\ u_f \end{bmatrix} \text{ ... control variables,} \qquad \mathbf{u_t} := \begin{bmatrix} u_{t2} \\ u_{t3} \\ \vdots \\ u_{t10} \end{bmatrix} \text{ ... barrel temperature control variables,} \tag{3.2}$$

where the barrel temperature control variables $\mathbf{u_t}$ can be measured and displayed in WinCC

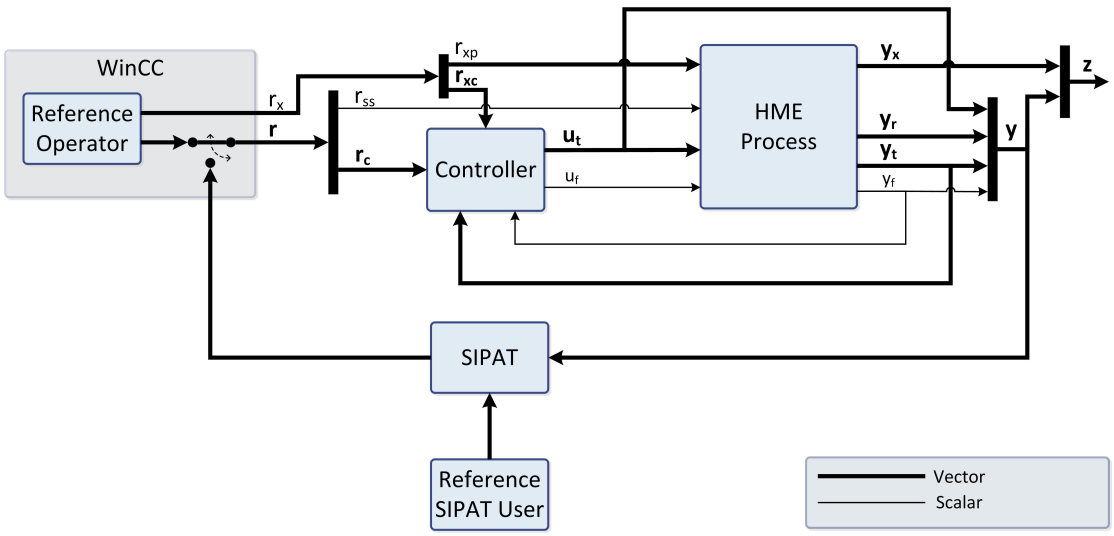



Figure 3.11: Control loop via SIPAT.

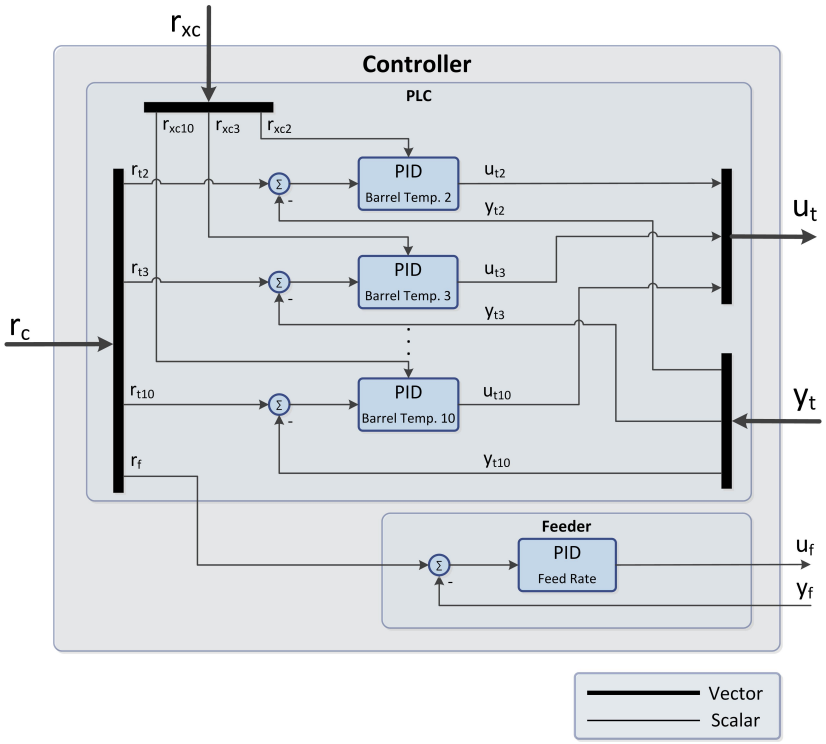



Figure 3.12: Inner controller for barrel temperatures and feed rate.

and SIPAT. The output parameters accessible for SIPAT are defined as

$$\mathbf{y} := \begin{bmatrix} \mathbf{y_r} \\ \mathbf{y_t} \\ y_f \\ \mathbf{u_t} \end{bmatrix} \text{ ... parameters accessible for SIPAT,}$$

$$\mathbf{y_r} := \begin{bmatrix} y_{mt} \\ y_{mp} \\ y_{md} \\ \mathbf{y_s} \end{bmatrix} \text{ ... process and material output parameters,}$$

$$\mathbf{y_t} := \begin{bmatrix} y_{t1} \\ y_{t2} \\ \vdots \\ y_{t10} \end{bmatrix} \text{ ... barrel temperature output parameters.} \tag{3.3}$$

WinCC can access more parameters, such as the internal parameters $\mathbf{y_x}$ (e.g. error messages), denoted as

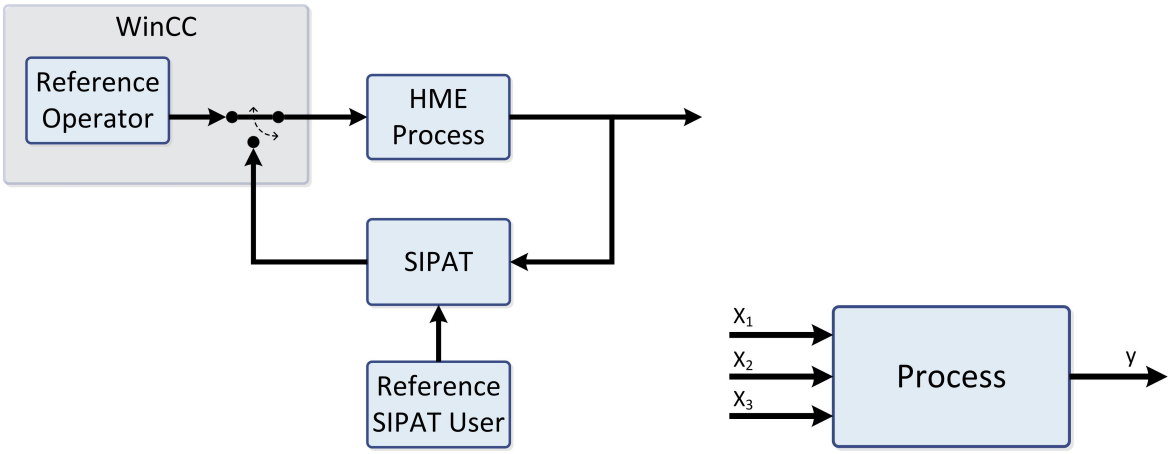
$$\mathbf{z} := \begin{bmatrix} \mathbf{y_x} \\ \mathbf{y} \end{bmatrix} \text{ ... output parameters accessible for WinCC.} \tag{3.4}$$

# 4 Use Case: Enable Design of Experiments

The HME process is a *multi-input multi-output* (MIMO) system. Thus, the extrusion process can be viewed as a 12 input x 16 output process. In order to develop a controller, it is desirable to reduce the dimensionality of the system. System analysis can be applied to identify and eliminate parameters with little or no correlation to product's quality characteristics of interest [17]. As a part of Quality by Design, an automatic DoE is applied on the HME process for process and product characterisation. To conduct such experiments, a feedback connection via SIPAT (Figure 4.1a) has to be established. Details about this procedure are pointed out in section 3.6.

## 4.1 Introduction to DoE and the Use Case

Design of Experiments is the methodology to design (plan) and statistically evaluate experiments. Therefore, a set of representative experiments is conducted in which all factors under investigation are varied simultaneously and systematically. Based on Figure 4.1b, this means the input factors $\mathbf{x}$ are varied while evaluating the responses $y$. The influential variables (factors) and the range of these factors can be determined to optimise the response.




(a) Schematic illustration of the control loop via SIPAT.

(b) Process with inputs $\mathbf{x}$ and one output $y$.

Figure 4.1: Illustration of the control loop and an example of a process with three inputs and one output.

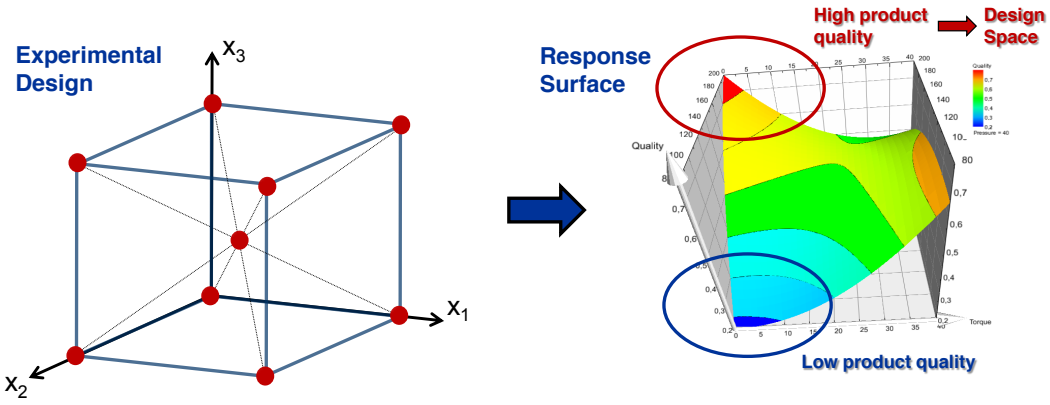



Figure 4.2: Development from the experimental design to the response surface.

Furthermore, the response variability should be minimised in order to improve process yield and reduce development time and the overall costs.

The aim of DoE is to maximise the information content from experimental series (relationship between inputs and outputs) while keeping the number of experiments low. As a consequence, the information on the relationship between inputs and outputs are statistically significant and effects of input variables on outputs are quantifiable.

Based on the dynamic DoE approach, the aim of this use case is to

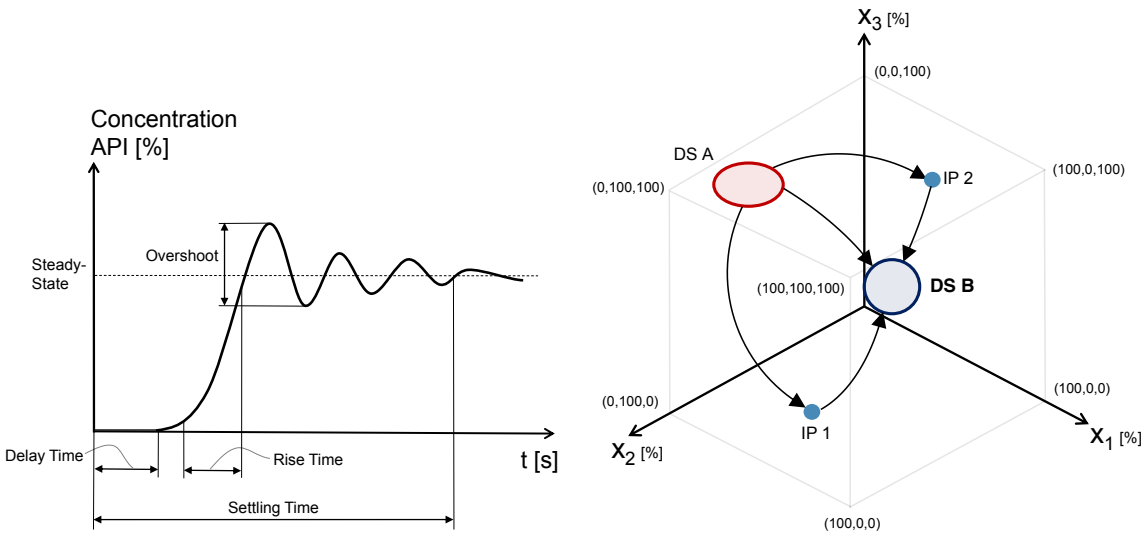
1. increase knowledge of the target product profile and get a product of intended quality (determination of a Design Space).
2. optimise process trajectories, e.g. from one formulation to another.

### 4.1.1 Determination of Design Space

Experiments have to be designed and afterwards performed to increase the knowledge of the TPP. Initially, the experimental objectives (i.e. purpose of the experiment), the variables (input factors) and the critical quality attributes as responses have to be defined. The input variables are varied simultaneously and in a systematic way. Thus, the experimental design as part of QbD contains the following steps:

1. Choose an experimental design (e.g. full factorial, fractional, d-optimal).
2. Conduct randomised experiments.
3. Analyse the data.
4. Create a multidimensional surface model.

Figure 4.2 illustrates the development from the experimental design to the response surface used for process understanding. Based on prior knowledge, the knowledge space represent the minimum and maximum values of the input factors in their experimentally investigated range. The knowledge space might contain regions producing unacceptable product quality.




(a) Step response including characteristic parameters.

(b) Trajectories from DS A to DS B.

Figure 4.3: Show a step response and trajectories concerning the use case.

Thus, an experimental design approach (in Figure 4.2 the full factorial design is used) has to be applied to determine the *design space* (DS). For example using the full factorial design: two levels per factor and one central point (red marked points) would be reached during the experiments to get detailed information on factors and interactions. This central point is defined as a standard reference experiment and new experiments are distributed in a symmetrical fashion. It requires relatively few runs per investigated factor and is the basis for all classical experimental designs. At a later stage of the experimental process, the central point may be determined by the already performed screening experiments and can be used for optimisation experiments [27, 28].

The response surface obtained from a regression model links the factor and the responses together and enables model interpretation, i.e. quality of fit/prediction. The model can be used to predict the best operating point for this set-up (screw configuration, formulation). However, investigation of the response surface plot results in a classification of high and low product quality regions. The high product quality region determines the design space based on these experiments. The design space specifies the multidimensional combination of input factors (e.g. screw speed, feed rate) that have been demonstrated to provide assurance of quality [27, 28, 29].

To sum up, the design space is defined out of the knowledge space after response surface modeling and performing a dynamic DoE. This procedure has to be performed for every new formulation and/or new screw configuration.

### 4.1.2 Trajectory Optimisation

The resulting design space using an experimental design approach as demonstrated in the previous section corresponds to only one specific formulation. Thus, various formulations result in different design spaces within the knowledge space. Adapting the manufacturing process from one product to another means manipulating the input parameters in order to approach the design space according to the new formulation. This trajectory can be optimised in respect to

- decrease the settling time, rise time, delay time, and/or overshoot (cf. Figure 4.3a),
- stabilise the process faster,
- increase the performance,
- reduce loss of product,
- decrease energy demand.

Comparing the trajectories (e.g. rise time, settling time) enables the estimation of an optimal trajectory. To optimise the result, the screening experiments can be used to define further experiments by inserting new IP(s).

Moreover, optimising trajectories also includes optimising the start-up of the process and finding the optimal order of product manufacture (e.g. starting with formulation B is better than with formulation A). To optimise the manufacture order, the experiments start at DS B, continue via IP(s) (e.g. using the optimal IPs obtained from previous experiments) and finally reach the DS A. Therefore, the experiments with the same IP(s), but different starting DSs have to be compared and will deliver the optimal order. Furthermore, optimising the start-up can be carried out by starting at the rest position of the process and then by approaching the desired DS via different IP(s).

## 4.2 Implementation

This use case involves Matlab, SIPAT, the HME process, and may also include Simca-Q. Figure 4.4 shows the basic procedure in order to manipulate the input feed rate. This can be easily adapted to another input parameter or expanded to multiple input parameters. The input sequences which determines the knowledge space for several input parameters might be stored in a `MAT`-file. Basically SIPAT calls a Matlab function which delivers the current samples. Moreover, SIPAT distributes the input samples to the appropriate OPC tags. In order to investigate the influence of input parameters on the product quality, the observed spectrum has to be intepreted as a concentration. This is performed in real-time by Simca-Q; the principle of the prediction of a concentration is discussed in section 2.2.3. After a successful experiment, the observed data can be analysed off-line e.g. in Matlab, where both input sequences and measurements including the predicted concentration are available time-aligned.

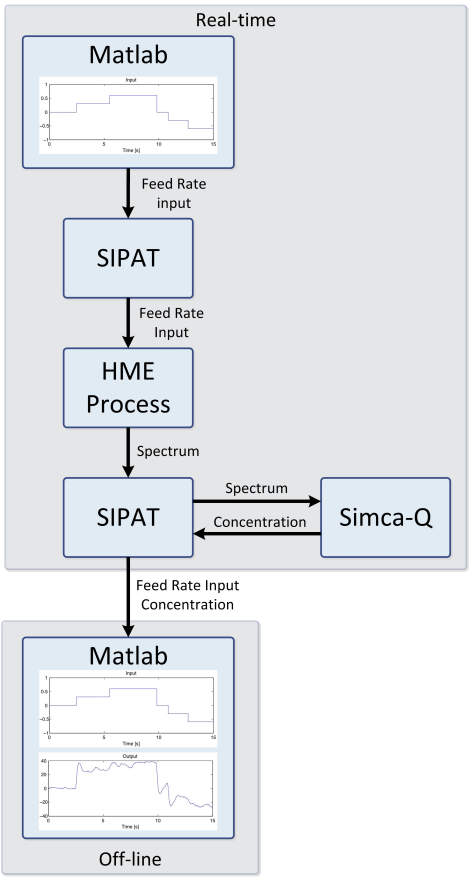



Figure 4.4: Basic procedure of an automatic DoE manipulating the input feed rate and observing the spectra.

The data flow diagram in Figure 4.5 shows the interactions of SIPAT, the NIR spectrometer, the Instrual PC, Simca-Q, the Client, and the process itself. The spectrometer PC delivers multi-value measurements (spectra), whereas the Industrial PC provides SIPAT via an OPC server scalar measurements (e.g. screw speed, barrel temperatures, feed rate). The base station executes methods and thereby it carries out the DoE method. An external PC (e.g. client) as Matlab SIPAT PC or the base station (a prerequisite is the execution of the SIPAT Matlab Service) is used for the execution of the Matlab scripts.

**Determination of Design Space:** Matlab via SIPAT is used to execute the automatic DoE. The Matlab script (DoE Matlab script) contains the data points and an execution schedule. A worksheet defines the execution schedule of the different experiments, e.g. lists all experiments in a structured way and then chooses a random run order. The generation of the input sequences has to be carried out off-line.

**Trajectory Optimisation:** Based on the automatic DoE approach new experiments are attached to the worksheet. As illustrated in Figure 4.3b, intermediate points (IPs) are

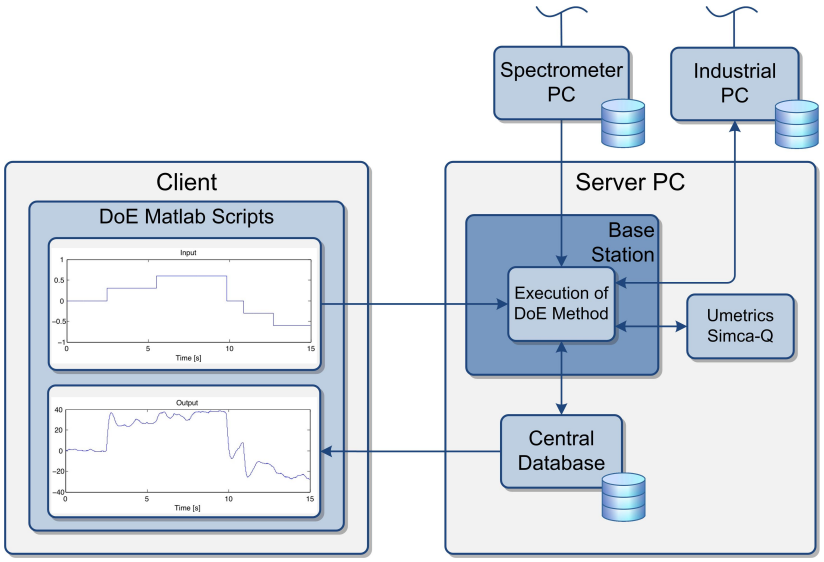



Figure 4.5: Data flow diagram for the DoE use case.

inserted to specify appropriate trajectories. Experiments are performed by starting from DS A via various IP(s) (e.g. IP 1 and IP 2) to the final DS B. All measurements are recorded by the DoE method and can be accessed from the SIPAT database for off-line analysing in order to optimise the trajectories.

# 5 Use Case: Modelling the HME Process

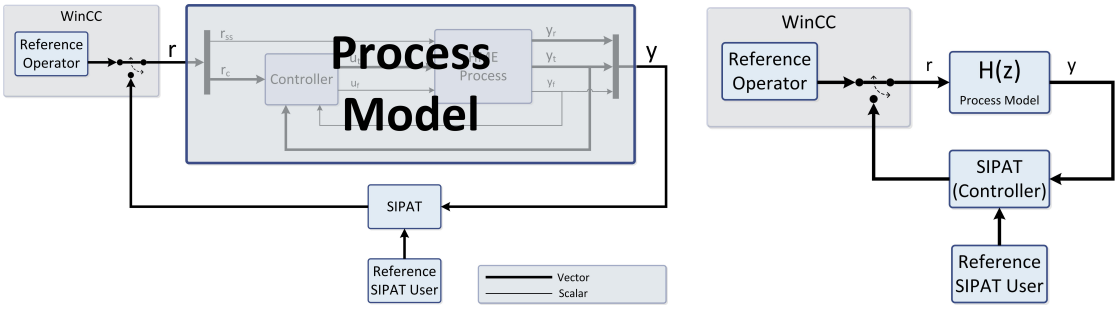
This chapter examines the identification of the HME process using SIPAT and Matlab. The first section introduces the identified model; section 5.3 presents the results of preliminary experiments and deals with the planning of the second stage experiments and gives an introduction to the modelling procedure. The system identification results are presented in section 5.4; in the last section the estimated model is used as basis for the development of a M-step ahead predictor.

## 5.1 Introduction to the Use Case

The identification of a linear time-invariant system is the basis for the development of a predictor or controller for this complex process. As shown in Figure 5.1a, the black box model (process model) containing the HME process and the internal controller (Figure 3.12) from $m$ inputs $\mathbf{r}$ to $n$ outputs $\mathbf{y}$ can be described by equation 5.1.

$$\mathbf{Y}(\mathbf{z}) = \mathbf{H}(\mathbf{z})\mathbf{R}(\mathbf{z}) \tag{5.1}$$

Taking the Z-transform of the input $\mathbf{r}$ and the output signal $\mathbf{y}$ yields to $\mathbf{R}(\mathbf{z})$ and $\mathbf{Y}(\mathbf{z})$. The system is a MIMO system and therefore the $m$ x $n$ discrete-time transfer function matrix $\mathbf{H}(\mathbf{z})$ (Figure 5.1b) contains a transfer function from every input to every output. Its $(k, j)$-element, $H_{k,j}(z)$, describes how the input $r_j$ influences the output $y_k$. Only varying the input $r_j$ and observing the output $y_k$ results in a *single-input single-output* (SISO) system with the transfer function $H_{k,j}(z)$ [30]. As mentioned in chapter 4, the dimensionality of the system has to




(a) The process model containing the internal controller and the HME process.

(b) The transfer function matrix $\mathbf{H}(\mathbf{z})$ representing the process model.

Figure 5.1: Modelling of the HME process.

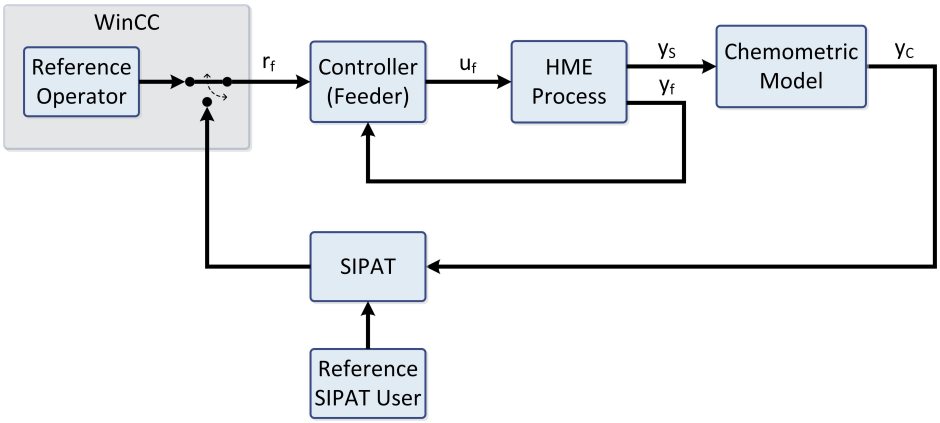



Figure 5.2: Simplification of the HME process.

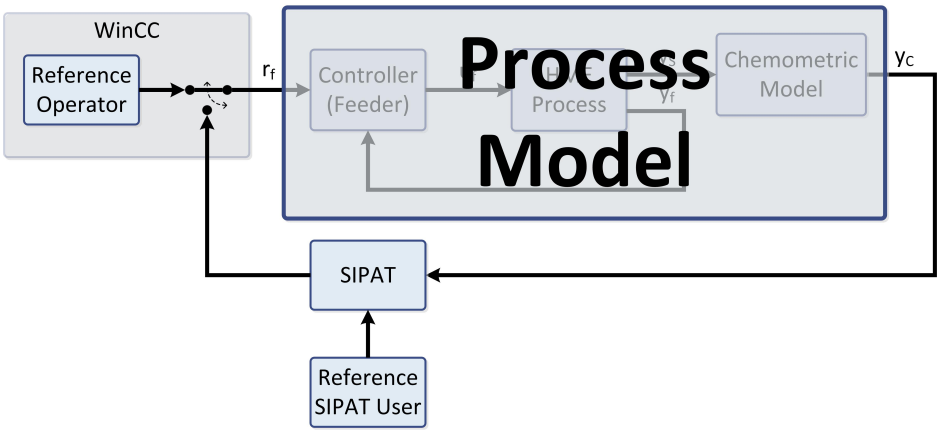



Figure 5.3: The process model contains the internal controller of the feeder, the HME process and the chemometric model.

be reduced. This results in a reduction of the size of the matrix $\mathbf{H}(\mathbf{z})$ and so fewer transfer functions have to be estimated.

This chapter deals with the identification of a SISO system, i.e. the process model is simplified from $m$ inputs and $n$ outputs to one input and one output. The input is chosen as the feed rate of the feeder containing the API, and the output is the measured API concentration. For this model the API concentration in the extrudate located at the die is denoted as measured, but it cannot be measured directly. As discussed in section 3.3.2, the spectrum can be measured and has to be converted into an API concentration. This mapping is carried out by an appropriate model, namely the chemometric model as depicted in Figure 5.2 (more details are given in section 5.2.1). As a consequence, the identification of the process model presupposes an accurate chemometric model.

The process model contains the internal controller of the feeder, the HME process and the chemometric model and therefore, it is a model from the reference feed rate $r_f$ to the API concentration $y_c$, as illustrated in Figure 5.3.

Obtaining good experimental results imposes several conditions and data fulfilling the prerequisites of the identification methods. The choice of the input signal relies on the criteria of sufficient excitation for determination of the spectral range of validity. Therefore, the excitation signal is of great importance for the outcome of the identification and is discussed in more detail in section 5.3.2 [31]. This excitation signal can be generated in Matlab and directly applied to the process via SIPAT.

The aims of modelling the HME process are:

- The knowledge of the feeding process and the impact of disturbances on the final product quality can be increased.
- This model could act as a basis for the control of the feed rate to get a product of intended quality.
- The implementation of a M-step ahead predictor presupposes a accurate process model.

## 5.2 Introduction to the Models

This section introduces the required models. Firstly, properties of the chemometric model are described and an identification method to develop an accure model is examined. Secondly, various different types of process models are presented. We start with the ARMAX model, followed by the Box-Jenkins model, and finally a state-space model are introduced.

### 5.2.1 Chemometric Model

The general purpose of the chemometric model is to extract necessary information of the observed spectrum to be able to predict the API concentration. Thus, the model has to be calibrated in respect to the output data set, the known API concentrations, and the input data set, the observed spectra. This procedure is called the calibration or identification stage and is illustrated in Figure 5.4a.

The model calibration could be carried out by applying split feeding or using various predefined premix to be able to vary the API concentration. The accuracy of the model is highly correlated to a well defined experimental setup. This implies a well adapted location of the Dynisco probe (more details are given in section 2.2.3) and in particular for split feeder; a robust and adjusted mechanical construction of the feeders guarantees reproducible results (more details are presented in section 3.3.2).

#### Identification Stage

The trainings data matrix $\mathbf{\Phi_{CM}}$ contains N data points with a scalar as input data and an $m$-dimensional vector as output data associated to one data point. The spectrometer delivers a spectrum in the wavelength range from $1100nm$ to $2200nm$ and a resolution of

$2nm$. Consequently, the output data is a vector with 551 elements ($m = 551$) and therefore, the trainings data matrix can be written as

$$\mathbf{\Phi_{CM}} = \begin{bmatrix} y_{c_1} & \mathbf{y_{s_1}} \\ y_{c_2} & \mathbf{y_{s_2}} \\ \vdots & \vdots \\ y_{c_N} & \mathbf{y_{s_N}} \end{bmatrix} \quad \mathbf{y_{s_i}} = \begin{bmatrix} y_{I_{1100}} \\ y_{I_{1102}} \\ \vdots \\ y_{I_{2200}} \end{bmatrix} \text{ with } i = 1, \ldots, N \tag{5.2}$$

with e.g. the measured intensity $y_{I_{1100}}$ at a wavelength of $1100nm$.

Apparently, one spectrum contains $m$ highly correlated variables and the significant information could be represent by $n_{PC}$ independent variables. The concept of *Principal Component Analysis* (PCA) applied on the observed data extracts the essential information. This method is based on the calculation of the eigenvectors, also called principal components, of the observed spectra. An appropriate selection of the principal components is used to describe a spectrum. However, PCA can be used to analyse the data set and is the basis of the *Partial Least Square Regression* (PLS-R) model. Amongst other, the following advantages of applying PLS-R instead of e.g. *Multiple Linear Regression* (MLR) can be given:

- The input variables can be highly correlated.
- There may be more input variables than measurements.

These advantages are very significant when using spectra to predict a scalar quality parameter, e.g. the API concentration. The PLS algorithm utilises the PCA approach to find fundamental relations between the input and output data set [6, 32]. Therefore, latent variables model the covariance structure of the trainings data set $\mathbf{\Phi_{CM}}$. The PLS-R model is depicted in equation 5.3.

$$y_k = b_0 + b_1 x_{1_k} + b_2 x_{2_k} + \cdots + b_{n_{PC}} x_{n_{PC_k}} + e_k \tag{5.3}$$

The scores $x_{1_k}$ to $x_{n_{PC_k}}$ are obtained by projecting the $k$ input data (spectrum) on the latent variables and the model error is denoted as $e_k$.

### Prediction Stage

After a successful model calibration, the model can be used for predictions of unknown chemical compositions considering the same materials as used in the identification stage. The PLS model, as denoted in equation 5.3, is used to calculate the unknown API concentration.

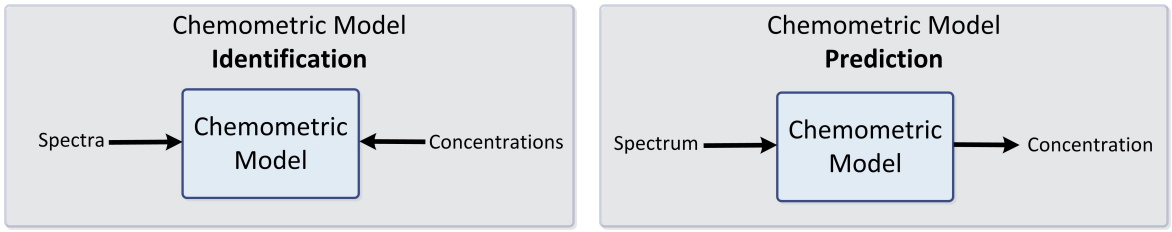
To sum up, the chemometric model represents the model from the measured spectrum $y_s$ to the API concentration $y_c$ and thus, the model converts the spectrum to a concentration, as depicted in Figure 5.4b.

### 5.2.2 Process Model

The process model, as defined in the introduction of this chapter includes the HME process, the chemometric model and the K-Tron gravimetric feeder with its internal controller, which is responsible for minimising the error between the reference feed rate $r_f$ and the measured feed rate $y_f$. Thus, modelling the process implies some challenges:

- Feeder is very inaccurate for small feed rates (e.g. 0.1 kg/h).
- A mathematical description of the HME process is very complex and contains nonlinear parameters.
- The development of the process model highly depends on the chemometric model.

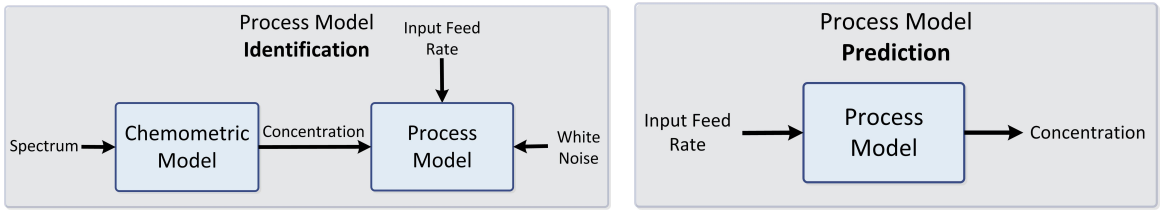
Basically, high frequency perturbations of the feed rate are dampen out by the backmixing ability of the extruder, whereas low frequency disturbances influence the final product quality [33]. The HME process dynamics are nonlinear in respect to the API concentration. E.g. paracetamol (API) is incorporated into calcium stearate (matrix): the presence of paracetamol increases the viscosity of the extrudate and results in a nonlinear behaviour depending on the API concentration [34]. Moreover, as described in section 2.2.3, the residence time distribution depends on the applied kneading elements. Thus, to achieve a final TPP at least a few kneading elements have to be applied resulting in a broader residence time distribution and consequently the process time delay (residence time) varies. However, the identification procedure has to consider these challenges.




(a) Identification stage with spectra and the reference concentration as input.

(b) Prediction stage with spectrum as input and the concentration as output.

Figure 5.4: Identification and prediction stage of the chemometric model.




(a) Identification stage with the feed rate and the reference concentration as input.

(b) Prediction stage with the feed rate as input and the concentration as output.

Figure 5.5: Identification and prediction stage of the process model

The process model from the input feed rate $r_f$ to the API concentration $y_c$ can be described by different kind of models. In the identification stage, an ARMAX and a Box-Jenkins model derived as discrete-time transfer functions and a model in discrete-time state-space representation using a subspace-based identification procedure are described.

### Identification Stage

For the process model identification the trainings data set $\mathbf{\Phi_{PM}}$ is composed by the input reference feed rate and the API concentration obtained from the chemometric model. The SISO model (one input and one output) is illustrated in Figure 5.5a and the training set can be described as

$$\mathbf{\Phi_{PM}} = \begin{bmatrix} r_{f_1} & y_{c_1} \\ r_{f_2} & y_{c_2} \\ \vdots & \vdots \\ r_{f_N} & y_{c_N} \end{bmatrix} . \tag{5.4}$$

**ARMAX Model:** Linear models for general purpose representing signals and noise are often modeled by means of *autoregressive moving average models with external input* (ARMAX). Describing the model error as a moving average of white noise adds flexibility to model disturbances influencing the process. The ARMAX model is given by

$$\begin{aligned} y_k = &- a_1 y_{k-1} - a_2 y_{k-2} - \cdots - a_{n_A} y_{k-n_A} + \\ &b_0 r_{k-d} + b_1 r_{k-d-1} + b_2 r_{k-d-2} + \cdots + + b_{n_B} r_{k-d-n_B} + \\ &w_k + c_1 w_{k-1} + c_2 w_{k-2} + \cdots + + c_{n_C} w_{k-n_C}, \end{aligned} \tag{5.5}$$

where $w_k$ is a stochastic white noise process with a variance $\sigma_w^2$ and the following properties:

$$\begin{aligned} \mathcal{E}\{w_k\} &= 0 \\ \mathcal{E}\{w_i w_j^T\} &= \delta_{ij}\sigma_w^2 \end{aligned} \tag{5.6}$$

with $\mathcal{E}\{\cdot\}$ and $\delta_{ij}$ respectively are the mathematical expectation operator and the Dirac function. Equation 5.5 can be rewritten in the following form

$$A(z^{-1})y_k = z^{-d}B(z^{-1})r_k + C(z^{-1})w_k \tag{5.7}$$

with the polynomials $A(z^{-1})$, $B(z^{-1})$ and $C(z^{-1})$ defined as

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-1} + \cdots + a_{n_A} z^{-n_A} \tag{5.8}$$

$$B(z^{-1}) = b_0 + b_1 z^{-1} + b_2 z^{-1} + \cdots + b_{n_B} z^{-n_B} \tag{5.9}$$

$$C(z^{-1}) = 1 + c_1 z^{-1} + c_2 z^{-1} + \cdots + c_{n_C} z^{-n_C}. \tag{5.10}$$

Moreover, the ARMAX model involves a dynamic model $G(z^{-1})$ and a disturbance model $H(z^{-1})$ which can be extracted from equation 5.7 in the following way

$$y_k = z^{-d}\frac{B(z^{-1})}{A(z^{-1})}r_k + \frac{C(z^{-1})}{A(z^{-1})}w_k \tag{5.11}$$

$$y_k = z^{-d}G(z^{-1})r_k + H(z^{-1})w_k \ . \tag{5.12}$$

Figure 5.6a gives a graphical representation of the ARMAX model in respect to the equations 5.11 and 5.12. One limitation of the ARMAX model is the property of having the same poles in the dynamic model and the disturbance model. This means for dominating disturbances acting early in the process (e.g. feeder variations) the ARMAX model is useful, whereas for disturbances entering late in the process (e.g. measurement noise) this model structure is limited.

Fitting the input-output data by the ARMAX model results in an identification of the following unknown parameters

$$\theta = \begin{bmatrix} a_1 & a_2 & \cdots & a_{n_A} & b_0 & b_1 & \cdots & b_{n_B} & c_1 & c_2 & \cdots & c_{n_C} \end{bmatrix}^T . \tag{5.13}$$

Consequently the calculation of the new output can be written as

$$\hat{y}_k = \phi_{\mathbf{k}}^{\mathbf{T}}\hat{\theta} \ , \tag{5.14}$$

with the estimated parameter vector $\hat{\theta}$ and the regressor vector defined as

$$\phi_{\mathbf{k}} = \begin{bmatrix} -y_{k-1} & \cdots & -y_{k-n_A} & r_{k-d} & \cdots & r_{k-d-n_B} & w_{k-1} & \cdots & w_{k-n_C} \end{bmatrix}^T . \tag{5.15}$$

This parameter identification can be carried out by Matlab predefining the process time delay $d$ and the orders $n_A$, $n_B$ and $n_C$ of the polynomials $A(z^{-1})$, $B(z^{-1})$ and $C(z^{-1})$, respectively. Basically a loss function $V$ is minimised to obtain the unknown parameters. Assuming the noise $w_k$ have a Gaussian or normal distribution

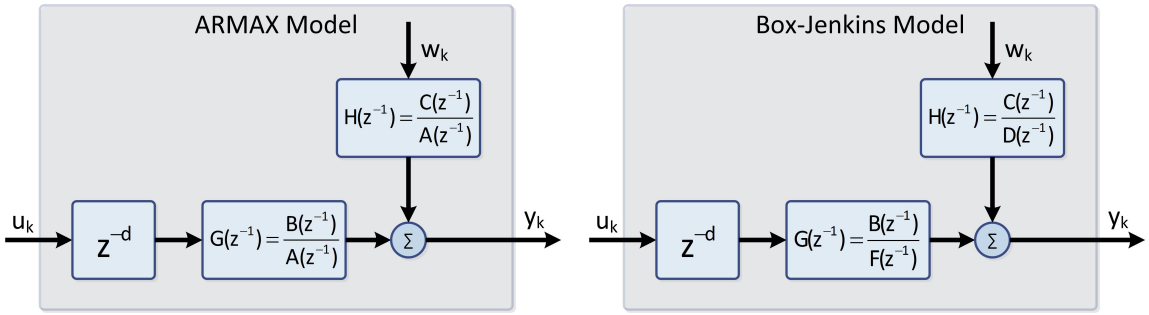
$$f_w(x) = \frac{1}{\sqrt{2\pi\sigma_w^2}}e^{-\frac{x^2}{2\sigma_w^2}} \ , \tag{5.16}$$

where $x$ is a random variable and the variance and mean are defined in equation 5.6. Therefore, the loss function $V$ can be expressed as

$$V(\hat{\theta}) = \frac{1}{2}\sum_{k=1}^{N}(y_k - \hat{y}_k) = \frac{1}{2}\sum_{k=1}^{N}\left(y_k - \phi_{\mathbf{k}}^{\mathbf{T}}\hat{\theta}\right) \tag{5.17}$$

An iterative procedure such as Newton-Raphson can be used to estimate the parameters. Additionally the identification of the parameters $\theta$ provides an estimate of the variance

$$\hat{\sigma}_w^2 = \frac{2}{N}V(\hat{\theta}) \ . \tag{5.18}$$




(a) ARMAX Model (b) Box-Jenkins Model

Figure 5.6: Comparison of the ARMAX and Box-Jenkins model.

**Box-Jenkins Model:** This type of model is similar to an ARMAX model, but offers more flexibility to model disturbances interfering the process model output. The structure of the Box-Jenkins can be expressed as

$$y_k = z^{-d}\frac{B(z^{-1})}{F(z^{-1})}r_k + \frac{C(z^{-1})}{D(z^{-1})}w_k \tag{5.19}$$

with the same stochastic white-noise properties as denoted in equation 5.6 and the polynomials $B(z^{-1})$ and $C(z^{-1})$ as defined in equations 5.9, 5.10 and $D(z^{-1})$, $F(z^{-1})$ denoted as

$$D(z^{-1}) = 1 + d_1 z^{-1} + d_2 z^{-1} + \cdots + d_{n_A} z^{-n_D} \tag{5.20}$$

$$F(z^{-1}) = 1 + f_1 z^{-1} + f_2 z^{-1} + \cdots + f_{n_C} z^{-n_F}. \tag{5.21}$$

The Box-Jenkins model is an extension of the ARMAX model and also describes an dynamic model $G(z^{-1})$ and a disturbance model $H(z^{-1})$ written in equation 5.23 and represented in Figure 5.6b.

$$y_k = z^{-d}\frac{B(z^{-1})}{F(z^{-1})}r_k + \frac{C(z^{-1})}{D(z^{-1})}w_k \tag{5.22}$$

$$y_k = z^{-d}G(z^{-1})r_k + H(z^{-1})w_k\ . \tag{5.23}$$

The identification of a Box-Jenkins model is more complex than of an ARMAX model, since more unknown parameters have to be estimated.

$$\theta = \begin{bmatrix} f_1 & f_2 & \cdots & f_{n_F} & b_0 & b_1 & \cdots & b_{n_B} & c_1 & c_2 & \cdots & c_{n_C} & d_1 & d_2 & \cdots & d_{n_D} \end{bmatrix}^T \tag{5.24}$$

The Box-Jenkins model can be identified using Matlab by predefining the process time delay $d$ and the orders $n_F$, $n_B$, $n_C$ and $n_D$. The algorithm to obtain the unknown parameters is similar to the approach used for the identification of an ARMAX model. Moreover, the variance $\sigma_w^2$ of the white noise process can also be estimated.

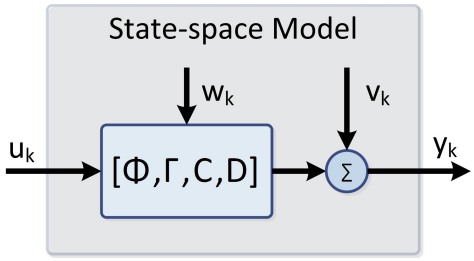



Figure 5.7: Identification of the state-space model.

**State-space model Identification:** This identification procedure directly estimates the matrices of a state-space realisation. Due to the influence of disturbances on the process and also on the output (measurement noise), an state-space model in the form of the system 5.25 has to be estimated. Figure 5.7 illustrates the state-space model identification problem.

$$\begin{aligned} \mathbf{x_{k+1}} &= \mathbf{\Phi x_k} + \mathbf{\Gamma r_k} + \mathbf{N}w_k \\ \mathbf{y_k} &= \mathbf{C x_k} + \mathbf{D r_k} + \mathbf{v_k} \end{aligned} \tag{5.25}$$

The zero-mean white noise sequence has to be statistically independent from the input $u_k$ and is given by the covariance matrix

$$\mathcal{E}\left\{ \begin{bmatrix} w_i \\ v_i \end{bmatrix} \begin{bmatrix} w_j^T & v_j^T \end{bmatrix} \right\} = \begin{bmatrix} R_1 & R_{12} \\ R_{12}^T & R_2 \end{bmatrix} \delta_{ij}. \tag{5.26}$$

The identification procedure requires that the pair $\left( \mathbf{\Phi} \quad \begin{bmatrix} \mathbf{\Gamma} & R_1^{\frac{1}{2}} \end{bmatrix} \right)$ is controllable and the pair $\begin{pmatrix} \mathbf{C} \\ \mathbf{\Phi} \end{pmatrix}$ is observable [35]. Controllability and observability describe the influence of inputs on the states of the system and how they show up in the output. Thus, if all states are controllable, the system is said to be controllable. The theorem of controllability as defined in [30] can be adapted to this system as follows:

*The controllable states of the system 5.25 form a linear subspace, viz, the range of the matrix (the controllability matrix)*

$$\mathcal{S}\left(\mathbf{\Phi}, \mathbf{\Gamma}'\right) = \begin{bmatrix} \mathbf{\Gamma}' & \mathbf{\Phi\Gamma}' & \mathbf{\Phi}^2\mathbf{\Gamma}' & \dots & \mathbf{\Phi}^{n-1}\mathbf{\Gamma}' \end{bmatrix} \text{ with } \mathbf{\Gamma}' = \begin{bmatrix} \mathbf{\Gamma} & R_1^{\frac{1}{2}} \end{bmatrix} \tag{5.27}$$

*where $n$ is the order of the system. The system is thus controllable if and only if $\mathcal{S}$ has full rank.*

Moreover, the observability definition can be adjusted from [30]:

*The unobservable states constitute a linear subspace, viz, the null space of the matrix (the*

*observability matrix)*

$$\mathcal{O}(\mathbf{\Phi}, \mathbf{C}) = \begin{bmatrix} \mathbf{C} \\ \mathbf{C\Phi} \\ \vdots \\ \mathbf{C\Phi}^{n-1} \end{bmatrix} . \tag{5.28}$$

*The system is thus observable if and only if $\mathcal{O}$ has full rank.*

Matlab is used to identify the state-space model based on a subspace-based identification algorithm. The order $n$ of the system has to be chosen to get the quadruple $\begin{bmatrix} \mathbf{\Phi} & \mathbf{\Gamma} & \mathbf{C} & \mathbf{D} \end{bmatrix}$ [35]. This algorithm also computes the vector $\mathbf{N}$ for the disturbance model.

#### Prediction Stage

The model maps the input feed rate to a concentration considering process dynamics and the process time delay. Depending on the used algorithm to derive a discrete-time transfer function estimate or a discrete-time state-space realisation, the model can be used for simulation or prediction. A graphical representation of the prediction stage is given in Figure 5.5b.

## 5.3 Experiments

Some information is required to plan good, precise and successful experiments and therefore it is necessary to make iterated experiments. Initially, first stage experiments as discussed in section 5.3.1 are performed wich are followed by second stage experiments. Section 5.3.2 presents requirements on the excitation signal and introduces a certain input sequence.

### 5.3.1 First Stage Experiments

First, the purpose of the model has to be clarified. Second, the design of an identification procedure which also includes the choice of the manipulated variables and when and where which signals are measured.

The initial requirement of this model is to increase the process knowledge. In further sections, the model is also used for simulation and prediction and therefore the identification procedure was reperformed. Due to the limited variables accessible for SIPAT, the input feed rate is chosen as the manipulated variable. Furthermore, this variable directly influences the API concentration in the extrudate.

Moreover, SIPAT is used as the computer-based data acquisition system. Here a certain sampling period and an aggregation function can be configured. The aggregation function is carried out by SIPAT to enable time-alignment of all measured input variables from various collector stations. To avoid loss of information regarding this aggregation function, the sampling period should be larger than the largest of the collector stations (extruder and

spectrometer). The spectrometer can deliver the measurements within 2 and 3 seconds, whereas the extruder provides the data faster. Furthermore, the sampling rate should be a trade-off between noise reduction and relevance for the dynamics [31, 36]. Consequently, the sampling period is chosen as 5 seconds.

In the following section an investigation and interpretation of the input/output relation are carried out. For all experiments the throughput of the extruder was chosen as 0.6 $kg/h$, which turned out to be a good choice for split feeding experiments. Calcium stearate (CaSt) is used as a matrix that is generally used as a lubricant in tablets and capsules. The API, paracetamol, incorporated in CaSt by the HME process successfully prepares the extrudate for a hot-strand cutter with two rotating knives as downstreaming process [34]. Previous experiments with this set-up enable basic understanding of the process and its configuration, but deeper knowledge, e.g. residence time, time constants, etc., is missing. Thus, the identification of the HME process in respect to a certain formulation is crucial to guarantee a final TPP.

Basically the first stage experiments involve simple experiments such as

- Step responses of available inputs (e.g. feed rate).
- Step disturbances (e.g. external influence of the feeder).

Here, the focus is on a basic qualitative understanding of the system and investigation of the causal relations between inputs and outputs. Furthermore, the process time delay, dominating time constants and, coherence spectrum can be evaluated.

### Analysing the Step Response

Various steps were performed while keeping a constant throughput. An increasing of the feed rate of the API feeder implicates decreasing the feed rate of the matrix feeder. Figure 5.8 illustrates the input feed rate and the API concentration as a function of the relative time.

By visually investigating the step responses, one may conclude:

- There is a process time delay.
- Settling time (including process time delay and rise time) is approximately 900 seconds.
- The process response is in a range of the input feed rate from 0 and 0.36 linear and thus a linear time-invariant model could be valid in this range.
- Due to the noisy process the process model has to include a disturbance model.

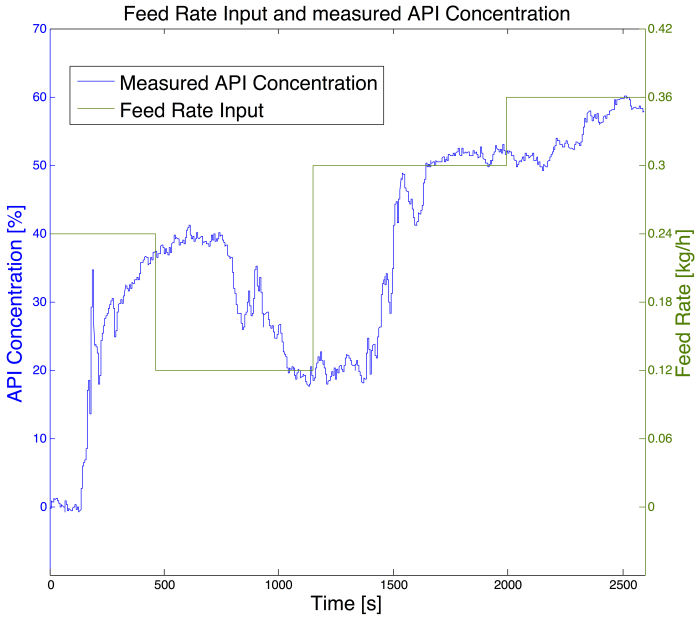



Figure 5.8: Input feed rate and the corresponding measured API concentration.

**Process Time Delay:** The correlation between the input feed rate and the API concentration provides an estimated process time delay (marked in Figure 5.9b). Considering the cross-covariance function

$$\hat{C}_{r_f y_c}(\tau) = \frac{1}{N-\tau} \sum_{k=\tau+1}^{N} r_k y_{f_{k-\tau}} \ , \tag{5.29}$$

the cross-correlation function is defined as

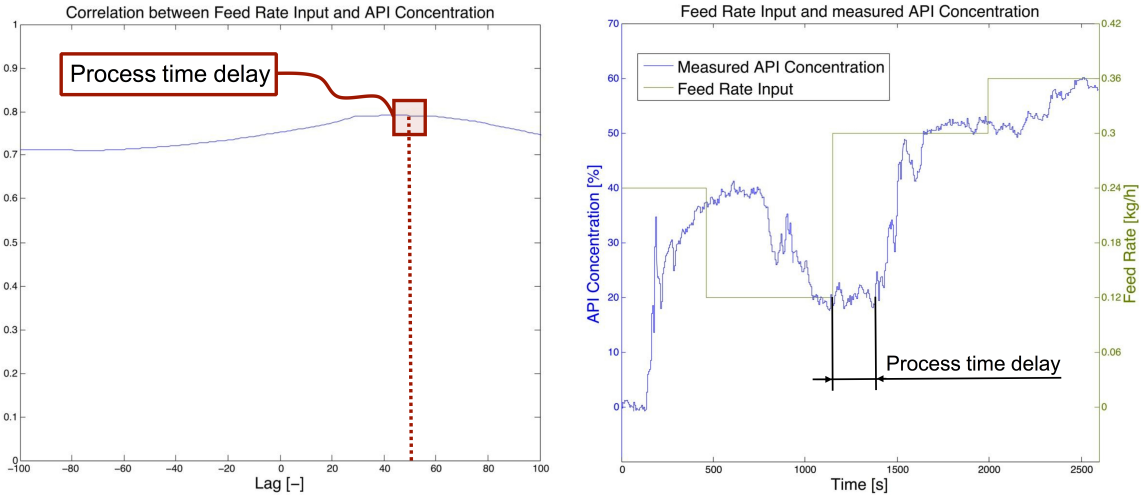
$$r_{r_f y_c} = \frac{1}{\hat{C}_{r_f y_c}(0)} \left[ \ \hat{C}_{r_f y_c}(1) \quad \dots \quad \hat{C}_{r_f y_c}(m) \ \right]^T \ . \tag{5.30}$$

Figure 5.9b illustrates the cross-correlation function $r_{r_f y_c}$ selecting $m$ as 100. The maximum of 46 belongs to the process time delay. With a sampling period of 5 seconds this results in a process time delay of 230 seconds.

**Coherence Spectrum Analysis:** The coherence spectrum expresses the degree of linear correlation in the frequency domain between input $r_f$ and the output $y_c$. The real API concentration $z_{c_k}$ is corrupted by a variable $v_k$ and results in the observed API concentration expressed as

$$y_{c_k} = z_{c_k} + v_k = g_k * r_{f_k} + v_k \ , \tag{5.31}$$

where $g_k * r_{f_k}$ denotes the discrete convolution between the impulse response $g_k$ and the input feed rate $r_{f_k}$. The hidden API concentration $z_{c_k}$ and the noise $v_k$ are not available to measure




(a) Correlation Analysis

(b) Input feed rate and API concentration as a function of time.

Figure 5.9: Analysing the process time delay.

and therefore the quadratic coherence spectrum between the input feed rate and the API concentration can be defined as the ratio

$$\gamma^2_{r_f y_c}(\omega) = \frac{|S_{r_f y_c}(j\omega)|^2}{S_{r_f r_f}(j\omega) S_{y_c y_c}(j\omega)} \ , \tag{5.32}$$

with the power cross spectrum $S_{r_f y_c}(j\omega)$, the input autospectrum $S_{r_f r_f}(j\omega)$, and the output autospectrum $S_{y_c y_c}(j\omega)$. The quadratic coherence always takes a value between $0 \leq \gamma^2_{r_f y_c}(\omega) \leq 1$. Rewriting equation 5.32 and substituting the cross power spectrum and the output autospectrum in respect to the linear input/output relationship (5.31) emphasises the properties of the coherence spectrum as derived in equation 5.33.

$$\begin{aligned} \gamma^2_{r_f y_c}(\omega) = \frac{|S_{r_f y_c}(j\omega)|^2}{S_{r_f r_f} S_{y_c y_c}(j\omega)} &= \frac{|G(j\omega)|^2 S_{r_f r_f}(j\omega)^2}{S_{r_f r_f}(j\omega)\left(|G(j\omega)|^2 S_{r_f r_f}(j\omega) + S_{vv}(j\omega)\right)} \\ &= \frac{1}{1 + \frac{S_{vv}(j\omega)}{S_{r_f r_f}(j\omega)|G(j\omega)|^2}} \end{aligned} \tag{5.33}$$

Applying the *discrete Fourier transform* (DFT) on the impulse response $h_k$ results in the transfer function $G(j\omega)$, and the noise autospectrum is denoted as $S_{vv}(j\omega)$ [37].

One can conclude from equation 5.33 that a value close to one indicates a low noise level ($S_{vv} \ll S_{r_f r_f}$) and there is a linear response between input and output of the type 5.31 [31, 36]. The coherence analysis can be used to get information about:

- nonlinearities,
- disturbance magnitudes,

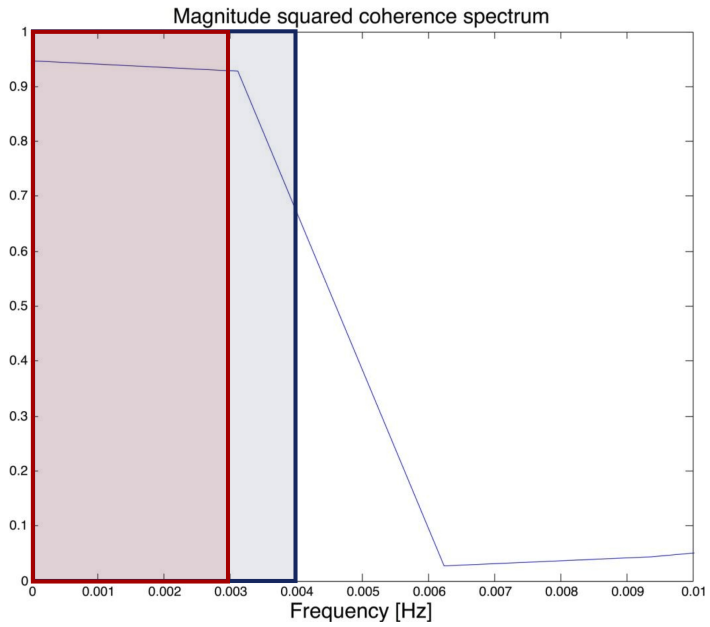



Figure 5.10: Coherence spectrum analysis to determine spectral properties of the data.

- spectral properties (e.g. periodic disturbances),
- the frequency range of such a linear model would be valid.

Based on the theoretical discussion and the coherence spectrum derived from the experimental data (Figure 5.10), the following conclusions can be drawn:

- The frequency range is limited by the process time delay: A time delay of 230 seconds results in a maximum frequency of 0.004 Hz (blue area).
- Up to approximately 0.003 Hz (red area): the noise level is low and a linear model could be valid in this range.

### 5.3.2 Planning of Second Stage Experiments and the Modelling Procedure

The first stage experiments provide basic information about the process in order to plan continued experiments. The process has to be systematically investigated. Figure 5.11 shows the iterative identification procedure which has a natural flow:

1. Collect data
2. Choose a model set
3. Pick the best model in the set. The model may be insufficient because of:
    - The model order was not appropriate.
    - The data set was not informative enough.

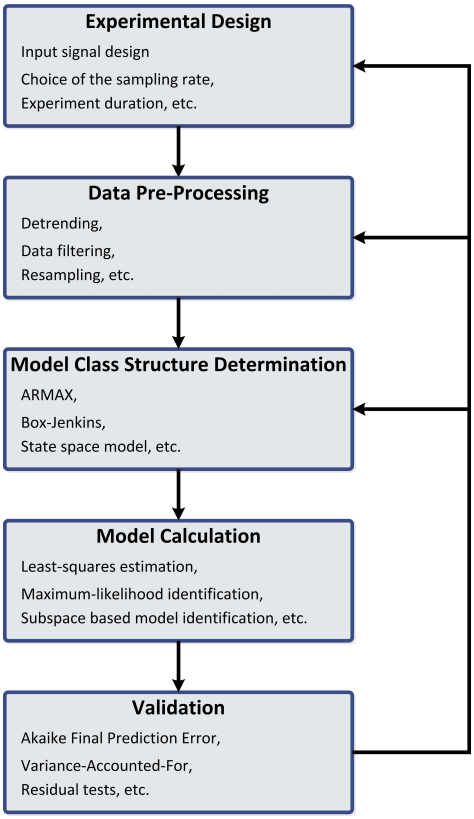



Figure 5.11: Iterative identification procedure.

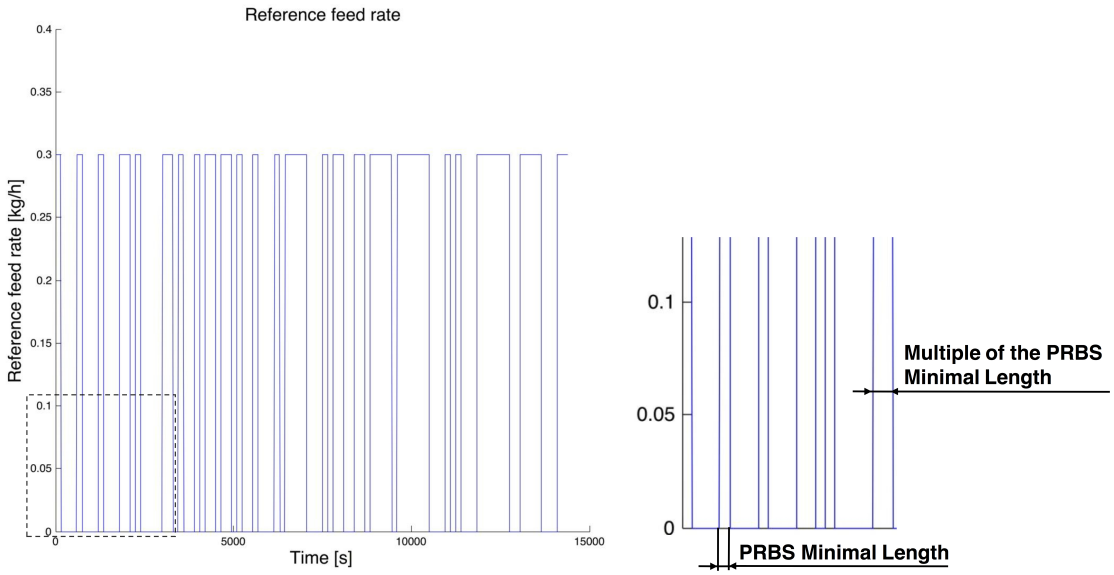
- The validation criterion was not well chosen.
- The numerical procedure failed to find the best model.

Addressing these problems leads to an iterative procedure. Careful experiment design, yielding data with good information is the basis of a successful identification application. The input sequence has an substantial influence on the observed data. The freedom of the input choice is limited by the process [31, 36].

### Training Set

In this section the requirements on the input sequence or training set are presented. In addition, an appropriate training set is chosen and analysed.

**Training Set Requirements:** The training set should cover the necessary ranges of the phenomena and interactions involved. Variations in the measurement conditions should be covered, in order to identify a robust model. Furthermore, the choice of the input for model-based methods relies on criteria of sufficient excitation for determination of the spectral range of validity, i.e. the input should contain a sufficient number of distinct frequencies.




(a) PRBS as reference feed rate. (b) Emphasise the PRBS minimal length.

Figure 5.12: Introduction to PRBS as excitation signal.

Therefore, the properties of the estimate only depend on the input spectrum and not on the waveform of the input. Furthermore, on the one hand the amplitude should be large enough to ensure a good SNR, but on the other hand the input has a limited amplitude [31, 36].

**Choice of a Training Set:** A *pseudo random binary sequence* (PRBS), as illustrated in Figure 5.12a, fulfills the requirements on the training set for the HME process. The PRBS is specified by its minimal length (Figure 5.12b), which can be chosen by the user with the following limitations:

- the process time delay and
- the process time constants.

**Spectral Analysis of the Training Set:** Figure 5.13 illustrates the spectrum of white noise, a PRBS as input feed rate and the input feed rate signal used in a first stage experiment. White noise would be the best choice to cover a wide frequency range, but it has not a specified signal amplitude. The frequency range of the input is limited by the process time delay. In comparison to the feeder reference signal used in the first stage experiments (various input steps) the PRBS covers a larger frequency range and, therefore, PRBS as input sequence provides more information about the system.

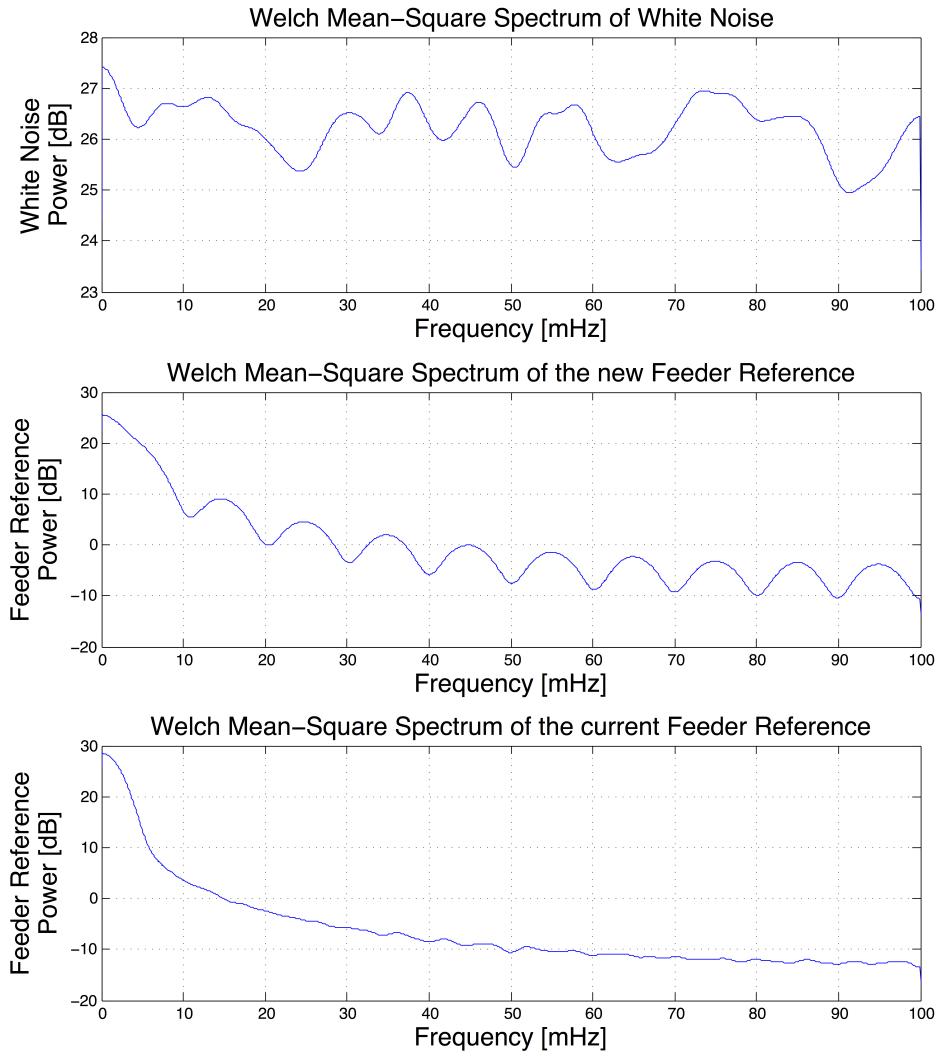



Figure 5.13: Spectral analysis of various different input sequences (white noise, PRBS and step changes).

### Model Validation

The concept of model validation verifies that the identified models qualify for the intended purpose. Usually the major objective is to obtain a model with least possible complexity while fulfilling the required model accuracy. On the one hand, the order of an model has to be validated considering the purpose of the model and on the other hand various types of models have to be compared in respect to suitable validation criteria. As statistical decision criteria the *Akaike Final Prediction Error* (FPE), percentage *Variance-Accounted-For* (VAF), and residual analysis are applied to choose an appropriate model order and to compare different types of models.

**Akaike Final Prediction Error (FPE):** Generally, the prediction error is expected to decrease as the number of parameters increase. This is caused by the fact that the prediction error is calculated for the same data set as used for the identification. An overparameterised model might give poor predictions on a new data set. Thus, a large number of parameters tends to overfitting; this is why a penalty term is introduced by modifying the loss function 5.17. The FPE is defined by the following equation:

$$FPE(p) = \hat{\sigma}_w^2 \left(1 + \frac{p}{N}\right) \ , \tag{5.34}$$

where $p$ is the number of estimated parameters and $N$ is the number of observations (cf. trainings data set $\mathbf{\Phi_{PM}}$ in equation 5.4). The variable $p$ is equal to the length of the unknown parameter vector $\theta$ as denoted in equation 5.13 for the ARMAX model and in equation 5.24 for the Box-Jenkins model. The estimated variance $\hat{\sigma}_w^2$ can be derived by equation 5.18. Regarding the FPE criterion, the model with the minum $FPE$ should be chosen [31, 36].

$$\hat{p} = \arg\min \ FPE(p) \tag{5.35}$$

**Variance-Accounted-For (VAF):** The VAF gives a measure of the correctness of a model. It compares the estimated output $\hat{\mathcal{Y}}$ with the output data $\mathcal{Y}$.

$$\mathcal{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \qquad \hat{\mathcal{Y}} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_N \end{bmatrix} \tag{5.36}$$

The calculation of the VAF is expressed in equation 5.37.

$$\tau_{vaf} = \left(1 - \frac{(\mathcal{Y} - \hat{\mathcal{Y}})^T (\mathcal{Y} - \hat{\mathcal{Y}})}{\mathcal{Y}^T \mathcal{Y}}\right) 100 \tag{5.37}$$

Thus, a perfect match of the estimated output and the observation results in a $\tau_{vaf}$ of 100%, whereas any mismatch will decrease this value [31].

**Residual Analysis:** The part of the data the model could not reproduce are the residuals, as denoted in 5.38 according to equation 5.14.

$$\epsilon_k = y_k - \hat{y}_k = y_k - \phi_{\mathbf{k}}^{\mathbf{T}} \hat{\theta} \tag{5.38}$$

The residual sequence should be structureless and uncorrelated with any other variable including inputs and outputs. Therefore, to extract this information from the residuals, an autocorrelation and cross-correlation test are performed. The autocovariance function of residuals is defined as

$$\hat{C}_{\epsilon\epsilon}(\tau) = \frac{1}{N - \tau} \sum_{k=\tau+1}^{N} \epsilon_k \epsilon_{k-\tau} \ . \tag{5.39}$$

The vector of residuals autocorrelation can be calculated by

$$r_{\epsilon\epsilon} = \frac{1}{\hat{C}_{\epsilon\epsilon}(0)} \begin{bmatrix} \hat{C}_{\epsilon\epsilon}(1) & \dots & \hat{C}_{\epsilon\epsilon}(m) \end{bmatrix}^T , \tag{5.40}$$

where $m$ specifies the time window in samples (lags) of which residual analysis should be investigated. Moreover, testing the independence of the residuals $\epsilon$ and the input $r$ is based on the cross-covariance function

$$\hat{C}_{r\epsilon}(\tau) = \frac{1}{N-\tau} \sum_{k=\tau+1}^{N} \epsilon_k r_{k-\tau} . \tag{5.41}$$

Thus, the cross-correlation vector can be denoted as

$$r_{r\epsilon} = \frac{1}{\hat{C}_{r\epsilon}(0)} \begin{bmatrix} \hat{C}_{r\epsilon}(1) & \dots & \hat{C}_{r\epsilon}(m) \end{bmatrix}^T . \tag{5.42}$$

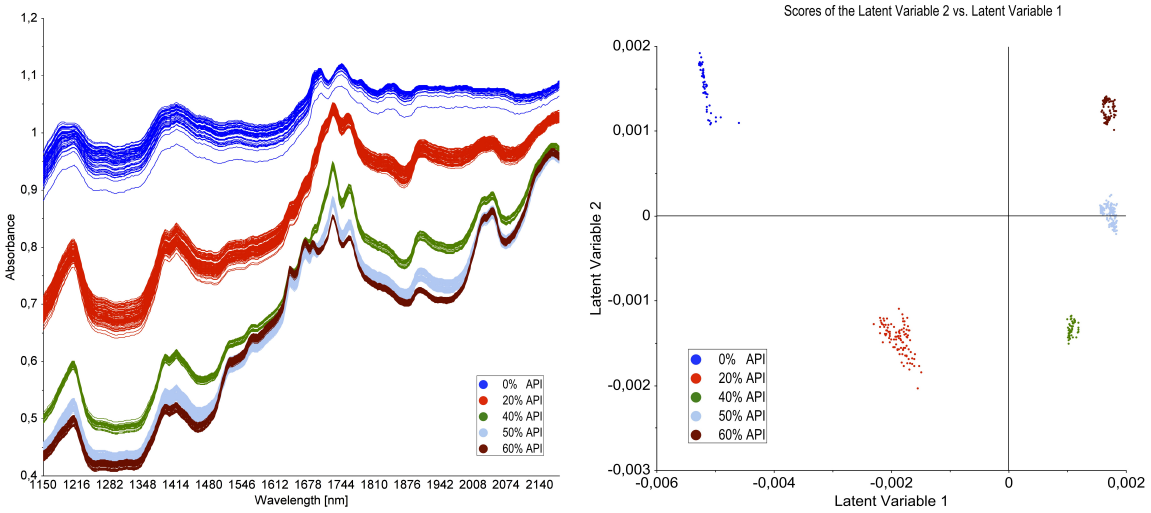
## 5.4 System Identification Results

This section presents the results of the identification of the chemometric and the process model. The process model identification compares three different types of model structures; (1) the ARMAX model, (2) a Box-Jenkins model and (3) a state-space model. The selection of an appropriate model order was carried out by comparing the AIC, the percentage VAF, and the results of residual analysis as discussed in section 5.3.2. The available data for identification and validation is depicted in Figure 5.8. The second part from ~1300 to ~2800 seconds was used as trainings data, whereas the validation data contains the entire sequence.

### 5.4.1 Chemometric Model

Figure 5.14a illustrates the raw spectra of the stationary process of various different amplitudes. Thus, only the spectra are used as trainings data when the process is in steady-state. The spectra are grouped according to the predefined API concentration, as illustrated in Figure 5.14b. This figure shows the score of the latent variable 2 against the latent variable 1. These scores are calculated based on the pretreated data set and in the further procedure three latent variables were used. A Savitzky-Golay filter enabled both smoothing and derivative calculation. Thus, the filter removes the baseline offset between spectra, and differences in baseline slopes between spectra, before the PLS algorithm is started [6, 32].

The predictions of the chemometric model are illustrated in Figure 5.8. The *Root Mean Square Error* (RMSE) using the trainings data as validation data is 1.3611 and the $R^2$ is 0.995.




(a) Raw spectrum.

(b) Scores of the latent variable 2 vs. latent variable 1.

Figure 5.14: Input spectra and result of the chemometric model.

### 5.4.2 ARMAX Model

The ARMAX model has the structure as denoted in equation 5.11. The missing parameters $n_A$, $n_B$ and $n_C$ are chosen as 3, 0, and 2 respectively. The corresponding estimated polynomials are

$$\hat{A}(z^{-1}) = 1 + 0.3649z^{-1} - 0.3471z^{-2} - 0.8502z^{-3} \tag{5.43}$$

$$\hat{B}(z^{-1}) = 28.06 \tag{5.44}$$

$$\hat{C}(z^{-1}) = 1 + 1.337z^{-}1 + 0.8893z^{-2} \ . \tag{5.45}$$

The dynamic model $\hat{G}(z^{-1})$ and the disturbance model $\hat{H}(z^{-1})$ as defined in equation 5.12 can be derived.

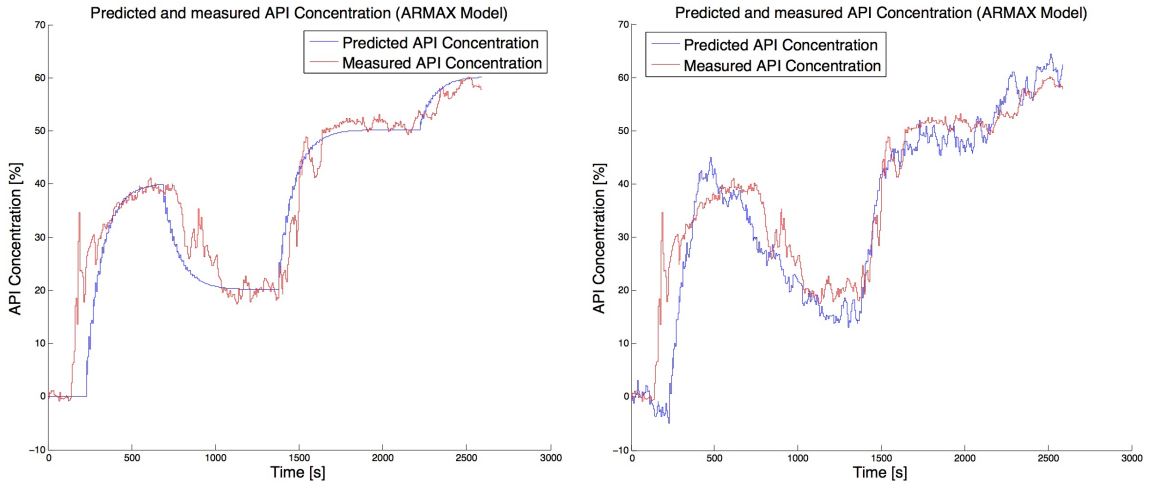
$$\hat{G}(z^{-1}) = \frac{28.06}{1 + 0.3649z^{-1} - 0.3471z^{-2} - 0.8502z^{-3}} \tag{5.46}$$

$$\hat{H}(z^{-1}) = \frac{1 + 1.337z^{-}1 + 0.8893z^{-2}}{1 + 0.3649z^{-1} - 0.3471z^{-2} - 0.8502z^{-3}} \ . \tag{5.47}$$

Including the process time delay $d$ of 46 samples obtained from the cross-correlation analysis in section 5.3.1, the ARMAX model can be written as

$$\hat{y}_k = z^{-46}\hat{G}(z^{-1})r_k + \hat{H}(z^{-1})w_k \ . \tag{5.48}$$

Figure 5.15a illustrates the predicted API concentration using the dynamic model $\hat{G}(z^{-1})$ and the measured API concentration. The simulation as depicted in Figure 5.15b included the dynamic model $\hat{G}(z^{-1})$ and a disturbance model $\hat{H}(z^{-1})$.




(a) Dynamic ARMAX model (b) Dynamic and disturbance ARMAX model.

Figure 5.15: Separation of the dynamic and the disturbance model of the ARMAX model.

The autocorrelation and cross-correlation with a m of 100 and a confidence interval of 99% is illustrated in Figure 5.16. Moreover, the model has a FPE of 1.0193; and the VAF is 88.0875%. Furthermore, the variance of the white noise process can be calculated in accordance to equation 5.18 resulting in a $\hat{\sigma}_w^2$ of 0.9802.

### 5.4.3 Box-Jenkins Model

The Box-Jenkins model structure is denoted in equation 5.22. Hence, the missing parameters $n_B$, $n_C$, $n_D$ and $n_F$ are chosen as 0, 3, 2, and 5 respectively. The estimated polynomials are

$$\hat{B}(z^{-1}) = 2.375 \tag{5.49}$$

$$\hat{C}(z^{-1}) = 1 - 0.5926z^{-1} - 0.07707z^{-2} - 0.1272z^{-3} \tag{5.50}$$

$$\hat{D}(z^{-1}) = 1 - 1.696z^{-1} + 0.7354z^{-2} \tag{5.51}$$

$$\hat{F}(z^{-1}) = 1 - 1.286z^{-1} + 0.3344z^{-2} + 0.4084z^{-3} - 1.273z^{-4} + 0.8304z^{-5}. \tag{5.52}$$

The dynamic model $\hat{G}(z^{-1})$ and the disturbance model $\hat{H}(z^{-1})$ are

$$\hat{G}(z^{-1}) = \frac{2.375}{1 - 1.286z^{-1} + 0.3344z^{-2} + 0.4084z^{-3} - 1.273z^{-4} + 0.8304z^{-5}} \tag{5.53}$$

$$\hat{H}(z^{-1}) = \frac{1 - 0.5926z^{-1} - 0.07707z^{-2} - 0.1272z^{-3}}{1 - 1.696z^{-1} + 0.7354z^{-2}} \ . \tag{5.54}$$

The Box-Jenkins model can be written as

$$\hat{y}_k = z^{-46}\hat{G}(z^{-1})r_k + \hat{H}(z^{-1})w_k \ . \tag{5.55}$$

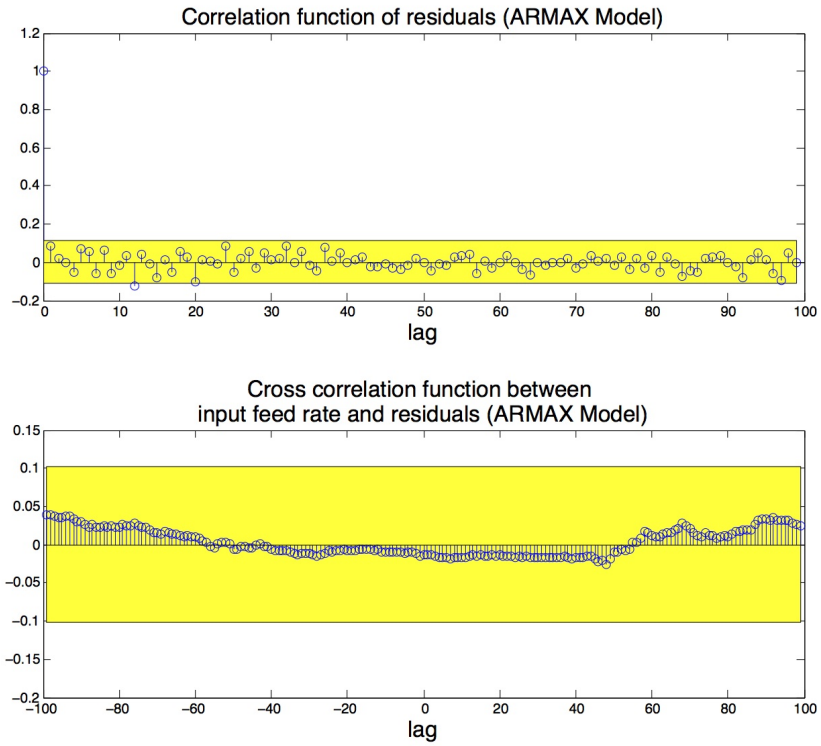
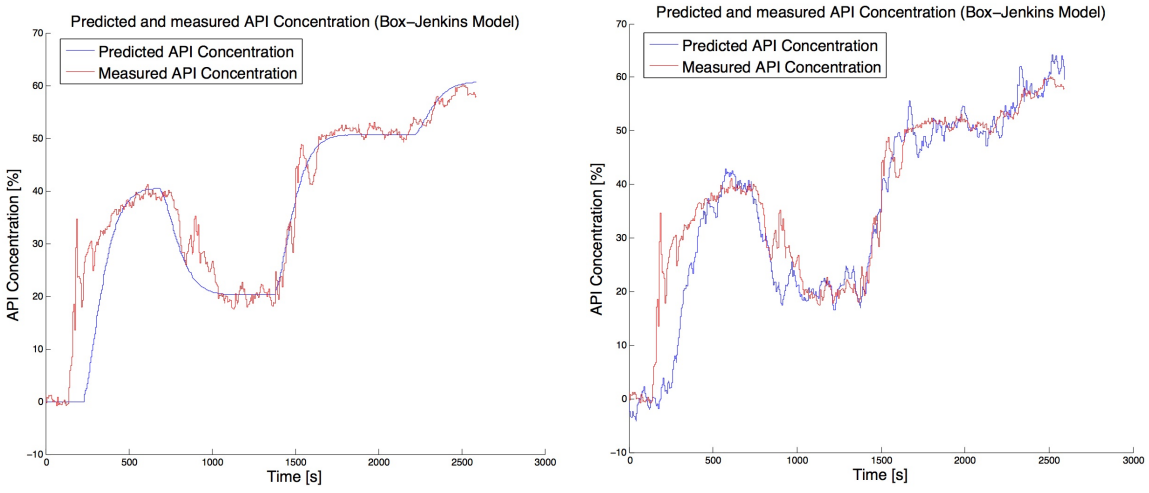



Figure 5.16: Residual analysis of the ARMAX model.




(a) Dynamic Box-Jenkins model.

(b) Dynamic and disturbance Box-Jenkins model.

Figure 5.17: Separation of the dynamic and the disturbance model of the Box-Jenkins model.

Figure 5.17a illustrates the predicted API concentration using the dynamic model $\hat{G}(z^{-1})$ and the measured API concentration. The simulation as depicted in Figure 5.17b included the dynamic model $\hat{G}(z^{-1})$ and a disturbance model $\hat{H}(z^{-1})$.

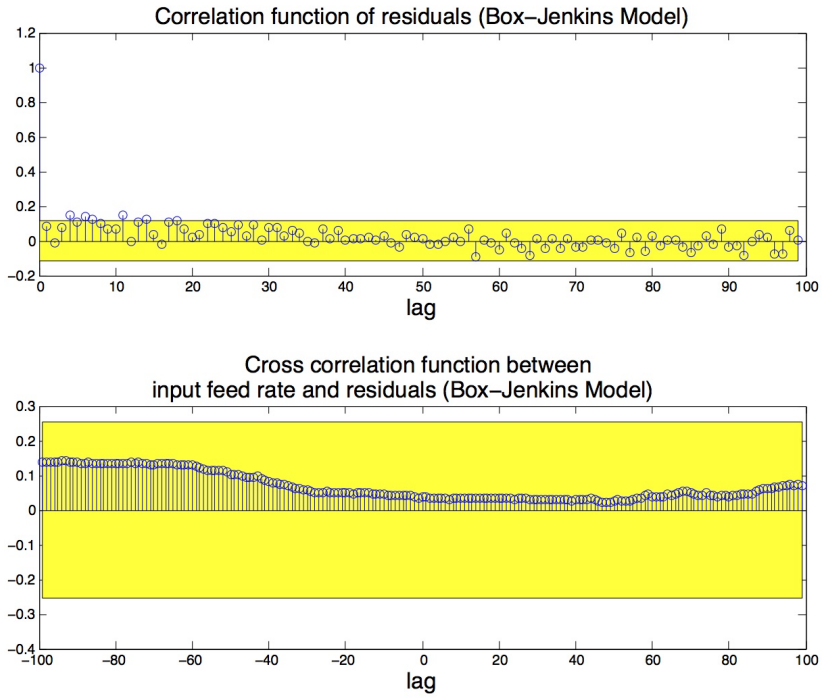



Figure 5.18: Residual analysis of the Box-Jenkins model.

Based on the same confidence interval of 99% and a window size of 100 as used for the residual analysis of the ARMAX model, the autocorrelation and cross-correlation are illustrated in Figure 5.18. Moreover, the model has a FPE of 0.7411, a VAF of 86.8788%, and a $\hat{\sigma}_w^2$ of 0.6906.

### 5.4.4 State-space Model

The identification of a state-space model using the subspace-based identification approach estimates the quadruple $\begin{bmatrix} \hat{\mathbf{\Phi}} & \hat{\mathbf{\Gamma}} & \hat{\mathbf{C}} & \hat{D} \end{bmatrix}$. The resulted state-space matrices are

$$\hat{\mathbf{\Phi}} = \begin{bmatrix} 0.96251 & -0.11263 & 0.095605 & 0.024555 & 0 & \cdots & 0 & 0.011251 \\ 0.078841 & 0.072243 & 0.49256 & 0.44502 & 0 & \cdots & 0 & 0.09877 \\ -0.0078429 & -0.79798 & -0.4446 & -0.11012 & 0 & \cdots & 0 & 0.09877 \\ -0.026657 & 0.29279 & -0.25105 & 0.48553 & 0 & \cdots & 0 & 0.09877 \\ 0 & 0 & 0 & 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & & \ddots & & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 0 \end{bmatrix}$$

$$\hat{\mathbf{\Gamma}} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \qquad \hat{\mathbf{C}}^T = \begin{bmatrix} 106.97 \\ -2.7408 \\ 1.3102 \\ -0.45587 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \qquad \hat{D} = 0 \ , \tag{5.56}$$

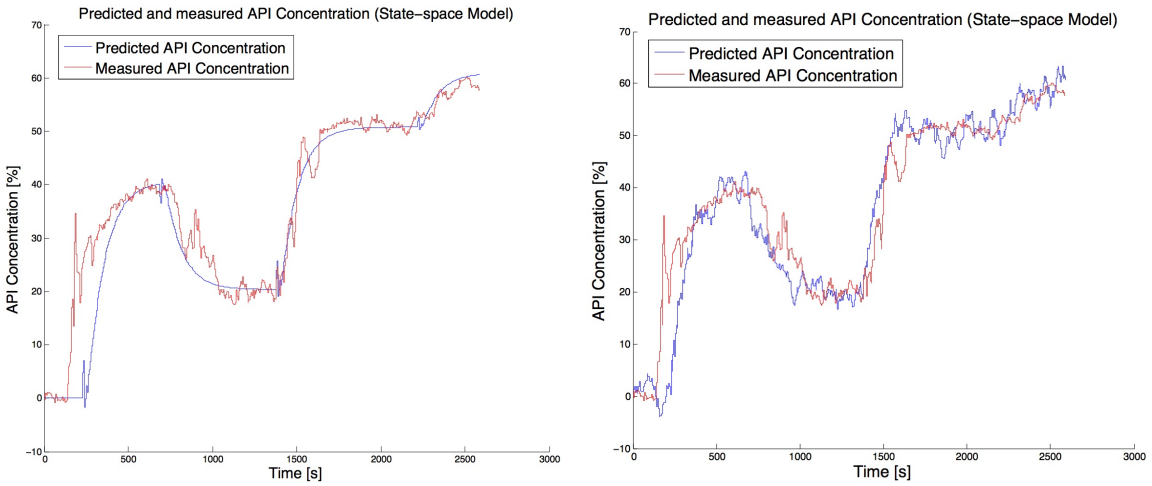
where the dynamic matrix $\hat{\mathbf{\Phi}}$ is a 49x49 square matrix and the vectors $\hat{\mathbf{\Gamma}}$ and $\hat{\mathbf{C}}^T$ both have 49 elements. The models are composed of the first 4 states which are responsible for the dynamics of the system and the rest of the states are required to ensure a total process time delay of 46. Figure 5.19a illustrates the simulation using the state-space matrices as denoted in 5.56. Additionally, the subspace-based identification algorithm computes the vector $\hat{\mathbf{N}}$, as denoted in equation 5.57, for the disturbance model.

$$\hat{\mathbf{N}} = \begin{bmatrix} 0.0094942 \\ 0.0024195 \\ 0.0051432 \\ 0.012639 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{5.57}$$

Figure 5.19b presents the result of the simulation including the disturbance model. The autocorrelation of the residuals and the cross-correlation of the residuals with the input sequence are depicted in Figure 5.20. Moreover, the FPE is 1.34146, the VAF is 86.6731% and $\hat{\sigma}_w^2$ is 1.1924.

### 5.4.5 Model based Simulations

This section presents simulation results using the previous identified models. Considering a constant throughput of 0.6 kg/h, a PRBS with a minimal length of 300 samples and three




(a) Dynamic state-space model.

(b) Dynamic and disturbance state-space model.

Figure 5.19: Seperation of the dynamic and the disturbance model of the state-space model.

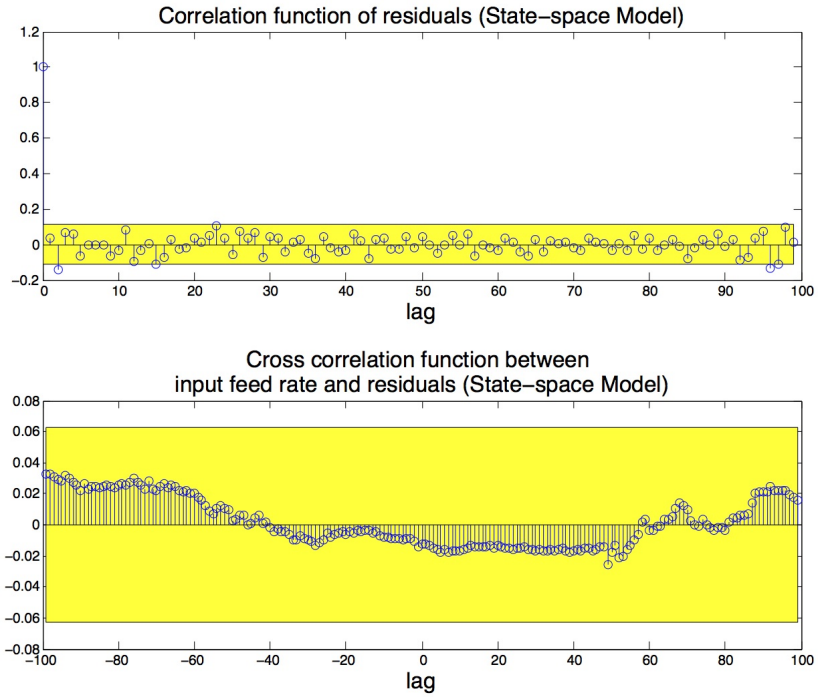
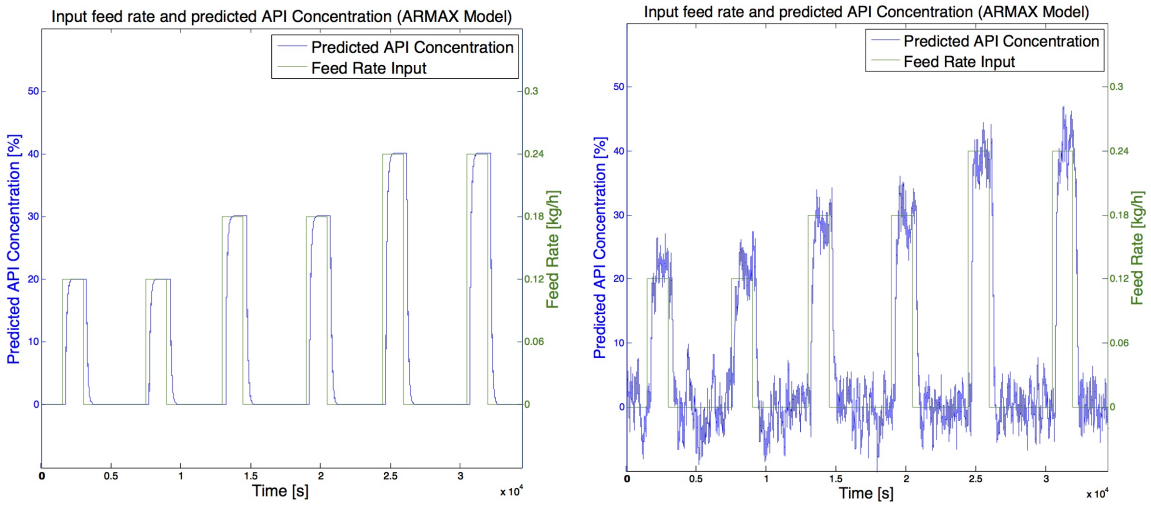


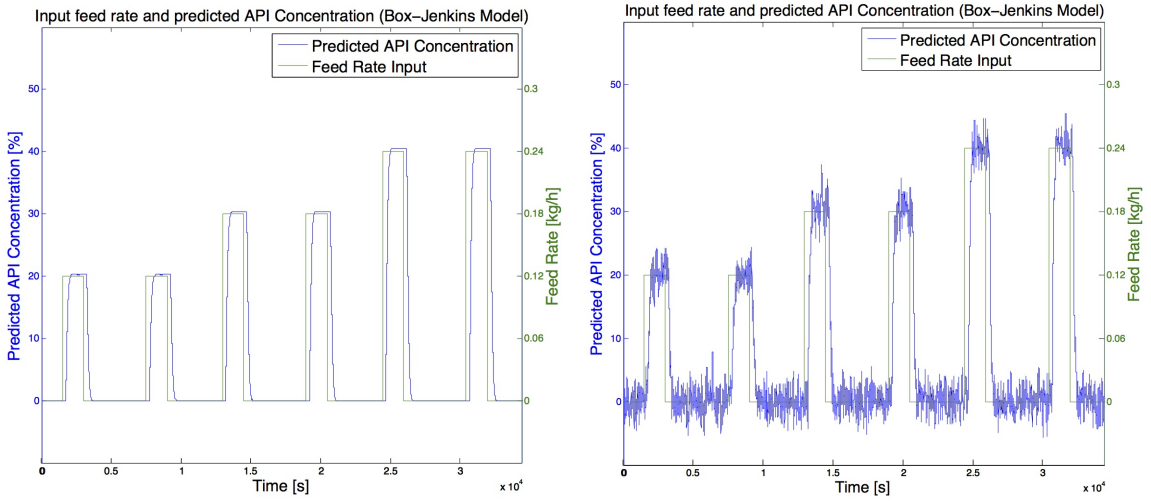

Figure 5.20: Residual analysis of the state-space model.

different amplitudes (0.12, 0.24 and 0.3 kg/h) is simulated. Figures 5.21, 5.22 and 5.23 illustrate the simulation results using the ARMAX, Box-Jenkins and state-space model. The image on the left hand side emphasises the process dynamics, whereas the predicted API concentration as illustrated in the right figure should represent the real process output including measurement noise and process disturbances.




(a) Dynamic model. (b) Dynamic and disturbance model.

Figure 5.21: Simulation of a certain input sequence with the ARMAX model.

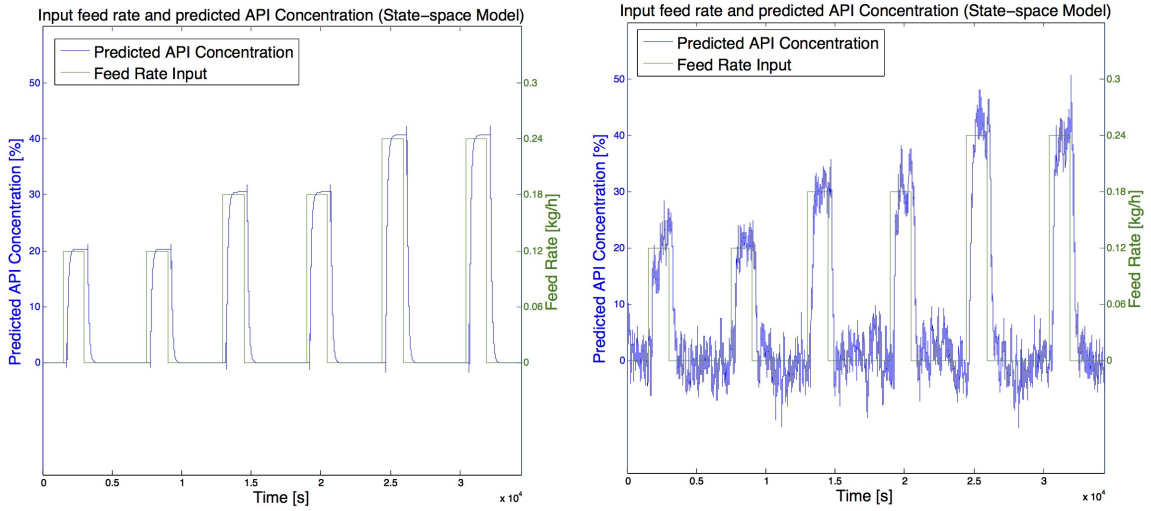



(a) Dynamic model. (b) Dynamic and disturbance model.

Figure 5.22: Simulation of a certain input sequence with the Box-Jenkins model.

### 5.4.6 Conclusion

The chemometric model was estimated by applying the PLS-R algorithm and selecting three latent variables. The data was pretreated by a Savitzky-Golay filter which turned out to provide better predictions compared to preprocessing the data with the *Standard Normal Variate* (SNV) method or using the raw spectra. However, for new data sets from other experiment configurations the SNV method might be a better way to preprocess the




(a) Dynamic model.

(b) Dynamic and disturbance model.

Figure 5.23: Simulation of a certain input sequence with the state-space model.

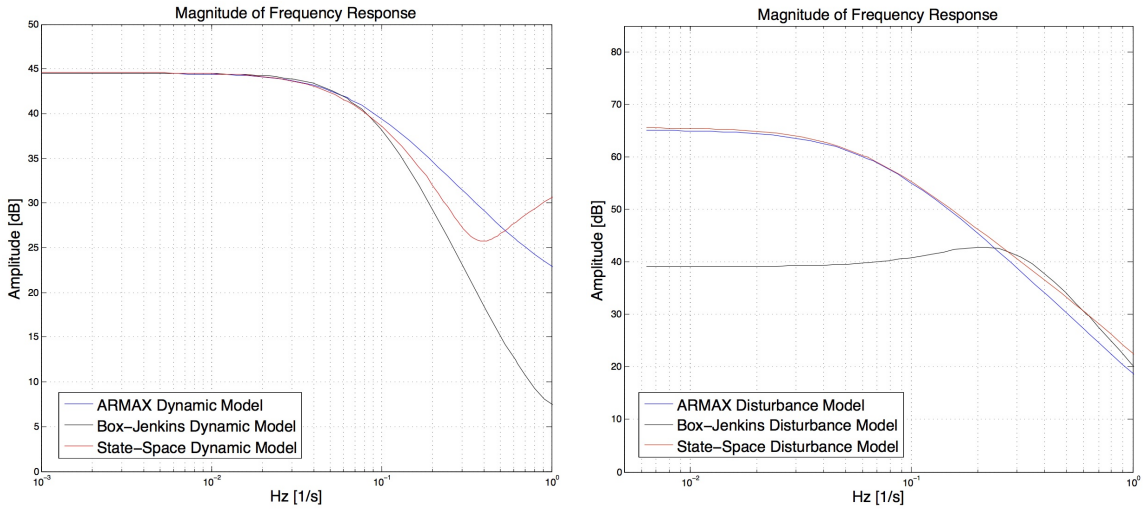
data, because multiplicative variations as physical properties (particle size, shape) cannot be removed by derivatives [6]. The accuracy of the chemometric model highly correlates with the process configuration, as discussed in section 5.3.1. Especially, the steady-state variations could be minimised by optimising the process (cf. Figure 2.10).

Three different approaches were used to estimate the process model. The ARMAX and the Box-Jenkins models are discrete-time transfer functions, whereas the subspace-based algorithm delivered a state-space model. The model structures of ARMAX and Box-Jenkins models are very similar. Especially the dynamic model provides almost the same predictions. This is also highlighted in Figure 5.24a, which compares the dynamic models in frequency domain. Up to a frequency of 0.005 the dynamic models of the ARMAX and Box-Jenkins structure are equal. Moreover, the dominating pole ($z_{p_1} = 0.94$) for the lower frequencies are almost the same, whereas the fast poles differ between the models. The slow pole $z_{p_1}$ corresponds to a frequency of 0.0103 Hz where higher frequencies have a lower magnitude as the steady-state gain of 44.57dB. Since the Box-Jenkins model has complex conjugated poles $z_{p_1,2} = 0.9383 \pm 0.0273i$, its magnitude response decreases faster than in the ARMAX model with a single pole [37]. The steady state gain of 44.57dB can be converted into a magnitude of 169.2388 by the expression denoted in equation 5.58.

$$\begin{aligned} y_{dB} &= 20 \log_{10}(y) \\ y &= 10^{\frac{y_{dB}}{20}} \end{aligned} \tag{5.58}$$

The steady-state magnitude can easily be verified by calculating the required gain to convert the maximal input of 0.6kg/h into the maximal API concentration of 100%. This results in a gain of 166.6 and, therefore, the steady-state gain of the model is a good approximation.

Furthermore, Figure 5.24b depicts the magnitude of the frequency response of the disturbance




(a) Dynamic models.

(b) Disturbance models.

Figure 5.24: Comparison of the frequency response of the estimated models.

models. Here, the variations between the models in the low frequency range are larger than for the dynamic models. According to the predictions used for validation (cf. Figures 5.15b, 5.17b and 5.19b) and especially to the simulations (c.f. Figures 5.21b, 5.22b and 5.23b), the Box-Jenkins model estimates the acting process disturbances and measurement noise on the HME process best. Due to the structure of the Box-Jenkins model (disturbance model can be completely separated from the dynamic model), it has more freedom to estimate the disturbances acting on the process than the other kind of models. Comparing the disturbance and dynamic model of the ARMAX approach, both have the same denominator polynomial $A(z^{-1})$ and therefore the poles are the same (except for pole-zero cancellations). Consequently, the pole $z_{p_1}$ (frequency of 0.0103 Hz) from the dynamic model (Figure 5.24a) also shows up in the disturbance model (Figure 5.24b).

Moreover, Table 5.1 contains the results of both validation criterions, VAF and FPE, and also includes the variance of the white noise process. This variance $\hat{\sigma}_w^2$ is directly proportional to the loss function (cf. equation 5.18). The Box-Jenkins model has the lowest variance which also bears the above stated argument that this type of model is the best concerning the disturbance model.

| Model Type | VAF [%] | FPE [-] | $\hat{\sigma}_w^2$ [-] |
|---|---|---|---|
| ARMAX model | 88.0875 | 1.0193 | 0.9802 |
| Box-Jenkins model | 86.8788 | 0.7411 | 0.6906 |
| State-space model | 86.6731 | 1.3415 | 1.1924 |

Table 5.1: Comparison of identified models

Examining the result of the VAF of all three models, the ARMAX model has the best fit according to this validation criterion. Since the Box-Jenkins model has the lowest FPE, it is a good trade-off between the model complexity and the prediction error. To sum up, the Box-Jenkins model is used for simulations, whereas the state-space model is the basis of the M-step ahead predictor, as described in the next section.

## 5.5 Real-Time Prediction

This section represents an application of the identified model which can be directly incorporated into SIPAT. Prediction of a quality parameter in real-time supports the operator by allowing process and product characterisation and validation at run-time. Predicting a quality parameter including the dynamics of the process presupposes an accurate process model in state-space realisation. Therefore, the state-space model 5.56 is the basis of the predictor.

### 5.5.1 Observer and Kalman Filter

Typically, the states $\mathbf{x}$ of an system cannot be observed and thus an observer reconstructing the states from the available measurement of the API concentration $y_c$ and the input feed rate $r_f$ is required. In the following description the observed API concentration is denoted as $y$ and the input feed rate as $r$.

The system described by the equations 5.59 and 5.60 can be used to simulate the system with a known input $r_k$ and an initial state vector $\mathbf{x_0}$.

$$\hat{\mathbf{x}}_{\mathbf{k+1}} = \mathbf{\Phi}\hat{\mathbf{x}}_{\mathbf{k}} + \mathbf{\Gamma} r_k \tag{5.59}$$

$$y_k = \mathbf{C}\hat{\mathbf{x}}_{\mathbf{k}} + D r_k \tag{5.60}$$

Rewriting equation 5.60 results in a quality expression

$$0 = y_k - \mathbf{C}\hat{\mathbf{x}}_{\mathbf{k}} - D r_k \tag{5.61}$$

of the simulation. If the simulated state vector $\hat{\mathbf{x}_{\mathbf{k}}}$ is exactly the true state vector $\mathbf{x_k}$, there would be no measurement noise. The quality expression as derived in equation 5.61 can be fed back in the following form:

$$\hat{\mathbf{x}}_{\mathbf{k+1}} = \mathbf{\Phi}\hat{\mathbf{x}}_{\mathbf{k}} + \mathbf{\Gamma} r_k + \mathbf{K}\left(y_k - \mathbf{C}\hat{\mathbf{x}}_{\mathbf{k}} - D r_k\right) \ . \tag{5.62}$$

Here, a vector $\mathbf{K}$ with $n$ elements is introduced where $n$ is the order of the system. This vector should be selected in respect to minimise the error between the observed state and the true state and thus the estimate error

$$\tilde{\mathbf{x}}_{\mathbf{k}} = \mathbf{x_k} - \hat{\mathbf{x}}_{\mathbf{k}} \tag{5.63}$$

is formed. The true state expression from 5.25 with the white noise properties as denoted in equation 5.26 and the estimated state from 5.59 are substituted in equation 5.63 to derive after some manipulations a dynamic description of the estimation error.

$$\begin{aligned}
\tilde{\mathbf{x}}_{\mathbf{k+1}} &= \mathbf{x}_{\mathbf{k+1}} - \hat{\mathbf{x}}_{\mathbf{k+1}} \\
&= \mathbf{\Phi}\mathbf{x}_{\mathbf{k}} + \mathbf{\Gamma} r_k + \mathbf{N} v_k - \left(\mathbf{\Phi}\hat{\mathbf{x}}_{\mathbf{k}} + \mathbf{\Gamma} r_k + \mathbf{K}\left(y_k - \mathbf{C}\hat{\mathbf{x}}_{\mathbf{k}} - D r_k\right)\right) \\
&= \mathbf{\Phi}\left(\mathbf{x}_{\mathbf{k}} - \hat{\mathbf{x}}_{\mathbf{k}}\right) - \mathbf{K}\left(\mathbf{C}\mathbf{x}_k + D r_k + w_k - \mathbf{C}\hat{\mathbf{x}}_{\mathbf{k}} - D r_k\right) + \mathbf{N} v_k \\
&= \mathbf{\Phi}\left(\mathbf{x}_{\mathbf{k}} - \hat{\mathbf{x}}_{\mathbf{k}}\right) - \mathbf{K}\mathbf{C}\left(\mathbf{x}_k - \hat{\mathbf{x}}_{\mathbf{k}}\right) + \mathbf{N} v_k - \mathbf{K} w_k \\
&= \underbrace{\left(\mathbf{\Phi} - \mathbf{K}\mathbf{C}\right)}_{\tilde{\mathbf{\Phi}}} \tilde{\mathbf{x}}_{\mathbf{k}} + \mathbf{N} v_k - \mathbf{K} w_k
\end{aligned} \tag{5.64}$$

$\mathbf{K}$ shows up in two different ways in expression 5.64. The estimation error dynamic matrix $\tilde{\mathbf{\Phi}}$ indicates the reconstruction speed. $\mathbf{K}$ directly influences the forgetting factor of old estimation errors. On the other hand, the vector $\mathbf{K}$ multiplying the measurement noise $w_k$ also influences the estimation error. This term expresses the sensitivity of the system on measurement noise, e.g. a large $\mathbf{K}$ gives a significant influence of the noise $w_k$ on the estimation error. As a consequence, the selection of $\mathbf{K}$ is a trade-off between sensitivity to the measurement disturbances and adaptability to the influence of the system disturbances [30, 31, 36]. Based on the white noise properties 5.26, the vector $\mathbf{K}$ can be calculated by

$$\mathbf{K} = \left(\mathbf{P}\mathbf{C}^{\mathbf{T}} + \mathbf{N} R_{12}\right) R_2^{-1} \tag{5.65}$$

with the matrix $\mathbf{P}$ as a solution from the Riccati equation

$$\mathbf{A}\mathbf{P} + \mathbf{P}\mathbf{A}^{\mathbf{T}} - \left(\mathbf{P}\mathbf{C}^{\mathbf{T}} + \mathbf{N} R_{12}\right) R_2^{-1} \left(\mathbf{P}\mathbf{C}^{\mathbf{T}} + \mathbf{N} R_{12}\right)^T + \mathbf{N} R_1 \mathbf{N}^{\mathbf{T}} = \mathbf{0} \ . \tag{5.66}$$

The proof of the equations 5.65 and 5.66 are shown in [30]. The resulting vector from equation 5.65 is denoted as the Kalman gain and thus the observer 5.67 is also called Kalman filter. For this system the direct term $D$ can be neglected, because there is no direct influence of the input on the output.

$$\hat{\mathbf{x}}_{\mathbf{k+1}} = \tilde{\mathbf{\Phi}}\hat{\mathbf{x}}_{\mathbf{k}} + \mathbf{\Gamma} r_k + \mathbf{K} y_k \tag{5.67}$$

$$\hat{y}_{k+1} = \mathbf{C}\hat{\mathbf{x}}_{\mathbf{k+1}} \tag{5.68}$$

This observer can directly be applied as an one-step ahead predictor, as denoted in equation 5.68. The future API concentration $\hat{y}_{c_{k+1}}$ can be predicted based on the current measurement $y_{c_k}$ and the input feed rate $r_{f_k}$, as illustrated in Figure 5.25a, using the one-step ahead predictor.

### 5.5.2 M-Step Ahead Predictor

Based on the one-step ahead predictor a M-step ahead predictor can be implemented. The kernel of the M-step ahead predictor is the observer 5.67. Using the estimated state vector

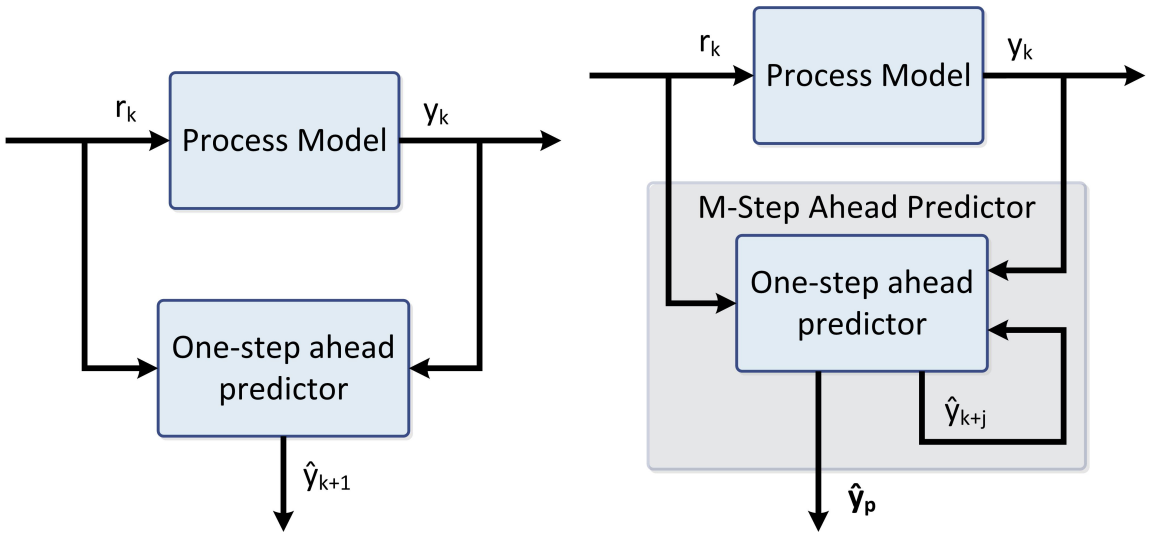
$\hat{\mathbf{x}}_{\mathbf{k+1}}$ and the calculated output $\hat{y}_{k+1}$ and assuming a constant input the new state $\hat{\mathbf{x}}_{\mathbf{k+2}}$ and corresponding output $\hat{y}_{k+2}$ can be calculated.

$$\begin{aligned}
\hat{\mathbf{x}}_{\mathbf{k+2}} &= \tilde{\mathbf{\Phi}}\hat{\mathbf{x}}_{\mathbf{k+1}} + \mathbf{\Gamma} r_k + \mathbf{K}\hat{y}_{k+1} \\
\hat{y}_{k+2} &= \mathbf{C}\hat{\mathbf{x}}_{\mathbf{k+2}} \\
&\vdots \\
\hat{\mathbf{x}}_{\mathbf{k+M}} &= \tilde{\mathbf{\Phi}}\hat{\mathbf{x}}_{\mathbf{k+M-1}} + \mathbf{\Gamma} r_k + \mathbf{K}\hat{y}_{k+M-1} \\
\hat{y}_{k+M} &= \mathbf{C}\hat{\mathbf{x}}_{\mathbf{k+M}}
\end{aligned} \tag{5.69}$$

Finally, the algorithm computes M predicted samples $\hat{\mathbf{y}}_{\mathbf{p}}$ in respect to the latest measurements and the input samples, as illustrated in Figure 5.25b. M is also known as the prediction horizon.

### 5.5.3 Implementation of the M-step Ahead Predictor

The M-step ahead predictor requires the collaboration of the spectrometer, SIPAT, Umetrics, and Matlab as illustrated in Figure 5.26. The spectrometer is the direct interface to the HME process and delivers a spectrum $y_{s_k}$ to SIPAT every 2 to 3 seconds depending on the internal time period of the spectrometer. SIPAT represents the superordinate unit which is responsible for coordinating and scheduling different task and guarantees a uniform sample period (5 seconds). A prerequisite of the M-step ahead predictor, is the observed API concentration and the current input sample. Hence, Umetrics including the chemometric model maps the measured spectrum to the corresponding API concentration, which is forwarded to the Matlab function.




(a) One-step ahead predictor.

(b) M-step ahead predictor.

Figure 5.25: Development of a M-step ahead predictor.

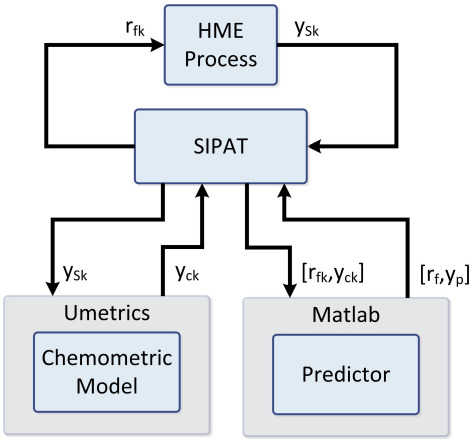



Figure 5.26: Basic data flow between SIPAT, Umetrics and Matlab.

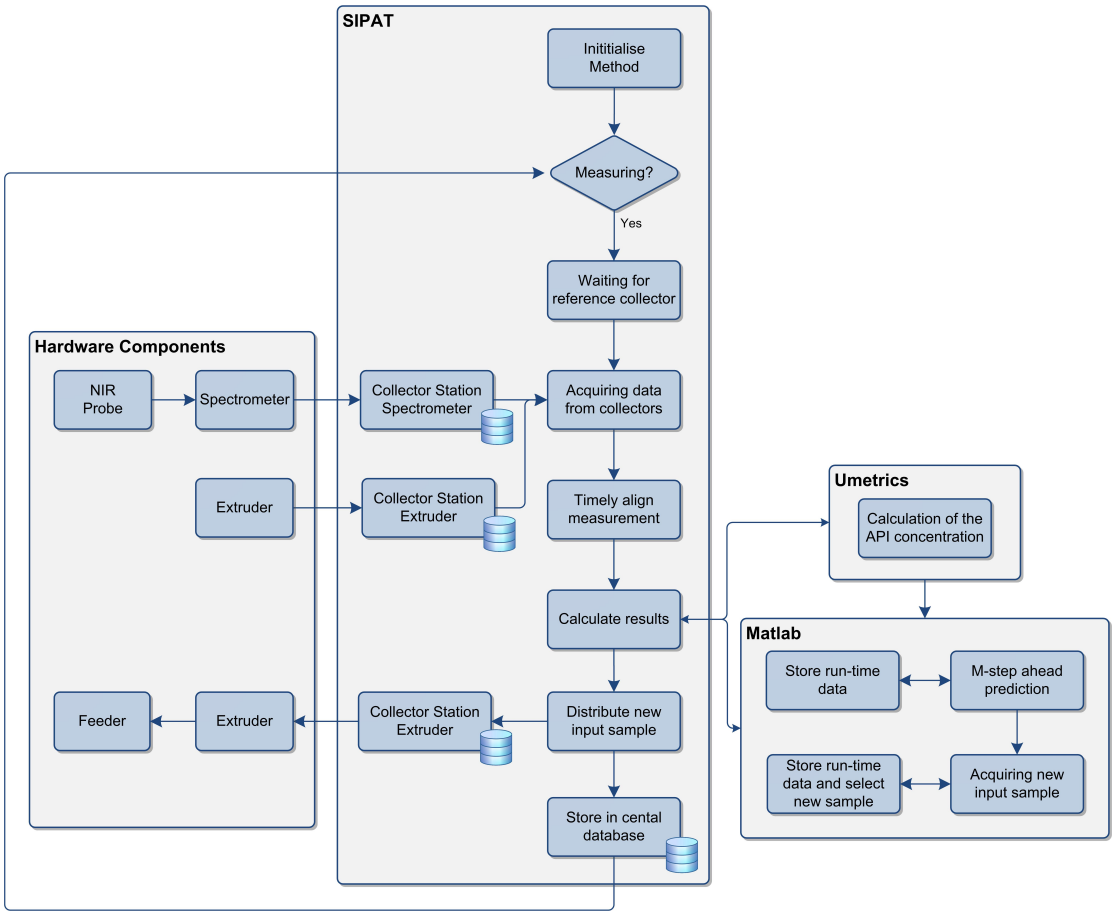



Figure 5.27: More detailed data flow diagram to illustrate the implementation of the M-step ahead predictor.

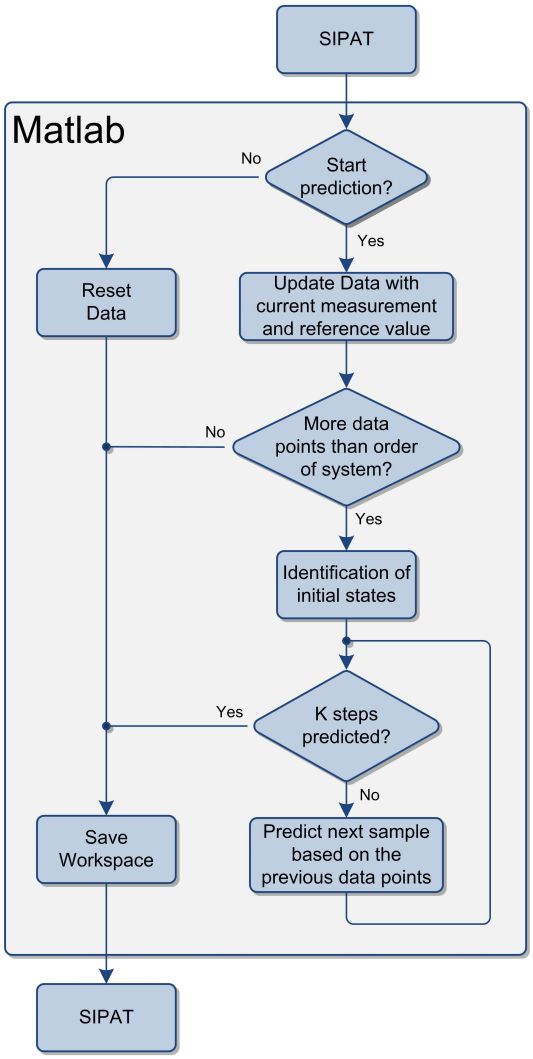



Figure 5.28: Implementation of the M-step ahead predictor in Matlab.

The data flow diagram as depicted in Figure 5.27 presents the entire procedure. According to the network architecture as described in chapter 3, both collector stations (spectrometer and extruder), Umetrics Simca-Q, and Mathworks Matlab as external calculation engines are involved. The main tasks of the included Matlab functions are; (1) calculation of M predicted samples and (2) delivering a new sample for the feed rate. The entire input sequence of the feed rate are predefined in Matlab and stored in a `MAT`-file. After acquiring a new sample of the input feed rate, SIPAT distributes this scalar via OPC technology to the extruder. The extruder forwards the value to the feeder, which finally modifies the feed rate. The Matlab function implementing the M-step ahead predictor, as illustrated in the data flow diagram 5.28, is called from SIPAT with the following arguments:

- current concentration $y_{c_k}$,
- the current reference feed rate $r_{f_k}$,

- number of predicted samples M (can vary in SIPAT during run-time) and
- a start flag (the SIPAT user has to start the prediction algorithm).

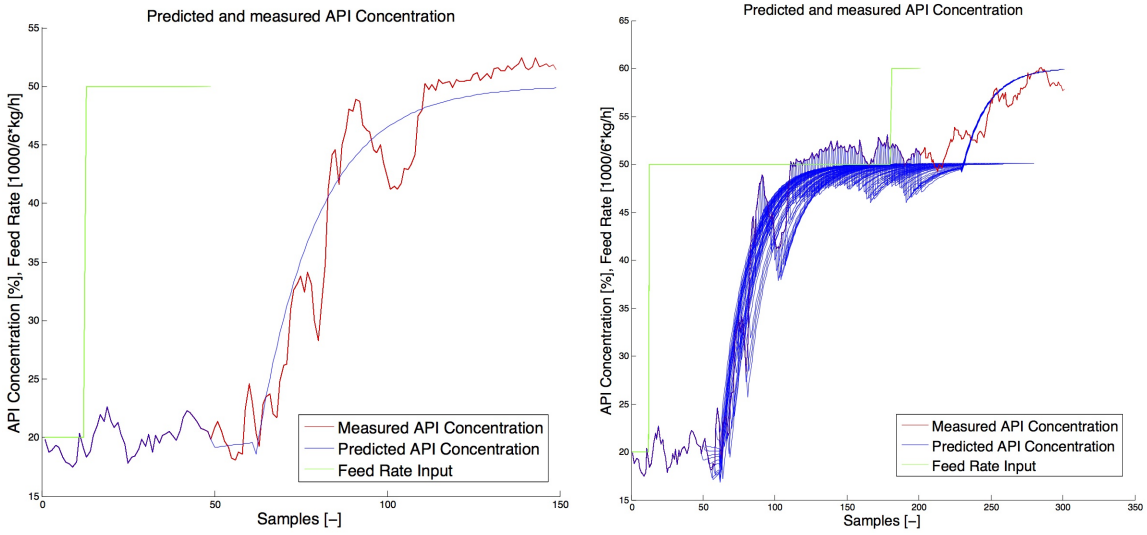
If the start flag is set, this function returns a matrix $\hat{\mathbf{y}}_\mathbf{p}$ (cf. equation 5.70) including all measurements and M predicted API concentration and the input feed rates. Otherwise all observed API concentrations and the input feed rates are returned.

$$\hat{\mathbf{y}}_\mathbf{p} = \begin{bmatrix} y_{c_1} & r_{f_1} \\ y_{c_2} & r_{f_2} \\ \vdots & \vdots \\ y_{c_k} & r_{f_k} \\ \hat{y}_{c_{k+1}} & \hat{r}_{f_{k+1}} \\ \vdots & \vdots \\ \hat{y}_{c_{k+M}} & \hat{r}_{f_{k+M}} \end{bmatrix} \tag{5.70}$$

Basically, to predict the output, the future input sequence should be available. Apparently this cannot be achieved in real-time and therefore, the future input samples are chosen as the latest available input value. The corresponding effect is shown in the next section.

### 5.5.4 Results

A simulation environment was build in Matlab to simulate the M-step ahead predictor. Here, the prediction horizon M was defined as 100 samples. At every timestamp the M future




(a) First 100 samples prediction after the first 49 available measurements.

(b) Result after the entire experiment.

Figure 5.29: Result of the M-step ahead predictor.

samples are calculated as depicted in Figure 5.29a. Due to the order of the system (49), at least 49 data points (reference values and measurements) have to be collected before the predictor can start. Figure 5.29b illustrates all predictions based on the latest measurement. It also emphasises the assumption of a constant input sequence for the prediction horizon (cf. samples 250 to 280).

# 6 Summary and Conclusion

In this work the integration of a twin-screw extruder and the NIR spectrometer in SIPAT are described. The HME process is integrated in SIPAT using OPC technology, whereas the importing of the spectrometer is based on a collector interface/driver. The heart of the SIPAT system is the Server PC including the base station and the database server. Starting a method can be carried out from every client PC and thus the process itself and the client PC can be physically separated (e.g. the client PC is at the office, whereas the extrusion process is located in the laboratory). Furthermore, SIPAT enables real-time monitoring of process parameters from every client PC. After a successful off-line model calibration, the obtained model can be easily imported into SIPAT. Several models can be used at the same time, which opens up the opportunity of validation of different chemometric models in real-time. The API concentration extracted from the observed spectrum represents a critical quality parameter. Therefore, real-time predictions of the API concentration are the basis for optimising the process subject to the intended product quality.

The two use cases demonstrated the capability of SIPAT concerning practical issues such as enabling an automatic DoE and real-time prediction of the concentration. Especially the second use case (chapter 5) allows real-time process analysing and characterisation of product quality attributes. The prerequisite for enabling an automatic DoE (use case in chapter 4) is the manipulation of input parameters out of SIPAT. Consequently, a feedback connection via SIPAT was established which has i.a. the following advantages:

- reduces programming effort (e.g. in order to send a certain input sequence and record corresponding measurements for off-line process analysis).
- increases flexibility (e.g. easy implementation of new controllers).
- enables adaptive process control.

To sum up, SIPAT is very flexible in the sense of importing new models (Umetrics, Camo, Matlab) and it allows the operator to analyse the process in real-time by monitoring critical process parameters. The import of Matlab functions enables the integration of complex calculations in the SIPAT system and therefore, advanced controllers can easily be integrated in the feedback loop of the system.

# 7 Outlook

The future work on the SIPAT system and the HME process will deal with the implementation of new methods in SIPAT. The current state of the system enables easy implementation of more advanced SIPAT methods including chemometric models, matlab functions as well as control actions. The generation of advanced controllers may be based on an estimated model of the HME process. The current SISO model should be extended to a MIMO model representing the HME process. Furthermore, the Kalman filter may be adapted and optimised by a more accurate analysis of the noise (separating the disturbance in measurement noise and process disturbances). However, the model procedure was performed for one appropriate formulation whereas the applied formulations are changed weekly or daily. Whilst the model order may be determined off-line and manually, the computation of the unknown parameters of the model for various formulations could be done automatically or in real-time by implementing adaptive models. This procedure would be based on both Use Cases shown in this work.

It would make sense to use the concept of *model predictive control* (MPC) to control the process. This means, the current control action is obtained by solving on-line an open-loop finite horizon optimal control problem based on current measurements. MPC facilitates the introduction of constraints on the inputs and outputs to prevent unfeasible control actions caused by nonlinearities (e.g. saturation). The success of MPC relies on the process model and thus it is recommended to spend considerable effort on system identification [30, 38]. The development of the model and the controller can be carried out in Matlab, which has the additional advantage of an easy integration of the controller in the system.

Innovative continuous manufacturing processes require monitoring of critical process parameters of each unit operation. Thus, the SIPAT system should be extend ed by subsequent processes of the extruder. Consequently, a hot die face pelletiser for strand cutting, which is directly attached to the die of the extruder at the RCPE should be integrated into the SIPAT system as a new collector station. A method imports the extruder, NIR and pelletiser and provides time-aligned data of both unit operations. An OPC server will be installed on a PC in order to establish a connection to the PLC of the pelletiser. Hence, a collector station has to be configured which acts as an OPC client. The communication between the PLC and SIPAT should be bidirectional in order to enable reading of the output parameters of the pelletiser as well as manipulating input parameters.

# Bibliography


[1] Johannes Khinast. Continuous Manufacturing & the Role of HME in Pharmaceutical Production. http://www.rcpe.at/pdfs/Downloads/ContinuousProcessing.pdf, accessed December 2011.

[2] Jörg Breitenbach. Melt extrusion: from process to drug delivery technology. *European Journal of Pharmaceutics and Biopharmaceutics*, 2002.

[3] Klemens Kohlgrüber, editor. *Co-Rotating Twin-Screw Extruders: Fundamentals, Technology, and Applications*. Hanser Gardner Publications, Inc., 2008.

[4] Leila Abboud and Scott Hensley. Factory shift: New prescription for drug makers: Update the plants. *The Wall Street Journal*, 2003.

[5] Genetic Engineering & Biotechnology News. FDA Backs PAT to Help Solve Manufacturing Woes. http://www.genengnews.com/articles/chitem.aspx?aid=2596&chid=3, accessed July 2011, 2008.

[6] Katherine A. Bakeev, editor. *Process Analytical Technology*. John Wiley & Sons, Ltd, 2nd ed. edition, 2010.

[7] Siemens AG. *SIPAT the software heart of PAT - White Paper*.

[8] Filip De Frenne. *User Manual, SIMATIC SIPAT Version 3.1.1*. Siemens AG.

[9] Marica Williams, Yiwei Tian, David S Jones, and Gavin P Andrews. Hot-melt extrusion technology: Optimizing drug delivery. *European Industrial Pharmacy*, 2010.

[10] Pradip S Partel, Jignesh P Raval, and Hemul V Patel. Review on the pharmaceutical applications of hot melt extruder. *Asian Journal of Pharmaceutical and Clinical Research*, 2010.

[11] Stefan Radl, Thomas Tritthart, and Johannes Khinast. A novel design for hot/melt extrusion pelletizers. *Chemical Engineering Science*, 2009.

[12] Jeffrey Hirsch. Online Monitoring Of Continuous Hot Melt Extrusion. *Pharmaceutical Technology Europe*, 2010.

[13] Ellen Verhoeven. *Hot-Melt Extrusion as Processing Technique for Multiparticulate Dosage Forms Containing Lipophilic and Hydroclic Polymers*. PhD thesis, Ghent University, 2008.

[14] Coperion GmbH. *Kundendokumentation ZSK 18 MC/SIPA*.

[15] Andreas Guldner. *Visualisierung Lastenheft*. Coperion.

[16] Gabriele Reich. Near-infrared spectroscopy and imaging: Basic principles and pharmaceutical applications. *Advanced Drug Delivery Reviews*, 2005.

[17] Timothy A. Haley and Steven J. Mulvaney. On-line system identification and control design of an extrusion coking process: Part i. System Indentification. *Food Control*, 2000.

[18] Liuping Wand, Stephen Smith, and Charlie Chessari. Continuous-time model predictive control of food extruder. *Control Engineering Practice*, 2008.

[19] HMS Industrial Networks GmbH. Profinet. http://www.feldbusse.de/Profinet/profinet.shtml, accessed July.

[20] Philippe De Tandt. *Exception Handling; SIMATIC SIPAT Version 3.1.1.* Siemens AG.

[21] Gerhard Schnell and Bernhard Wiedemann. *Bussysteme in der Automatisierungs- und Prozesstechnik.* Vieweg Verlag, 6. auflage edition, 2006.

[22] Günter Baumann and Matthias Damm. Industrielle Kommunikation: System übergreifend und flexibel. *PRAXIS Profiline - OPC*, 2003.

[23] HMS Industrial Networks GmbH. Industrial Ethernet. http://www.feldbusse.de/CClink-IE/cclinkIE.shtml, accessed July 2011.

[24] HMS Industrial Networks GmbH. Profibus. http://www.feldbusse.de/Profibus/profibus.shtml, accessed July.

[25] Marshall T. Rose and Dwight E.Cass. ISO Transport Service on top of the TCP. *Northrop Research and Technology Center*, 1987.

[26] Siemens AG. *WinCC: Communication Manual*, 1999.

[27] Umetrics AB. *Design of Experiments: Principles and Applications.* Umetrics Academy, 2008.

[28] Kim H. Esbensen. *Multivariate Data Analysis - In Practice: An Introduction to Multivariate Data Analysis and Experimental Design.* CAMO Process AS, 2004.

[29] International Conference on Harmonisation of Technical Requirements for Registration of pharmaceuticals for Human Use. ICH Harmonised Tripartite Guidline: Pharmaceutical Development Q8(R2), 2009.

[30] Torkel Glad and Lennart Ljung. *Control Theory: Multivariable and Nonlinear Methods.* Taylor & Francis, 2000.

[31] Rolf Johansson. *System Modeling and Identification.* Prentice Hall, 2011.

[32] Rudolf W. Kessler. *Prozessanalytik.* Wiley-VCH Verlag GmbH & Co. KGaA, 2006.

[33] David J. am Ende, editor. *Chemical Engineering in the Pharmaceutical Industry: R&D to Manufacturing.* John Wiley & Sons, inc., 2011.

[34] Eva Roblegg, Evelyn Jäger, Aden Hodzic, Gerold Koscher, Stefan Mohr, Andreas Zimmer, and Johannes Khinast. Development of sustained-release lipophilic calcium stearate pellets via hot melt extrusion. *European Journal of Pharmaceutics and Biopharmaceutics*, 2011.

[35] B. Haverkamp and M. Verhaegen. *SMI Toolbox: State Space Model Identification Software for Multivariable Dynamical Systems*. Delft University of Technology, 1997.

[36] Lennart Ljung. *System Identification: Theory for the User*. Prentice Hall PTR, 2009.

[37] Alan V. Oppenheim and Ronald W. Schafer. *Discrete-Time Signal Processing*. Prentice Hall, 3 edition, 2010.

[38] Rolf Johansson. *Predictive and Adaptive Control*. Lund University, Dept. Automatic Control, 2010.