Evaporative batch crystallisation model validation and linearisation for Model Predictive Control

Peter Tijl

DCT 2005.35

Traineeship report

Coach: dr. ir. S.K. Bermingham
Supervisor: prof. dr. ir. A.C.P.M. Backx
Professor: prof. dr. ir. M. Steinbuch

Eindhoven Technical University
Department Mechanical Engineering
Dynamics and Control Technology Group

London, July 2004
Summary

Crystallisation is a separation and purification process, used in the production of a wide range of materials. It involves the formation of one or more solid phases from a liquid phase or amorphous solid phase. Although it is one of the older unit operations, little is understood on how process conditions influence product quality. Therefore, a generic first principle crystallisation modelling framework has been developed [1] in the general PROcess Modelling System (gPROMS) software. The first part of this project is to apply this framework in order to describe a specific evaporative batch crystallisation process as performed by Kemira. Next, the unknown kinetic and initial crystal size distribution (CSD) parameters need to be determined in order to validate the adjusted model. A large number of exploratory simulations is carried out to investigate the sensitivity of the measured variables with respect to these key parameters. Measurement data obtained from experiments carried out by Kemira is used to perform parameter estimation with a gPROMS built-in routine that combines formal mathematically based optimisation techniques with the maximum likelihood formulation for the objective function. One of the problems encountered in parameter estimation was caused by very high values of the objective function gradient. Many possible reasons were investigated and the solution was found in tightening the absolute and relative tolerances of the parameter estimation solver. Now the model validation is performed, the next step is to obtain a stable linear model that can be used for Model Predictive Control (MPC) with the INCA software. Therefore, the software architecture that connects the various software applications (gPROMS, INCA and Matlab) is set up, using a data server as a central node for communication. The Kemira model is linearised using the gPROMS built-in LINEARISE function, resulting in an instable linear model. First it is believed that the non-linear model is instable itself and proportional feedback is applied to stabilise the non-linear Kemira model, without success. A simple non-linear evaporative batch crystallisation model is developed and all corresponding linear models obtained using the LINEARISE function are marginally stable. Linearisation of continuous non-linear gPROMS models always seems to work fine and results in stable linear models. This gives rise to the thought that linearisation of batch processes resulting in stable linear models is not feasible with the current linear system theory. The second approach to obtain a stable linear model is to apply system identification, which proved to be successful for a continuous crystallisation process. System identification is carried out for both the Kemira and the simple model resulting in stable linear models, with poor predictive quality however. Unfortunately, the obtained stable linear models are not used in combination with the INCA software due to a lack of time. A possible reason for the difference in predictive quality of the linear batch process models and the continuous process model all obtained with system identification is the limited freedom of the input signal. An evaporative batch crystallisation process is not in steady-state except for the trivial case of zero heat input. In the continuous case, positive and negative deviations of an input can be applied around a steady state value resulting in a linear model with substantially better predictive quality.
# Table of contents

**Table of contents**  

1  **Introduction**  

2  **Project formulation**  

  2.1  **Assignment**  

  2.2  **Objectives**  

3  **Crystallisation**  

  3.1  **Crystallisation mechanisms**  

    3.1.1  Nucleation  

    3.1.1.1  Primary nucleation  

    3.1.1.2  Secondary nucleation  

    3.1.2  Growth  

    3.1.2  Dissolution  

    3.1.3  Attrition and breakage  

    3.1.4  Agglomeration  

  3.2  **Crystallisation methods**  

    3.2.1  Cooling crystallisation  

    3.2.2  Flash-cooling crystallisation  

    3.2.3  Evaporative crystallisation  

3.3  **Crystalliser types**  

    3.3.1  Draft Tube Baffle (DTB)  

    3.3.2  Forced-circulation (FC)  

    3.3.3  Fluidized bed  

3.4  **Operation modes**  

    3.4.1  Batch  

    3.4.2  Semi-batch  

    3.4.3  Continuous  

3.5  **Crystal Size Distribution (CSD)**  

    3.5.1  Number density distribution  

    3.5.2  Cumulative number density distribution  

    3.5.3  Volume density distribution  

    3.5.4  Cumulative volume density distribution  

    3.5.5  Normalised cumulative volume density distribution  

    3.5.6  Quantiles of the distribution  

    3.5.7  Two log normal distribution  

4  **Process description**  

  4.1  **Crystalliser type and process conditions**  

  4.2  **Mother liquor components**  

  4.3  **Crystallisation method and operation mode**  

  4.4  **Available measurement data**
1 Introduction

This report describes the work I have done during my internal traineeship at Process Systems Enterprise Ltd. in London. The structure of this report and the various topics it discusses will be presented in the introduction. In the second chapter a project formulation is presented with a general assignment description and the project objectives. The process of interest during this internal traineeship is crystallisation. Crystallisation is a separation and purification process, used in the production of a wide range of materials. It involves the formation of one or more solid phases from a liquid phase or amorphous solid phase. A general introduction on crystallisation elaborating on crystallisation phenomena and terminology is presented in the third chapter of this report. The specific crystallisation process that is subject of my internal traineeship is described in the fourth chapter. Although crystallisation is one of the older unit operations, there is little fundamental understanding on the crystallisation process. In order to predict crystallisation process and product specifications, a general first principle model of solution crystallisation has previously been developed [1]. In the fifth chapter the relevant properties of the general model and the adjustments made in order to describe the specific crystallisation process are discussed. Using process measurements, certain parameters of the customised model are estimated by optimising a maximal likelihood formulation. After the parameter estimation and model validation are successfully performed, a validated model is obtained for this specific crystallisation process as discussed in chapter six. The aim is to use this model to configure a Model Predictive Controller (MPC). This type of advanced process control can be used to keep the process within its constraints during normal operation but moreover, to predict an optimal trajectory for highly non-linear events such as process start-up, grade-change and shut-down. MPC was not applied due to difficulties in obtaining a stable linear model and lack of time. In chapter seven the efforts made to obtain a stable linear model are described. Finally, conclusions are drawn and recommendations for further research are presented.
2 Project formulation

2.1 Assignment

The description of the assignment of my internal traineeship at Process Systems Enterprise Ltd. is:

'Modelling of industrial crystallisation processes for Kemira with subsequent configuration of Advanced Process Control systems using the same model.'

2.2 Objectives

In the scope of this internal traineeship the objectives are:

1. To learn how to use the gPROMS software for modelling, simulation, parameter estimation and optimisation.

2. To achieve understanding of the basis principles of crystallisation and how these basic principles are modelled.

3. To learn how gPROMS operates and communicates online in the INCA environment and the Model Predictive Control and State Estimator applications in particular.

3 Crystallisation

Crystallisation is a separation and purification process, used in the production of a wide range of materials. It involves the formation of one or more solid phases from a liquid phase or amorphous solid phase. Crystallisation is one of the older unit operations in the chemical industry and it differs from most unit operations because of the presence of a solid product. The main advantages of crystallisation are a high purity in one process step, a low level of energy consumption and relatively mild process conditions. Although crystallisation is widely used it is still not well understood. This is a disadvantage and problems in terms of product quality and process operation are frequently encountered. One of these problems related to product quality requirements is an excess of fine particles, resulting in bad filterability. Applications of crystallisation can be found in producing inorganic materials such as potassium chloride (fertiliser), organic materials such as paraxylene (raw material for polyester). An enormous number of and diversity in crystallisation processes is found in the pharmaceutical, organic fine chemical and dye industries. This chapter elaborates on crystallisation in terms of the mechanisms involved, the various methods, crystalliser types, operation modes and the different representations of the crystal size distribution (CSD).
3.1 Crystallisation mechanisms

In this subchapter the main crystallisation mechanisms that influence the crystal population are discussed.

3.1.1 Nucleation

Nucleation is the formation of new crystalline material. The driving force for nucleation is *supersaturation*, which is defined as the difference in chemical potential between the solid and the liquid phase. A distinction is made between two mechanisms of formation of new crystalline material, known as primary nucleation and secondary nucleation.

3.1.1.1 Primary nucleation

The formation of new crystalline material from a clear liquid is called primary nucleation. This type of nucleation can be subdivided in heterogeneous and homogeneous nucleation. In heterogeneous nucleation, the liquid contains microscopic foreign particles such as dust or dirt and the primary nucleation takes place on these particles. In homogeneous nucleation, these foreign particles are absent and primary nucleation occurs as a result of local fluctuations of concentration in the liquid. In practice, the liquid will always contain small particles and heterogeneous nucleation is far more likely to occur than homogeneous nucleation.

3.1.1.2 Secondary nucleation

The formation of new nuclei at the surface of parent crystalline material is referred to as ‘birth’ or secondary nucleation. The main source of parent material in the liquid is *attrition*. Attrition is the discontinuous separation of very small particles from a parent crystal due to collisions of the parent crystal with the impeller in the pump, the vessel wall and other crystals. Whereas primary nucleation requires a very high level of supersaturation, secondary nucleation occurs at a moderate level.

3.1.2 Growth

The next step in the crystallisation process is the growth of the small sized particles formed during nucleation. In the absence of agglomeration and breakage, growth, together with nucleation, determines the final particle size distribution of the crystal population. The driving force for crystal growth is again supersaturation. Crystal growth is a process of mass transfer, surface integration and heat transfer. The mass transfer step involves the diffusion of growth units such as ions, atoms or molecules towards the crystal surface. Next, orientation and adsorption of the growth unit takes place in the surface integration step. Heat transfer occurs simultaneously with both steps and is usually not rate-limiting apart from melt crystallisation. In general crystals have different growth rates at different surfaces, referring to the increase in length per time of a surface in the direction normal to that specific surface. However, a single linear growth rate of the characteristic crystal length is often used.
3.1.2 Dissolution
Dissolution of crystals takes place when the solution is undersaturated. Dissolution is not quite the opposite of growth as it does not require the surface integration step and the rate-limiting step is therefore mass transfer away from the crystal surface. Therefore, when the dissolution takes place, crystals are easily rounded off as its corners and edges are the regions where mass transfer is least rate limiting.

3.1.3 Attrition and breakage
Attrition and breakage are both a result of crystal collisions with the pump, the vessel wall or other crystals. The impact of these collisions can result in increased internal crystal stress. The stress will accumulate with repeated collisions, ultimately leading to crystal fracture. The distinction between attrition and breakage is made by the size of the particles after the original crystal has fractured. Breakage is referred to as the separation of a crystal into two or more similar sized crystals. The separation of a crystal into one slightly smaller crystal and many much smaller fragments is named attrition. The amount of impact energy required for breakage is considerably more then for attrition. As stated before in the part on secondary nucleation, attrition is a main source for parent material from which new crystals are born.

3.1.4 Agglomeration
The mass formed by the cementation of individual particles is referred to as an agglomerate. For agglomeration to take place, first of all two or more crystals have to collide. When these crystals are held together by interparticle forces, such as Van der Waals, electrostatic and steric forces, they form a mass called an aggregate. Growth between the crystals in the aggregate is the final cementation step, resulting in an agglomerate. In solution crystallisation processes, agglomeration is usually an undesired phenomenon as the agglomerates can entrap mother liquid. Mother liquor inclusions can result in caking behaviour downstream the crystallisation process or during storage. Caking means the inclusions fracture and the contained mother liquid comes out, cementing multiple crystals together as the solvent evaporates and the supersaturated mother liquor crystallises. This gives considerable problems in product storage and processing.

3.2 Crystallisation methods
Several methods used for crystallisation that differ in feed temperature and heat input are briefly discussed in this subchapter and an overview is presented in table 3.1.

<table>
<thead>
<tr>
<th>crystallisation method</th>
<th>feed temperature</th>
<th>heat input</th>
<th>vapour liquid equilibrium</th>
</tr>
</thead>
<tbody>
<tr>
<td>cooling</td>
<td>( T_{\text{crys}} &gt; 0 )</td>
<td>(&lt; 0)</td>
<td>no</td>
</tr>
<tr>
<td>flash-cooling</td>
<td>( T_{\text{crys}} &gt; 0 )</td>
<td>0</td>
<td>yes</td>
</tr>
<tr>
<td>flash-cooling with additional evaporation</td>
<td>( T_{\text{crys}} &gt; 0 )</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>evaporative</td>
<td>( T_{\text{crys}} \leq 0 )</td>
<td>&gt; 0</td>
<td>yes</td>
</tr>
</tbody>
</table>
3.2.1 Cooling crystallisation
Cooling is the most common method of crystallisation as it is energetically favourable and simple to operate. Typically, a high temperature concentrated solution is delivered into a jacketed crystalliser equipped with a stirrer and often a cooling coil to increase the cooling surface area. The solution is stirred and cold water or cooling agent is pumped through the jacket and coil if present. Such ‘natural’ cooling is continued until the temperature of the solution is nearly that of the cooling medium. A disadvantage of the cooling method is the early fouling of the cooling surfaces and feed point with crystallising solute.

3.2.2 Flash-cooling crystallisation
The flash cooling method avoids these fouling problems on the cooling surface encountered in direct cooling crystallisation, at the cost of a more complicated installation. Additional evaporation can be applied to reduce the batch time.

3.2.3 Evaporative crystallisation
Evaporative crystallisation is economically more favourable than cooling crystallisation if the solubility of the solute has very small or ‘inverse’ temperature dependence. In practice, evaporative crystallisers usually operate at constant temperature and near vacuum pressure. Maintaining this near vacuum pressure introduces extra difficulties in operation.

3.3 Crystalliser types
In this subchapter three commonly used crystalliser types are described. They are the Draft Tube Baffle (DTB), the Forced-circulation (FC) and the Fluidized bed crystalliser. An overview of these types in terms of the product median size and average residence time is shown in Figure 3.1.

![Figure 3.1](image_url)  
Figure 3.1 Expected median crystal size range as a function of residence time and nucleation rate.
3.3.1 Draft Tube Baffle (DTB)
The DTB crystalliser shown in Figure 3.2 consists of a body in which growing crystals are circulated from the lower part to the boiling surface by means of a large, rotating propeller circulator.

Surrounding the suspended magma of growing crystals is an annular settling zone from which a stream of mother liquor can be removed bearing fine crystals. The fine crystals in the mother liquor leaving the baffle zone are sent to a following stage, settler, or heat-exchanger in the case of an evaporative-DTB crystalliser. The mother liquor is returned to the suction of the propeller circulator after the fines have been destroyed by heating, mixing with dilute feed or water, depending on the flow sheet. Incoming feed is also mixed at the eye of the propeller, resulting in a temperature rise in the order of 1 °C. Boiling takes place in the top zone, where the level of supersaturation is higher than the average level in the crystalliser. DTB crystallisers are often equipped with an elutriation leg below the body to classify the crystals. When destruction of fines is not needed or wanted, the settling zone is omitted and the internal circulation rate is set to have the minimum nucleating influence on the suspension, whilst ensuring suspension of the crystals. This type of crystalliser is referred to as the Draft Tube (DT) crystalliser.
3.3.2 Forced-circulation (FC)

The crystalliser shown in Figure 3.3 is known as the forced-circulation (FC) crystalliser.

The crystalliser consists of a body that is sized for vapour release with a liquid level high enough to enclose a volume that is sufficient for retention of the growing crystals. The circulation rate must be large enough to maintain suspension of the crystals present. Suction from the lower portion of the body passes though a circulation pump and a heat exchanger and back to the body through either a tangent or a vertical inlet. The heat exchanger is omitted in those cases where adiabatic cooling is sufficient to produce a yield of crystals. The most common use of this crystalliser is as an evaporative crystalliser with materials having relatively flat solubility or inverted solubility. This type of crystalliser is probably the most widely used, found in diameter sizes varying from 0.6 to over 12 meter. The FC crystalliser is recommended for the crystallisation of a wide range of organic and inorganic compounds requiring a high evaporation rate such as NaCl, sodium sulphate, citric acid and sugar.
3.3.3 Fluidized bed

The third and final type of crystalliser discussed in this subchapter is the fluidized bed or Oslo crystalliser. This type of crystalliser is especially suited for producing larger, coarser crystals, due to the fact that this crystalliser does not have a pump displacing the crystal suspension. In the fluidized bed design, the crystals are kept in suspension in a fluidized bed, which is maintained by a large circulation of mother liquor. Figure 3.4 is a schematic representation of a fluidized bed crystalliser.

![diagram](image-url)

Figure 3.4 Swenson fluidized bed crystalliser.

Hot concentrated feed solution is mixed with a circulated stream of liquid and pumped to the vaporiser where solvent is being evaporated adiabatically. Liquid that is supersaturated at this point leaves the vaporiser through the downcomer and enters the suspension chamber at the lower part. Here the incoming fluid agitates the crystals present in the suspension chamber and maintains them in a fluidized condition. A portion of the coarsest crystals can be removed continuously by pumping the slurry directly from the lower part of the suspension chamber. In the upper part of the suspension chamber, the product crystals separate from the mother liquor leaving only a small quantity of fines to recirculate with the incoming feed solution. Unwanted fine crystals are dissolved by mixing with solvent and returned to the crystalliser where the solvent will be evaporated.
3.4 Operation modes

The three modes in which a crystalliser can be operated are discussed in this subchapter.

3.4.1 Batch

For applications involving relatively small amounts of material (under 50 tons of product per day) it is often convenient to use a batch crystalliser. Another reason to make use of a batch process is when losses must be kept to a minimum, usually when expensive materials are involved. Batch operation also has useful applications where the cooling range is very wide, such as in handling material whose initial feed concentration corresponds to relatively high pressure and whose final mother liquor temperature corresponds to room temperature or significantly lower. In such systems the use of batch crystallisation avoids the shock introduced to the system in continuous equipment by mixing high-temperature feed solutions with relatively low-temperature mother liquor.

3.4.2 Semi-batch

Semi-batch or fed batch operation differs from batch operation in the sense that the crystalliser is fed with mother liquor during the batch. This approach results in higher production yield as more dissolved product is fed than with normal batch operation. It differs from continuous operation as no product is removed during the batch.

3.4.3 Continuous

Batch handling of wet or semidry crystalline materials can present considerable difficulties as compared with storing and handling dried crystalline materials. The present tendency in most processing plants is to use continuous equipment wherever possible. Continuous operation allows adjusting the operating conditions to a relatively fine degree to get the best results in terms of product characteristics and energy consumptions. Furthermore, it permits the use of a smaller labour force and as it results in a continuous utility demand, minimizes the size of boilers, cooling towers and power generating facilities. Continuous operation also minimizes the capital investment, not only in the crystalliser but also in the feed and product storage. Continuous separating devices have been developed to the point where they can operate reliably for long periods of time and the drying of the crystalline products is generally done continuously as well. This results in better product storage as less mother liquor is entrapped thus reducing the chance of caking.

3.5 Crystal Size Distribution (CSD)

The number, characteristic size, shape and volume of crystals are important variables in the crystallisation process. Crystals formed during crystallisation will not have one single size, but varying sizes within a certain range. The population balance equation (PBE) in appendix A.3 describes the time evolution of the crystal size distribution.
(CSD) in terms of number density \( n(L) \), which is the number of crystals of a certain size per unit of volume [6]. The CSD can be described in multiple forms. In this chapter, one example distribution is represented in the most commonly used forms. This example population is defined using the Mixed Suspension Mixed Product Removal (MSMPR) description of the crystal population for simplicity reasons.

\[
n(L) = n_0 \exp\left(-\frac{L}{G\tau}\right)
\]  

(3.1)

Here, \( L \) is an equidistant vector of crystal sizes starting from 25 µm up to 375 µm. Furthermore, \( n_0 \) is the number of crystals with zero length with a value of \( 10^{20} \), the growth rate \( G \) is 0.01 µm/s and the mean retention time \( \tau \) is 4500 seconds.

### 3.5.1 Number density distribution

The PBE describes the time evolution of the CSD in terms of a number density distribution \( n(L) \). This is the starting point from which the various other representations are derived. In Figure 3.5 the example population is shown in terms of a number density distribution.

![Number density distribution](image)

Figure 3.5  Number density distribution of the example population.

Due to the large numbers of crystals, another common form of CSD representation is the natural logarithm of the number density. The natural logarithm of the number density distribution of the example population is displayed in figure 3.6.
Figure 3.6  Natural logarithm of number density distribution of the example population.

3.5.2 Cumulative number density distribution

The number density distribution can be transformed into the cumulative number density distribution using

\[
N(L) = \int_0^L n(L) dL
\]  \hspace{1cm} (3.2)

Measurement data of the CSD is usually in the form of cumulative number density distribution. When the measurement data is used in the PBE, it has to be converted back to the number density distribution with

\[
n(L) = \left. \frac{dN}{dL} \right|_L
\]  \hspace{1cm} (3.3)

The cumulative number density distribution representation of the example population is shown in Figure 3.7.
Figure 3.7 Cumulative number density distribution of the example population.

3.5.3 Volume density distribution

The crystal volume is generally approximated using the volume shape factor \( k_v \) in

\[
V_{cryx} = k_v \cdot L^3
\]  

(3.4)

In a similar way, the volume density distribution is derived from the number density distribution with

\[
\nu(L) = n(L) \cdot k_v \cdot L^3
\]  

(3.5)

Figure 3.8 displays the volume density distribution of the example population with a volume shape factor \( k_v = 1 \).
3.5.4 Cumulative volume density distribution

Similar to the cumulative number density distribution, the cumulative volume density distribution is calculated using

\[ V(L) = \int_0^L v(L) dL = \int_0^L n(L) \cdot L^2 dL \]  

(3.6)

The cumulative volume density distribution of the example population is shown in Figure 3.9.
Figure 3.9  Cumulative volume density distribution of the example population.

3.5.5 Normalised cumulative volume density distribution

The cumulative volume density distribution can be normalised between zero and one by dividing the cumulative volume density distribution by the volume of the largest crystals present. This is equal to dividing by the cumulative volume density at an infinite crystal length.

\[
\tilde{V}(L) = \frac{V(L)}{V(L = \infty)}
\]  

(3.7)

The resulting normalised cumulative density distribution is shown in Figure 3.10.
3.5.6 Quantiles of the distribution

A quantile is the crystal length up to which a certain volume percentage of the total crystal volume is covered. The quantiles are best derived from the normalised cumulative density distribution. For example, the 50% quantile $X_{50}$ is the crystal length that corresponds with the normalised cumulative density value of 0.5. The formal expression is

$$\frac{q}{100} = \hat{F}(X_q) \quad (3.8)$$

The quantiles of the example distribution are shown in Table 3.2

<table>
<thead>
<tr>
<th>Quantile</th>
<th>Volume percentage $q$ [%]</th>
<th>Crystal length [µm]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X_5$</td>
<td>5</td>
<td>61.22</td>
</tr>
<tr>
<td>$X_{10}$</td>
<td>10</td>
<td>77.66</td>
</tr>
<tr>
<td>$X_{16}$</td>
<td>16</td>
<td>92.87</td>
</tr>
<tr>
<td>$X_{25}$</td>
<td>25</td>
<td>112.21</td>
</tr>
<tr>
<td>$X_{50}$</td>
<td>50</td>
<td>161.43</td>
</tr>
<tr>
<td>$X_{75}$</td>
<td>75</td>
<td>221.49</td>
</tr>
<tr>
<td>$X_{84}$</td>
<td>84</td>
<td>252.69</td>
</tr>
<tr>
<td>$X_{90}$</td>
<td>90</td>
<td>280.94</td>
</tr>
<tr>
<td>$X_{95}$</td>
<td>95</td>
<td>314.54</td>
</tr>
</tbody>
</table>
3.5.7 Two log normal distribution

Finally, to describe the initial CSD in the crystallisation model that is discussed in chapter 5, the CSD is parameterised as a bimodal distribution consisting of two log normal distributions on a volume density basis, given by

\[
v(L) = \frac{\varphi}{L} \frac{1}{\ln(\sigma_{g,1})\sqrt{2\pi}} \exp\left(-\frac{(\ln\left(\frac{L}{L_{0g,1}}\right))^2}{2(\ln(\sigma_{g,1}))^2}\right) + \frac{(1-\varphi)}{L} \frac{1}{\ln(\sigma_{g,2})\sqrt{2\pi}} \exp\left(-\frac{(\ln\left(\frac{L}{L_{0g,2}}\right))^2}{2(\ln(\sigma_{g,2}))^2}\right)
\]

This CSD representation introduces five parameters being the median and variance of both log normal distributions and the ratio between the surface areas covered by these distributions. The example population has only one distribution, thus the weighing parameter \(\varphi\) is equal to 1, leaving the parameters of the second log-normal distribution redundant. In Table 3.3 the two log normal parameter values of the example distribution are presented.

<table>
<thead>
<tr>
<th>parameter</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>(L_{0g,1}) ((L_{0-g1}))</td>
<td>152.36</td>
</tr>
<tr>
<td>(\sigma_{g,1}) ((sd_{g1}))</td>
<td>1.9457</td>
</tr>
<tr>
<td>(L_{0g,2}) ((L_{0-g2}))</td>
<td>n.a.</td>
</tr>
<tr>
<td>(\sigma_{g,2}) ((sd_{g2}))</td>
<td>n.a.</td>
</tr>
<tr>
<td>(\varphi) ((Frac_1))</td>
<td>1</td>
</tr>
</tbody>
</table>

4 Process description

In this case study, the process under investigation is the evaporative batch crystallisation of Potassium dihydrogen phosphate (KDP) as performed by Kemira. The process will be discussed in this chapter in terms of crystalliser type and process conditions, mother liquor components, crystallisation method and operation mode and finally the available measurement data.

4.1 Crystalliser type and process conditions
The pilot scale batch crystallisation system is a 20 litre metal reactor vessel, equipped with baffles and a mixer. The heat for evaporation is provided by hot water running through a jacket on the outside and a heating coil on the inside of the reactor vessel, resulting in a mother liquor temperature between 50 and 65 °C. A water ring vacuum pump was used to create the required low pressure. For measurement purposes, pneumatic control valves were sequentially manipulated to withdraw a sample from the suspension.

### 4.2 Mother liquor components

The mother liquor contains three components, of which the crystallising component is Potassium dihydrogen phosphate (KDP). Water is used as a solvent and Potassium hydroxide (KOH) is added in order to adjust the pH-level of the solution. The states in which these components are assumed to occur are presented in Table 4.1.

<table>
<thead>
<tr>
<th>component</th>
<th>state</th>
</tr>
</thead>
<tbody>
<tr>
<td>KDP</td>
<td>solid, liquid</td>
</tr>
<tr>
<td>KOH</td>
<td>liquid</td>
</tr>
<tr>
<td>water</td>
<td>liquid, vapour</td>
</tr>
</tbody>
</table>

The solubility of KDP depends on the concentration of KOH and temperature. Measurements were taken by Kemira and the following solubility correlation has been obtained.

\[ S_{\text{KDP}} = 8.6464 + 1.933 \cdot C_{\text{KOH}} + 0.63545 \cdot T \]  

(4.1)

The measurement data and the correlation prediction are displayed in Figure 4.1.
The initial feed was nearly saturated and contained 46 gram KDP per 100 gram of water and 2.5 gram KOH per 100 gram of water, with a total weight of 24 kg according to [5]

### 4.3 Crystallisation method and operation mode

The crystallisation method used in this specific process is evaporative crystallisation, where part of the solvent is evaporated, resulting in a supersaturated solution. The crystalliser is operated batch wise. Mother liquor is fed into the reactor before the beginning of the batch and the only mass transfer between the reactor and the environment during the batch is solvent vapour leaving the reactor. Heat exchange takes place during the batch, as the reactor loses heat to the surroundings. The hot water in the jacket and coil provide the heat required to maintain a constant temperature in the reactor.

### 4.4 Available measurement data

Experiments have been carried out in which the manipulated variables were the flow rate of the heating water and the batch time [5]. The flow rate of the heating water has a strong effect on the rate of solvent evaporation. Therefore, this evaporation rate has been measured indirectly during the experiments. Another variable that is measured indirectly is yield, which is the weight percentage of KDP crystals.

\[
yield = \frac{M_{KDP,S}}{M_{KDP,S} + M_{KDP,L} + M_{water,L} + M_{KOH,L}} \cdot 100\%
\]  

(4.2)
An overview of the experiments and the corresponding process variables is provided in Table 4.2.

Table 4.2  Overview of experiments and process variables

<table>
<thead>
<tr>
<th>Experiment</th>
<th>Flow rate of heating water [l/h]</th>
<th>Average evaporation rate [kg/h]</th>
<th>Batch time [min]</th>
<th>Final yield [wt%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1320</td>
<td>6.08</td>
<td>110</td>
<td>36.4</td>
</tr>
<tr>
<td>2</td>
<td>1035</td>
<td>4.88</td>
<td>121</td>
<td>29.7</td>
</tr>
<tr>
<td>3</td>
<td>780</td>
<td>3.32</td>
<td>152</td>
<td>21.0</td>
</tr>
<tr>
<td>4</td>
<td>550</td>
<td>2.31</td>
<td>181</td>
<td>15.5</td>
</tr>
<tr>
<td>5</td>
<td>345</td>
<td>1.19</td>
<td>300</td>
<td>12.4</td>
</tr>
</tbody>
</table>

A laser scatter (LS) analyser is used to measure the crystal size distribution (CSD) at two moments during the batch. The data is represented in the form of a cumulative normalised volume fraction distribution. As an example the available data from experiment 1 is displayed in Figure 4.2.

A laser scatter (LS) analyser is used to measure the crystal size distribution (CSD) at two moments during the batch. The data is represented in the form of a cumulative normalised volume fraction distribution. As an example the available data from experiment 1 is displayed in Figure 4.2.

Figure 4.2  CSD of experiment 1 measured with LS analyser.

5  Crystallisation process model

The first principle crystallisation model that is used to describe the crystallisation process consists of a generic part and a process specific part. The generic part contains among other things the equations of conservation of mass (Appendix B.1) and enthalpy (Appendix B.2), the population balance equation (Appendix B.3) and the
discretisation method. The process specific part of the model includes the kinetic model, the physical properties of the mother liquor components, the solubility equation and the vapour-liquid equilibrium. In this chapter, the applied kinetic model is introduced and simulations are carried out to investigate the effect of the discretisation grid type and the number of grid points on accuracy and simulation time. Furthermore, exploratory simulations are performed to investigate the sensitivity of the model outputs in relation to the inputs.

5.1 Kinetic model

The kinetic model describes the formation of new crystals, crystal growth and dissolution. For the Kemira process it is assumed that during the batch no new crystals are formed, neither by attrition nor primary nucleation. This process gets its nuclei from primary nucleation and the resulting initial CSD will be estimated. From that point, the crystals are assumed to grow size dependant. The corresponding crystal growth rate equation is

\[ G(L) = k \left(1 + a \cdot L\right)^b \cdot \sigma^c \]  

Making use of the relative supersaturation defined by

\[ \sigma = \frac{C_{KDP} - C_{KDP_{sat}}}{C_{KDP_{sat}}} \]  

This kinetic model introduces four unknown parameters that need to be estimated together with the initial CSD parameters using the available measurements.

5.2 Grid type and number of grid points

The modelling tool gPROMS can simultaneously solve mixed systems of algebraic equations (AEs) and ordinary differential equations (ODEs) with respect to time. These systems are referred to as systems of differential algebraic equations (DAEs). Beside DAEs, the crystallisation model contains a partial differential equation (PDE) and a number of integral equations (IEs). In order to solve these equations in gPROMS, they must be reduced to a set of DAEs. These reductions involve all derivatives with respect to independent variables other than time. Therefore, the independent variable domains need to be discretised. The numerical methods for reduction require an equidistant grid. However, it can be favourable to make use of a higher resolution locally. To breach this gap, an intermediate normalised crystal size domain with equidistant spacing between the grid points is introduced. This intermediate grid allows other grid types to be applied as long as they are mapped to the intermediate grid that is used for reducing the PDE and IEs to DAEs.

The two grid types that are implemented in the model are a linear and a logarithmic grid. Apart from the grid type, the number of grid points can also be specified in the model. To investigate the effect of grid type and the number of grid points on the
accuracy of the solution, a number of simulations of experiment one is carried out with both a linear and a logarithmic grid using 50, 100, 200 and 400 grid points for both grid types. The variable of interest is the $X50$ quantile, which is the median of the crystal size. The results are displayed in Figure 5.1 and in more detail in Figure 5.2.
Figure 5.1  Median crystal size for various grid types and number of grid points.

Figure 5.2  Median crystal size for various grid types and number of grid points (zoomed in).
From Figure 5.2 it can clearly be seen that a higher number of grid points consistently results in a lower value for the median crystal size. A similar effect is observed for the difference between a linear grid and a logarithmic grid type both using the same number of grid points. A logarithmic grid type consistently results in a lower value for the median crystal size compared to a linear grid type with the same number of grid points. As a simulation with more grid points is expected to lead to a more accurate solution, it follows that a logarithmic grid type should result in a more accurate solution than a linear grid type.

Apart from the median of the crystal size, other variables of interest are the width of the crystal size distribution and the CPU time. A measure for the width of the crystal size distribution is the ratio of the $X_{90}$ and the $X_{10}$ quantile. A summary of the effect of various grid types and number of grid points on the final value of all variables of interest is provided in Table 5.1.

<table>
<thead>
<tr>
<th>variable</th>
<th>lin 50</th>
<th>log 50</th>
<th>lin 100</th>
<th>log 100</th>
<th>lin 200</th>
<th>log 200</th>
<th>lin 400</th>
<th>log 400</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X_{50}$ [µm]</td>
<td>250.85</td>
<td>243.59</td>
<td>239.08</td>
<td>223.41</td>
<td>217.39</td>
<td>214.57</td>
<td>211.46</td>
<td>210.29</td>
</tr>
<tr>
<td>$X_{90}$ / $X_{10}$</td>
<td>2.0012</td>
<td>2.4942</td>
<td>1.7108</td>
<td>1.8707</td>
<td>1.5161</td>
<td>1.5577</td>
<td>1.3705</td>
<td>1.3816</td>
</tr>
<tr>
<td>CPU [s]</td>
<td>202.36</td>
<td>102.94</td>
<td>949.22</td>
<td>387.82</td>
<td>5011.3</td>
<td>1486.2</td>
<td>24207</td>
<td>5019.6</td>
</tr>
</tbody>
</table>

The choice for a logarithmic grid with 100 grid points is considered to be a reasonable compromise between accuracy and CPU time and this combination will be applied in further simulations and model validations.

### 5.3 Exploratory simulations

Now the crystallisation model has been adjusted to the specific Kemira process. Certain parameters of the model are unknown and measurements will have to be used to perform parameter estimation and model validation. First of all, exploratory simulations are carried out to determine the sensitivity of the measured variables with respect to the key parameters. These key parameters can be divided into three groups as shown in Table 5.2.

<table>
<thead>
<tr>
<th>“known”</th>
<th>“unknown”</th>
<th>kinetic model</th>
</tr>
</thead>
<tbody>
<tr>
<td>process conditions</td>
<td>initial CSD</td>
<td>$k_g$</td>
</tr>
<tr>
<td>$M_{T,0}$</td>
<td>$L_{0_g1}$</td>
<td>$a$</td>
</tr>
<tr>
<td>$E$</td>
<td>$sd_g1$</td>
<td></td>
</tr>
<tr>
<td>$L_{0_g2}$</td>
<td>$sd_g2$</td>
<td>$b$</td>
</tr>
<tr>
<td>$sd_g2$</td>
<td>$Frac_1$</td>
<td>$c$</td>
</tr>
</tbody>
</table>

From the documentation on the experiments carried out by Kemira [5], values are obtained for the process conditions parameters. The initial CSD and kinetic model
parameters are to be estimated making use of the data available from measurements of the yield and the CSD in terms of cumulative volume fraction.

### 5.3.1 Final yield

The influence of the process conditions parameters on the final yield is examined in several simulations of experiment 1. The sensitivity of the final yield with respect to the total initial mass of mother liquor $M_{T,0}$ is displayed in Figures 5.3a and 5.3b.

![Figure 5.3a](image) Sensitivity of yield to $M_{T,0}$.  
![Figure 5.3b](image) Relative sensitivity of yield to $M_{T,0}$.

As becomes clear from Figures 5.3a and 5.3b, the final yield strongly depends on the total initial mass of mother liquor. An increase in $M_{T,0}$ results in a decrease of yield, which is expected as $M_{T,0}$ is the denominator of equation (4.2), the definition of yield. The value for the initial total mass is not known within 2 significant digits from [5], this introduces uncertainty in the yield measurements.

The sensitivity of the final yield with respect to the average evaporation rate $E$ is displayed in Figures 5.4a and 5.4b.

![Figure 5.4a](image) Sensitivity of yield to $E$.  
![Figure 5.4b](image) Relative sensitivity of yield to $E$.

From Figures 5.4a and 5.4b it can be seen that the final yield is highly sensitive to the value of the average evaporation rate $E$. This can be explained by considering that when more solvent (water) evaporates, more solids crystallise due to oversaturation. An increase of 5% in $E$ results in an increase of over 10% in final yield. Again this introduces uncertainty in the yield measurements.
Since an average value for the evaporation rate is used, several simulations have been carried out to investigate the influence of various evaporation schemes, leading to the same average evaporation rate. The results are displayed in Figure 5.5.

![Graph showing various evaporation rate schedules with identical average evaporation rates.](image)

Figure 5.5 Various evaporation rate schedules with identical average evaporation rates.

As made clear from Figure 5.5, the evaporation rate schedule does not affect the final yield. As long as the average of the evaporation rate is identical, the same amount of water is evaporated at the end of the batch, resulting in an identical final yield for all schedules.

In a similar way, the sensitivities of the yield with respect to the initial CSD and the kinetic parameters have been investigated. Deviations of up to 50% have been applied and the yield proved to be virtually insensitive to variations in these parameters. This can be explained by considering conservation of mass. Yield is defined as the mass fraction of crystals and the two process parameters $E$ and $M_{T,0}$ directly influence the mass balance of the solution in the reactor. The initial CSD parameters describe the distribution of crystals at the beginning of a batch and the kinetic parameters determine how these distributions evolve in time. Neither of both parameter sets influence the total mass of crystals nor the mass of evaporated solvent. Therefore, it is expected that the final yield is insensitive to both the kinetic and the CSD parameters.

The kinetic and the initial CSD parameters do influence the path of the yield during a batch, like the evaporation rate in Figure 5.5. This can be explained by the interaction between these parameters and the saturation level of the solution. A large population of small crystals has a larger total surface area than a smaller population of bigger crystals with similar total weight. Consequently, a larger population of smaller crystals will take up more solid material, decreasing the level of saturation. This process is also influenced by the kinetic parameters, for they describe size dependant
growth of the crystal population. Although these transitional aspects influence the final CSD in terms of quantiles of the distribution, they result in the same final yield according to the extensive number of simulations.

### 5.3.2 Quantiles

The simulations used to investigate the effect deviation of the key parameters has on the final yield can not be used to investigate its effect on the quantiles. The reason is that the initial solution was slightly undersaturated with the concentrations as stated in the documentation of the experiment [5]. All present crystals dissolved as a result and therefore the initial CSD had no influence on the quantiles at the end of the batch. A new set of experiment simulations is performed, making sure the initial solution is saturated. In Figures 5.6a and 5.6b the results are displayed for deviations in the evaporation rate.

![Figure 5.6a](image1.png) **Sensitivity of X50 to E.**

![Figure 5.6b](image2.png) **Relative sensitivity of X50 to E.**

There appears to be a linear relationship between the evaporation rate and the median crystal size, without a very strong dependency. This can be explained by realising that an increase in E results in an increase of the saturation level, which leads to faster growth of the larger crystals as described by the kinetic model. The results for the total initial mass of mother liquor are presented in Figures 5.7a and 5.7b.

![Figure 5.7a](image3.png) **Sensitivity of X50 to MT,0.**

![Figure 5.7b](image4.png) **Relative sensitivity of X50 to MT,0.**

Here, again an approximate linear relation is observed, this relation is however inverse to the previous one and less sensitive. This behaviour is expected as an
increase in $M_{T,0}$ results in a decrease in supersaturation as the same amount of water is evaporated. Next, the sensitivity of the parameters of the initial CSD is investigated. Although the value of $Frac_1$ is set to 0.5, the $X50$ is very insensitive to both parameters of the second log normal distribution, $L0\_g2$ and $sd\_g2$. The results for the first parameter of the initial CSD, $L0\_g1$ are displayed in Figures 5.8a and 5.8b.

Again we observe a linear relationship between the $L0\_g1$ and the final $X50$, which is expected as both are a measure for the median crystal size. In Figure 5.9a and 5.9b the relationship for $sd\_g1$ is displayed.

The results for variations in $Frac_1$ are shown in Figure 5.10a and 5.10b.
Variations in \(sd\_g1\) and \(Frac\_1\) display a similar effect on \(X50\) although the effect of \(sd\_g1\) is stronger. This explained by considering that an increase in \(sd\_g1\) results in a wider initial CSD and the increase in smaller crystals leads to a decrease in supersaturation.

Now the influence of all these parameter on the final values of the outputs \(yield\) and \(X50\) is investigated, the conclusion can be drawn that all these relations are reasonably linear. However, the behaviour of especially \(X50\) during the beginning of a batch is strongly non-linear as can be seen in Figure 5.11.

![Figure 5.11 \(X50\) as a function of time during simulation of batch experiment 1.](image)

### 6 Model validation

The unknown model parameters are the initial CSD and the kinetic parameters. These parameters are to be estimated using the gPROMS built-in routine that combines formal mathematically based optimisation techniques with the maximum likelihood formulation for the objective function

\[
\Phi = \frac{N}{2} \ln(2\pi) + \frac{1}{2} \min_{\theta} \left\{ \sum_{i=1}^{NE} \sum_{j=1}^{NV} \sum_{k=1}^{NM} \left[ \ln\left(\sigma_{ijk}^2\right) + \frac{(\bar{z}_{ijk} - z_{ijk})^2}{\sigma_{ijk}^2} \right] \right\} \\
(6.1)
\]

where

- \(N\) total number of measurements taken in all experiments
- \(\theta\) set of model parameters to be estimated, which may be subject to given bounds, i.e. \(\theta_j \leq \theta \leq \theta_{uj}\).
- \(NE\) number of experiments performed.
$NV_i$ number of variables measured in experiment $i$.
$NM_{ij}$ number of measurements of variable $j$ in experiment $i$.
$\sigma^2_{ijk}$ variance of the $k^{th}$ measurement of variable $j$ in experiment $i$.
$z_{ijk}$ $k^{th}$ measured value of variable $j$ in experiment $i$.
$z_{ijk}^m$ $k^{th}$ (model-) predicted value of variable $j$ in experiment $i$.

It is assumed that the variance of the measurements is 10% of the measured value.

### 6.1 Objective function gradient

A first attempt is made to estimate the first initial CSD parameter, $L0\_g1$ of experiment 1 using CSD data in the form of quantiles. Although the execution output stated an optimal solution was found, this first attempt did not result in a satisfactory parameter value as it was either on its bounds or very close to its initial value. A possible reason for this behaviour is the very high value (in the order of $10^{30}$) $gPROMS$ calculated for the objective function gradient, $gobj$. To determine if this high value is correct, $gobj$ has been determined alternatively using finite differencing. This was done by applying small perturbations to the initial parameter value and observing the change in the value of the objective function at the start of the first iteration. The alternatively determined gradient was calculated in three ways using backward, forward and central differencing according to (6.2), (6.3) and (6.4) respectively

\[
\frac{df(x)}{dx} = \frac{f(x) - f(x - \Delta x)}{\Delta x} \quad (6.2)
\]
\[
\frac{df(x)}{dx} = \frac{f(x + \Delta x) - f(x)}{\Delta x} \quad (6.3)
\]
\[
\frac{df(x)}{dx} = \frac{f(x + \Delta x) - f(x - \Delta x)}{2\Delta x} \quad (6.4)
\]

In order to compare the results, the same procedure was carried out for two other parameter estimation cases. From the same process, the estimation of the initial total mass $M_{T,0}$ was used. As an absolute benchmark, a parameter estimation example case is used, from which a parameter $K10$ is estimated. A summary of the results with perturbations of 1% around the initial parameter values is presented in Table 6.1

<table>
<thead>
<tr>
<th>parameter</th>
<th>$gobj$</th>
<th>backward finite differencing</th>
<th>forward finite differencing</th>
<th>central finite differencing</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L0_g1$</td>
<td>$6.5551 \times 10^{30}$</td>
<td>189.56</td>
<td>196.18</td>
<td>192.87</td>
</tr>
<tr>
<td>$M_{T,0}$</td>
<td>$3.9952 \times 10^{13}$</td>
<td>-96.174</td>
<td>-81.183</td>
<td>-88.679</td>
</tr>
<tr>
<td>$K10$</td>
<td>34456</td>
<td>34755</td>
<td>34163</td>
<td>34459</td>
</tr>
</tbody>
</table>

Since the alternatively determined objective function gradient of the benchmark process, with parameter $K10$, is nearly equal to $gobj$ as calculated in the $gPROMS$
parameter estimation routine, there is reason to believe the $gobj$ calculation in the crystallisation process is not correct.

Possible causes of the incorrect $gobj$ calculations are:

1. The ‘Foreign Object’ calculates incorrect derivatives
2. The tolerance setting are not tight enough

Ad.1
As the Foreign Object calculations are not required while estimating $M_{T,0}$, the Foreign Object call is commented out in order to avoid the Foreign Object calculating any derivatives. This had no effect on the value of $gobj$ while estimating $M_{T,0}$.

Ad. 2
The absolute and relative tolerance settings of the DAE solver are tightened. By default, they both have a value of $10^{-5}$. The resulting values of $gobj$ calculated by gPROMS with corresponding absolute and relative tolerance settings are presented in Table 6.2.

<table>
<thead>
<tr>
<th>Absolute tolerance</th>
<th>Relative tolerance</th>
<th>$gobj$</th>
<th>central finite differencing</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-5}$</td>
<td>$10^{-5}$</td>
<td>$-8.14145 \cdot 10^{33}$</td>
<td>$-88.679$</td>
</tr>
<tr>
<td>$10^{-7}$</td>
<td>$10^{-7}$</td>
<td>$-3.40982 \cdot 10^9$</td>
<td>$-88.679$</td>
</tr>
<tr>
<td>$10^{-8}$</td>
<td>$10^{-8}$</td>
<td>$-88.5272$</td>
<td>$-88.679$</td>
</tr>
<tr>
<td>$10^{-8}$</td>
<td>$10^{-5}$</td>
<td>$8.48624 \cdot 10^{36}$</td>
<td>$-88.679$</td>
</tr>
<tr>
<td>$10^{-5}$</td>
<td>$10^{-8}$</td>
<td>$-88.3874$</td>
<td>$-88.679$</td>
</tr>
</tbody>
</table>

The results in Table 6.2 lead to believe that the value of the relative tolerance of the DAE solver is the essential setting and that it should be set tight enough for a correct calculation of $gobj$.

### 6.2 Parameter estimation

Both the absolute and the relative tolerance are set to $10^{-8}$ and parameter estimation is performed using measurement data of experiments 1 to 5. First, attempts are made to estimate the initial CSD parameters using size fractions of the CSD as measured values. This proved unsuccessful and therefore the quantiles of the experimental CSD data are calculated and used in stead of size fractions. Various experiments and combinations of experiments are used to perform the parameter estimation and an optimal solution was found with measurement data from experiments 2 and 4 combined. It was found that the initial CSD can best be described with a single log normal distribution, resulting in $Frac_1 = 1$ and therefore the parameters $L0_g2$ and $sd_g2$ are made redundant and thus not estimated. The results are presented in Table 6.3.
Table 6.3  Results model validation using experiments 2 and 4.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Optimal estimate</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L_{0 _g1}$</td>
<td>69.9364</td>
</tr>
<tr>
<td>$sd_{_g1}$</td>
<td>1.73892</td>
</tr>
<tr>
<td>$L_{0 _g2}$</td>
<td>n.a.</td>
</tr>
<tr>
<td>$sd_{_g2}$</td>
<td>n.a.</td>
</tr>
<tr>
<td>Frac_{1}</td>
<td>1</td>
</tr>
<tr>
<td>$k_{g}$</td>
<td>18.6461</td>
</tr>
<tr>
<td>$a$</td>
<td>0.00241657</td>
</tr>
<tr>
<td>$b$</td>
<td>1.29629</td>
</tr>
<tr>
<td>$c$</td>
<td>1.49545</td>
</tr>
</tbody>
</table>

A comparison is made between the model predictions and the measurement values in Figures 6.1 and 6.2 for experiments 2 and 4 respectively.

Figure 6.1  Comparing measured values and model prediction for experiment 2
As can be observed in Figures 6.1 and 6.2, the model predictions match better in experiment 2 than in experiment 4. The gPROMS parameter estimation output file reports the $t$-95% value of the estimated parameters is smaller than a reference value for all parameters. Therefore, all estimates are considered statistically insignificant. It is however the best estimate available for these parameters and this set of parameters will be applied.

7 Dynamic optimization

7.1 Software architecture

The intention is to use tools from the INCA suite to apply Model Predictive Control (MPC) to a fictive process, simulated by the non-linear gPROMS model. Matlab is used to apply input signals and to trigger gPROMS. Therefore, connections need to be established between the INCA tools, gPROMS and Matlab via a data server. The application setup is shown in Figure 7.1 and the individual applications and their connections are described in the following subchapters. The communication configuration files for the Kemira process can be found in Appendix B.
7.1.1 Data server
The data server is an OPC server that is used as a module where various clients can interchange data in a sophisticated manner. The communication is specified in the configuration file `PlantSignalDefinition.csv` in Appendix B.1. This file contains the definitions of Data signals specifying process inputs and outputs, INCA signals to configure, start and stop the applications attached to the data server and Simulator control signals to define the gPROMS tasks such as simulate and linearise.

7.1.2 Scheduler
The scheduler iteratively triggers the different applications in a specific order, which is defined in the configuration file `scheduler.csv` in Appendix B.2. Besides the order, the timing of the sequence can also be specified.

7.1.3 gPROMS
The non-linear gPROMS model simulates the actual crystallisation process. In the gPROMS task named `OPCCOM`, see Appendix B.3, the process inputs and outputs are defined and the tag names as used in the data server are assigned. Furthermore, in this task various actions are specified such as simulate and linearise that can be called from Matlab.

7.1.4 Matlab
Like the scheduler, Matlab can be used to trigger the applications attached to the data server. For analysis purposes, Matlab is more convenient due to the ability to create user-defined signals and perform actions that return results to the Matlab environment. The scheduler is more convenient for actual operation, once the entire system of applications is working properly. The communication makes use of a file.
named \textit{mxOPC.dll} and numerous hierarchical m-files have been built and collected in what is called the \textit{EKO engine} to perform a wide variety of actions and calculations.

7.1.5 INCAEngine

INCAEngine is the on-line Model Predictive Controller component of IPCOS Technology's INCA-suite for process identification and control of a plant. It contains the optimization and control algorithms and calculates the control signals. The INCAEngine can implement advanced control systems based on prioritized and least squares control. In this case the algorithm makes use of a state space model of the process to predict the future moves of the outputs and to solve the optimization problem. The configuration file \textit{controller.csv} defines tags such as inputs, outputs and connections between the variables used in the data server, see Appendix B.4. Furthermore, the controller and tuning settings are defined in this file.

7.1.6 INCAview

INCAview is an application to visualize the variables present in INCAEngine. The control engineer can use this tool to maintain the system, check the performance of the controller and change the tuning parameters if the control objective is altered.

7.2 Linearisation

The Model Predictive Controller (MPC) in the INCAEngine makes use of a linear model to predict future outputs of the process. The controller uses predictions of this linear model to calculate optimal process inputs using an optimization algorithm. It may be necessary to use multiple linear model obtained in various operating points to describe the entire non-linear operating region. The linear state space models can be obtained from the non-linear model using gPROMS' built-in \textit{LINEARISE} function.

7.2.1 The \textit{LINEARISE} function

The gPROMS model contains multiple non-linear Differential Algebraic Equations (DAEs) that in general can be written as

\[ f(x, \dot{x}, u, y) = 0 \]  \quad (7.1)

Where \( x \), \( u \) and \( y \) are vectors containing respectively the model variables, inputs and outputs. The \textit{LINEARISE} functions returns a continuous linear state space representation of the non-linear model in a specific operating point in the form of

\[
\dot{X} = AX + BU \\
Y = CX + DU
\]  \quad (7.2)

Here \( X \), \( U \) and \( Y \) are vectors with the linear model states (which gPROMS determines automatically with the least number possible), inputs and outputs respectively. These vectors and the matrices \( A \), \( B \), \( C \) and \( D \) are returned in a gPROMS output file.
For the Kemira process the following input $u$ and outputs $y_1$ to $y_4$ are defined:

\[
\begin{align*}
    u &= \text{evaporation rate} \\
    y_1 &= X_{10} \\
    y_2 &= X_{50} \\
    y_3 &= X_{90} \\
    y_4 &= \text{mass vapour produced}
\end{align*}
\]

A linear model is extracted during a batch simulation, it contains 55 states and is instable as some eigenvalues of the system matrix $A$ are positive. INCAEngine cannot make use of an instable model due to the fact that the model predictions would go to infinity. Several actions are taken to try to work around this problem, these actions are described in the following chapters.

7.2.2 Obtaining a stable linear model

First it is believed the non-linear model is marginally stable or even unstable itself and proportional feedback is applied on the total crystal mass (variable name \textit{crys} \textit{mass}) to internally stabilise the non-linear model. Although the output \textit{mass vapour produced} is stabilised, the linear models remain instable. Next, proportional feedback is applied on one of the state variables named $\text{log} \_\text{num}(25)$. Again, the corresponding obtained linear model is instable. Finally, a simplified crystallisation model is used to compare the results for the stability of linear models extracted at various moments. This simple model has the following input $u$ and outputs $y_1$ to $y_4$:

\[
\begin{align*}
    u &= \text{evaporation rate} \\
    y_1 &= M_0 \\
    y_2 &= M_1 \\
    y_3 &= M_3 \\
    y_4 &= \text{mass vapour produced}
\end{align*}
\]

The simple model does not contain a population balance equation, but makes use of moments of the distribution. The zeroth, first and third moments of the distribution, named $M_0$, $M_1$ and $M_3$, represent the total number of crystals, the total size of crystals and the total volume of the crystals respectively. Although not all the output signals are similar to the Kemira model, both models describe evaporative batch crystallisation and the input for both models is the evaporation rate of its solvent.

An input signal in the form of an impulse is applied to both the Kemira and the simple model once they are in steady state. In the case of evaporative batch crystallisation, steady state is only obtained when the input is zero. Linear models A, B and C are extracted at different moments as shown in Figure 7.1.
Figure 7.1  Input signal and moments when linear models A, B and C are extracted.

The responses of the non-linear Kemira and the simple model are displayed in Figures 7.2 and 7.3 respectively.

Figure 7.2  Response of non-linear Kemira model.
Figure 7.3  Response of non-linear simple model.

From these responses it can be observed that both non-linear models are marginally stable, which can be explained by the pure integrating action that comes from \( u \rightarrow y_d \) given by

\[
\frac{d}{dt}(y_d) = u
\]  

(7.5)

In Figure 7.4 the responses of the non-linear simple model and the three linear models of the simple model are shown.
From Figure 7.4 it can be concluded that none of the linear models of the simple model is unstable. To back up this conclusion, the eigenvalues of the system matrices of the linear models are presented in Table 7.1 and none of these eigenvalues are positive.

The responses of the non-linear and the linear Kemira models are displayed in Figure 7.5.
The responses of the linear model A and C appear to be stable. However, the eigenvalues of the system matrices still contain non-negative entries that are presented in Table 7.2.

Table 7.2 Non-negative eigenvalues of the system matrices in the linear Kemira models.

<table>
<thead>
<tr>
<th></th>
<th>model A</th>
<th>model B</th>
<th>model C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>6.0607e-007</td>
<td>0</td>
<td>2.9141e-011</td>
</tr>
<tr>
<td></td>
<td>4.1087e-023</td>
<td>0.00033413 + 0.00263331i</td>
<td>1.2339e-014 + 3.7588e-013i</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0.00033413 - 0.00263331i</td>
<td>1.2339e-014 - 3.7588e-013i</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.0020003</td>
<td>2.8079e-013</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7.1608e-029</td>
<td>3.4777e-020</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

When simulations are performed with the linear model for an extended period of time, the instable behaviour of model A becomes visible as displayed in Figure 7.6.
Linear model C does not exhibit instable behaviour. Additional linear models are obtained after the impulse in order to investigate their behaviour towards the end of the input signal where model C is obtained. The moments in time when the additional linear models are obtained are presented in Table 7.3.

Table 7.3 Overview of additional linear models obtained after impulse.

<table>
<thead>
<tr>
<th>name</th>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
<th>C5</th>
<th>C6</th>
</tr>
</thead>
<tbody>
<tr>
<td>time [s]</td>
<td>7200</td>
<td>7260</td>
<td>7320</td>
<td>7380</td>
<td>7440</td>
<td>7500</td>
</tr>
</tbody>
</table>

All responses of the additional models are shown in Figure 7.7.
It appears that the instable behaviour decreases as the moment the linear model is obtained lies further from the impulse. This is confirmed by Figure 7.8 where the maximal positive eigenvalues are displayed for the additional linear models.

Figure 7.7 Responses of non-linear (NL) and additional linear (L<sub>C1</sub> to L<sub>C6</sub>) Kemira models.

Figure 7.8 Maximal eigenvalues for additional linear Kemira models L<sub>C1</sub> to L<sub>C6</sub>.
The magnitude of the impulse value is reduced to investigate its effect on the instable behaviour. In Table 7.4 an overview is presented of the maximal eigenvalues of the linear Kemira models obtained with input signals with various impulse values.

<table>
<thead>
<tr>
<th>impulse value</th>
<th>model A</th>
<th>model B</th>
<th>model C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6.0607e-007</td>
<td>0.00033413 + 0.0026333i</td>
<td>2.9141e-011</td>
</tr>
<tr>
<td>0.1</td>
<td>6.0607e-007</td>
<td>6.2595e-006 + 0.00040847i</td>
<td>4.1308e-009</td>
</tr>
<tr>
<td>0.01</td>
<td>6.0607e-007</td>
<td>3.1143e-005</td>
<td>1.0472e-009</td>
</tr>
</tbody>
</table>

Especially the models B, taken at the moment of the impulse, are strongly dependant of the impulse value. The responses are compared in Figure 7.9 using a logarithmic scale to visualize the effect.

Figure 7.9  Responses of linear Kemira model B with various impulse values.

Clearly, the instable behaviour decreases with a decreasing impulse value. In Figure 7.10 the response of linear model B with an impulse value of 0.01 is displayed to shown that this model is however still instable.
The conclusions that can be drawn from these tests and simulations using the gPROMS build-in `LINEARISE` function are first of all that both the Kemira and the simple non-linear model are marginally stable, due to the purely integrating action. Secondly, the linear models obtained for the simple model are all marginally stable as well as the eigenvalues of the system matrices are negative or zero. The quality of these models depends on the moment they are extracted and the model obtained at the end of the input signal gives the best results, which are still not very good. All linear models obtained for the Kemira process are instable as their system matrices contain positive eigenvalues. A smaller impulse value results in decreased instable behaviour of the models obtained at the moment the impulse is applied. In a similar way, models obtained further away from the moment the impulse is applied also have decreasing instable behaviour.

As the value of the input (evaporation rate) has a positive value during an experiment simulation, the `LINEARISE` function cannot be used for the Kemira process to obtain linear models suitable for Model Predictive Control using the INCA tools.

### 7.2.3 System identification

As the gPROMS build-in `LINEARISE` function does not result in stable linear models for the non-linear Kemira model, another approach is followed using system identification. With the process input and corresponding output data available, the deviations around steady state are used to subtract a stable linear FIR or state-space model with the INCAModeler tool. The necessary data is logged with INCATest, another tool from the INCA suite.
System identification proved to be successful in the case of a continuous draft tube crystallisation model; it even produced better predictions than the *LINEARISE* function. This was caused by wrong derivatives in the Foreign Object for the Gahn kinetic model. The inputs for this continuous crystallisation process are the propeller frequency and the volume outflow. A comparison is made between the responses of the non-linear gPROMS response, the linear model obtained using *LINEARISE* and the linear model obtained by system identification. The results for deviations of the 50% quantile of this continuous are displayed in Figure 7.11.

As can be observed in Figure 7.11, the output of the linear model obtained with system identification matches the non-linear model very good, apart from the part where the two different input signals are attached together at the 1250th sample. The output of the model obtained with the *LINEARISE* function is acceptable.

For the system identification of both the Kemira and the simple batch process, the only steady state input is zero, and the deviations can only be positive due to the nature of the input and the batch mode of the process. In Figure 7.12 the input signal for system identification is shown.
The results for the simple batch process are displayed in Figure 7.13.

Here the differences between the non-linear gPROMS model and both the linear models are considerable, except for $y_4$ the mass vapour produced. In the previous
subchapter it became clear that the linear Kemira models obtained were unstable. The linear model obtained with system identification is compared with the non-linear model in Figure 7.14.

![Graphs showing comparison between linear and non-linear models](image)

Figure 7.14   Results system identification for the Kemira model.

Similar to the linear model obtained by system identification for the simple model, \( y_d \) the mass vapour produced is well predicted. In contrast, the \( y_I \) does not match the non-linear output at all. The reason for these large differences for both the Kemira and the simple model could be that there is a lack of information in the logged data. This is probably caused by the fact that the input signal is not a deviation around a steady state value, but only contains positive impulse values.
Conclusions and recommendations

Looking back it can be concluded that most of the work in the assignment description is carried out and that nearly all project objectives specified in the second chapter are met. Unfortunately, the performance of the linear model obtained by system identification is not assessed for Model Predictive Control due to a lack of time. Regarding parameter estimation the following conclusions are drawn:

- For parameter estimation purposes, quantiles are a better way of CSD representation then size fractions.

- The relative and absolute tolerances of the dynamic parameter estimation solver need to be set tight enough ($10^{-8}$ proved to be sufficient in the Kemira case) in order for the objective function gradient to be calculated correctly.

- The results in parameter estimation were obtained using two of the five experiment measurements. Although statistically insignificant, they are the best results available and therefore applied to improve the model predictions.

With respect to the work done to obtain a stable linear model suitable for MPC purposes, the following conclusions are drawn:

- All attempts to obtain a stable linear Kemira model using the gPROMS built-in $\text{LINEARISE}$ function failed. For a simple evaporative batch crystallisation process this approach proved to be successful.

- System identification and the $\text{LINEARISE}$ function produced satisfactory results for a continuous crystallisation process. In the case of both the Kemira and the simple batch model system identification resulted in stable linear models, both with poor predictive quality.

- This lack of predictive quality is probably caused by the limitation of only being able to apply a positive input in the form of an impulse due to the batch character of the process. An evaporative batch crystallisation process is not in steady-state except for the trivial case of zero input. In the continuous case, positive and negative deviations of an input can be applied around a steady state value resulting in a linear model with substantially better predictive quality.

Finally, recommendations for further research on this topic are:

- Investigate why the $\text{LINEARISE}$ function returns instable linear models; this could be caused by numerical issues, inherent to the Kemira model or even inherent to batch process models in general as use of the $\text{LINEARISE}$ function for continuous processes has always been successful in returning stable, linear models so far.

- Determine the influence various input signals have on the predictive quality of linear models obtained by system identification.
• Apply the stable linear models for MPC purposes and assess their performance, possibly in combination with the physical crystallisation process.
References


Nomenclature

\( a \) kinetic parameter [-]
\( b \) kinetic parameter [-]
\( c \) kinetic parameter [-]
\( C \) concentration [g / 100g water]
\( \text{Frac}_{\text{1}} \) relative weighting of the two log-normal distribution [-]
\( G(L) \) linear crystal size dependent growth rate [m · s\(^{-1}\)]
\( k_g \) growth rate constant [-]
\( k_v \) volume shape factor [-]
\( L \) crystal length [\( \mu \text{m} \)]
\( L_{0_{\text{g1}}} \) location parameter of first log-normal distribution [\( \mu \text{m} \)]
\( L_{0_{\text{g2}}} \) location parameter of second log-normal distribution [\( \mu \text{m} \)]
\( m \) mass [kg]
\( M \) molar mass [kg · mole\(^{-1}\)]
\( M_0 \) zeroth moment of the distribution [\#]
\( M_1 \) first moment of the distribution [m]
\( M_2 \) second moment of the distribution [m\(^3\)]
\( M_3 \) third moment of the distribution [m\(^3\)]
\( n(L) \) crystal number density [\# · m\(^{-3}\) · \( \mu \text{m}^{-1}\)]
\( N(L) \) cumulative number density [\# · m\(^3\)]
\( n_0 \) number of crystals with zero length [\#]
\( N_I \) number of liquid/solid inlet streams [\#]
\( N_O \) number of liquid/solid outlet streams [\#]
\( N_P \) number of phases [\#]
\( N_R \) number of reactions [\#]
\( q \) volume percentage [%]
\( r \) reaction rate [mole · s\(^{-1}\)]
\( s d_{\text{g1}} \) spread parameter of first log-normal distribution [-]
\( s d_{\text{g2}} \) spread parameter of second log-normal distribution [-]
\( t \) time [s]
\( T_{\text{crys}} \) crystallisation temperature [K]
\( v(L) \) volume density [\# · m\(^3\) · m\(^{-3}\) · \( \mu \text{m}^{-1}\)]
\( V \) volume [m\(^3\)]
\( V(L) \) cumulative volume density [\# · m\(^3\)]
\( \tilde{V} \) normalised cumulative volume density [\# · m\(^3\) · m\(^{-3}\)]
\( X_{q} \) \( q \) % quantile [\( \mu \text{m} \)]

Greek

\( \phi \) relative weighting of the two log-normal distribution [-]
\( \phi_{\text{mol}} \) molar flow rate [mol · s\(^{-1}\)]
\( \phi_v \) volumetric flow rate [m\(^3\) · s\(^{-1}\)]
\( \phi_m \) mass flow rate [kg · s\(^{-1}\)]
\( \nu \) stoichiometric coefficient [-]
\( \sigma \) relative supersaturation [-]
\( \Phi \) objective function value [-]
\( \tau \) mean retention time [s]
Subscripts

\(aggl\) agglomeration
\(attr\) attrition
\(break\) breakage
\(crys\) crystal
\(dis\) dissolution
\(grow\) growth
\(in\) ingoing flow
\(KDP\) potassium dihydrogen phosphate
\(KOH\) potassium hydroxide
\(L\) liquid phase
\(nucl\) primary nucleation
\(out\) outgoing flow
\(S\) solid phase
\(V\) vapour phase
\(water\) water

Indices

\(i\) index of liquid phase component
\(k\) index of control volume inlet
\(l\) index of control volume outlet
\(p\) index of liquid phase reaction
\(q\) index of solid phase
Appendix A  Equations of conservation

Here, the equations of conservation are introduced for the general model of a control volume with volume $V$, temperature $T$, pressure $P$, number of liquid phase components $NC_L$, number of liquid phase reactions $NR_L$, number of solid (crystalline) phases $NP_S$, with $NI$ inlets for solid and liquid phases, $NO$ outlets for solid and liquid phases and one vapour outlet.

A.1 Mass balance for liquid phase component $i$

\[
\frac{dm_{L,i}(t)}{dt} = \sum_{k=1}^{NI} \phi_{m,L,i,\text{in},k}(t) - \sum_{l=1}^{NO} \phi_{m,L,i,\text{out},l}(t) - \phi_{m,V,i,\text{out}}(t) + M_i \cdot \sum_{p=1}^{NR} \nu_{L,p,i} \cdot r_p + M_i \cdot \sum_{q=1}^{NP} \nu_{S,q,i} \cdot \left[ \phi_{\text{mol,nucl},q} + \phi_{\text{mol,grow},q} - \phi_{\text{mol,diss},q} \right] \quad (A.1)
\]

Initial condition:

\[ m_{L,i}(t = 0) = m_{L,i,0} \quad i = 1, \ldots, NC_L \]

A.2 Enthalpy balance

By assuming the kinetic energy, potential energy and shaft work can be neglected, the energy balance is reduced to the following enthalpy balance:

\[
\frac{\dot{d}H}{dt} = \sum_{k=1}^{NI} \phi_{H,k} - \sum_{l=1}^{NO} \phi_{H,l,\text{out}}(t) + \phi_{H,V,\text{out}} + \dot{Q} \quad (A.2)
\]

Initial condition:

\[ H(t = 0) = H_0 \]
A.3 Population balance for solid phase $q$

\[
\frac{\partial n_q(L,t) \cdot V(t)}{\partial t} = -V(t) \frac{\partial (n_q(L,t) \cdot G_q(L,\sigma))}{\partial L}
\]

number rate in and out

\[
+ \sum_{k=1}^{Nf} \varphi_{V,in,k} \cdot n_{q,in,k} - \sum_{l=1}^{NO} \varphi_{V,out,l} \cdot n_{q,out,l}
\]

number production rate due to primary nucleation and dissolution at the critical nucleus size

\[
+ (\varphi_{n,q,nucl} - \varphi_{n,q,dis}) \cdot V(t)
\]

number production rate due to attrition, breakage and agglomeration

\[
+ (\varphi_{n,q,attr} + \varphi_{n,q,break} + \varphi_{n,q,aggl}) \cdot V(t)
\]

Initial condition:

\[
n_q(L,t = 0) = n_{q,0}(L) \quad q = 1,\ldots, NP_S
\]

Boundary condition:

\[
n_q(L = 0,t) = \frac{B_{q,0}(t)}{G_q(L = 0,t)} \quad \text{if } G \geq 0 \quad q = 1,\ldots, NP_S
\]

\[
n_q(L = \infty, t) = 0 \quad \text{if } G < 0 \quad q = 1,\ldots, NP_S
\]
## Appendix B  Configuration files

### B.1 PlantSignalDefinition.csv

<table>
<thead>
<tr>
<th># Signal</th>
<th>Attribute</th>
<th>Init. Value</th>
<th>Type</th>
<th>Unit</th>
<th>Section</th>
<th>Disp. Name</th>
</tr>
</thead>
<tbody>
<tr>
<td># Process OUTPUTS #</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X10</td>
<td>pv</td>
<td>220.84</td>
<td>D</td>
<td>PO</td>
<td>value</td>
<td>X10</td>
</tr>
<tr>
<td>X10</td>
<td>status</td>
<td>0</td>
<td>L</td>
<td>PO</td>
<td>status</td>
<td>X10</td>
</tr>
<tr>
<td>X10</td>
<td>ideal</td>
<td>220.84</td>
<td>D</td>
<td>PO</td>
<td>value</td>
<td>X10</td>
</tr>
<tr>
<td>X50</td>
<td>pv</td>
<td>656.75</td>
<td>D</td>
<td>PO</td>
<td>value</td>
<td>X50</td>
</tr>
<tr>
<td>X50</td>
<td>status</td>
<td>0</td>
<td>L</td>
<td>PO</td>
<td>status</td>
<td>X50</td>
</tr>
<tr>
<td>X50</td>
<td>ideal</td>
<td>656.75</td>
<td>D</td>
<td>PO</td>
<td>value</td>
<td>X50</td>
</tr>
<tr>
<td>X90</td>
<td>pv</td>
<td>900</td>
<td>D</td>
<td>PO</td>
<td>value</td>
<td>X90</td>
</tr>
<tr>
<td>X90</td>
<td>status</td>
<td>0</td>
<td>L</td>
<td>PO</td>
<td>status</td>
<td>X90</td>
</tr>
<tr>
<td>X90</td>
<td>ideal</td>
<td>900</td>
<td>D</td>
<td>PO</td>
<td>value</td>
<td>X90</td>
</tr>
<tr>
<td>total_mass_evap</td>
<td>pv</td>
<td>0</td>
<td>D</td>
<td>PO</td>
<td>value</td>
<td>total_mass_evap</td>
</tr>
<tr>
<td>total_mass_evap</td>
<td>status</td>
<td>0</td>
<td>L</td>
<td>PO</td>
<td>status</td>
<td>total_mass_evap</td>
</tr>
<tr>
<td>total_mass_evap</td>
<td>ideal</td>
<td>0</td>
<td>D</td>
<td>PO</td>
<td>value</td>
<td>total_mass_evap</td>
</tr>
<tr>
<td># Process INPUTS #</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>evap_rate</td>
<td>sp</td>
<td>0</td>
<td>D</td>
<td>PI</td>
<td>value</td>
<td>evap_rate</td>
</tr>
<tr>
<td>evap_rate</td>
<td>status</td>
<td>0</td>
<td>L</td>
<td>PI</td>
<td>status</td>
<td>evap_rate</td>
</tr>
<tr>
<td>evap_rate</td>
<td>pv</td>
<td>0</td>
<td>D</td>
<td>PI</td>
<td>value</td>
<td>evap_rate</td>
</tr>
<tr>
<td>evap_rate</td>
<td>base</td>
<td>0</td>
<td>D</td>
<td>PI</td>
<td>value</td>
<td>evap_rate</td>
</tr>
<tr>
<td>evap_rate</td>
<td>operlow</td>
<td>-1</td>
<td>D</td>
<td>PI</td>
<td>value</td>
<td>evap_rate</td>
</tr>
<tr>
<td>evap_rate</td>
<td>operupp</td>
<td>100</td>
<td>D</td>
<td>PI</td>
<td>value</td>
<td>evap_rate</td>
</tr>
<tr>
<td>evap_rate</td>
<td>active</td>
<td>0</td>
<td>L</td>
<td>PI</td>
<td>status</td>
<td>evap_rate</td>
</tr>
<tr>
<td># INCA SIGNALS #</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Timer</td>
<td>SyncFrom</td>
<td>0</td>
<td>L</td>
<td>GEN</td>
<td>SYNC</td>
<td></td>
</tr>
<tr>
<td>IncaTest</td>
<td>SyncTo</td>
<td>0</td>
<td>L</td>
<td>GEN</td>
<td>SYNC</td>
<td></td>
</tr>
<tr>
<td>IncaTest</td>
<td>SyncFrom</td>
<td>0</td>
<td>L</td>
<td>GEN</td>
<td>SYNC</td>
<td></td>
</tr>
<tr>
<td>IncaAutTest</td>
<td>SyncFrom</td>
<td>0</td>
<td>L</td>
<td>GEN</td>
<td>SYNC</td>
<td>SyncFromIncaAutTest</td>
</tr>
<tr>
<td>IncaTest</td>
<td>StatusMsg</td>
<td>OK</td>
<td>S</td>
<td>GEN</td>
<td>IncrTest</td>
<td>IncrTest</td>
</tr>
<tr>
<td>IncaTest</td>
<td>StatusLvl</td>
<td>0</td>
<td>L</td>
<td>GEN</td>
<td>IncrTest</td>
<td>IncrTest</td>
</tr>
<tr>
<td>IncaTest</td>
<td>StatusAck</td>
<td>0</td>
<td>L</td>
<td>GEN</td>
<td>IncrTest</td>
<td>IncrTest</td>
</tr>
<tr>
<td>IncaTest</td>
<td>SampleTime</td>
<td>1</td>
<td>D</td>
<td>GEN</td>
<td>IncrTest</td>
<td>IncrTest</td>
</tr>
<tr>
<td>IncaTest</td>
<td>SampleNumber</td>
<td>0</td>
<td>L</td>
<td>GEN</td>
<td>IncrTest</td>
<td>IncrTest</td>
</tr>
<tr>
<td>IncaTest</td>
<td>MasterOnOff</td>
<td>0</td>
<td>L</td>
<td>GEN</td>
<td>IncrTest</td>
<td>IncrTest</td>
</tr>
<tr>
<td>IncaTest</td>
<td>shutdown</td>
<td>0</td>
<td>L</td>
<td>GEN</td>
<td>IncrTest</td>
<td>IncrTest</td>
</tr>
<tr>
<td>IncaEngine</td>
<td>SyncTo</td>
<td>0</td>
<td>L</td>
<td>GEN</td>
<td>SYNC</td>
<td></td>
</tr>
<tr>
<td>IncaEngine</td>
<td>SyncFrom</td>
<td>0</td>
<td>L</td>
<td>GEN</td>
<td>SYNC</td>
<td></td>
</tr>
</tbody>
</table>
### B.2 Scheduler.csv

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Ts</th>
<th>StartSync</th>
<th>EndSync</th>
<th>Over samp</th>
<th>Max Start</th>
<th>Max End</th>
<th>Level</th>
<th>Pre Start IDs</th>
<th>Pre Start Delays</th>
<th>After End IDs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>File2DataServer</td>
<td>60</td>
<td>SyncTo</td>
<td>File2DataServer.</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>10</td>
<td>[]</td>
<td>[]</td>
<td>[4]</td>
</tr>
<tr>
<td>2</td>
<td>Matlab</td>
<td>60</td>
<td>SyncTo</td>
<td>Matlab.SyncTo</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>3</td>
<td>GpromsPlant</td>
<td>60</td>
<td>SyncTo</td>
<td>GpromsPlant.SyncTo</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>4</td>
<td>IncaTest</td>
<td>60</td>
<td>SyncTo</td>
<td>IncaTest.SyncTo</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>[]</td>
<td>[]</td>
<td>[4]</td>
</tr>
<tr>
<td>5</td>
<td>IncaEngine</td>
<td>60</td>
<td>SyncTo</td>
<td>IncaEngine.SyncTo</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>[]</td>
<td>[]</td>
<td>[4]</td>
</tr>
</tbody>
</table>

59
**B.3 OPCCOM**

**PARAMETER**

<table>
<thead>
<tr>
<th>Plant</th>
<th>AS</th>
<th>MODEL Kemira_stirred_vessel_20_liter</th>
</tr>
</thead>
</table>

**VARIABLE**

| finished              | AS                      | LOGICAL                              |
| Ts                    | AS                      | REAL                                 |
| re_init               | AS                      | LOGICAL                              |
| Simulate              | AS                      | LOGICAL                              |
| VarInfo               | AS                      | LOGICAL                              |
| Jac                   | AS                      | LOGICAL                              |
| SaveState             | AS                      | INTEGER                              |
| Lin                   | AS                      | LOGICAL                              |

**SCHEDULE**

**SEQUENCE**

PAUSE signalid "open:IPCOS.server.1,localhost"
PAUSE signalid "sync:gPROMSplant.syncto"
Ts := 0;
#Task Flags:
Finished := FALSE;
Re_init := FALSE;
Simulate := FALSE;
SaveState := 0;
VarInfo := FALSE;
Jac := FALSE;
Lin := FALSE;

WHILE NOT (Finished) DO
SEQUENCE
PAUSE signalid "wait:900000"
GET signalid "getopc"
   Finished := "Finished";
   Re_init := "Re_init";
   Simulate := "Simulate";
   SaveState := "SaveState";
   VarInfo := "VarInfo";
   Jac := "Jac";
   Lin := "Lin";
END #get

IF (re_init) THEN
   RESTORE "state2TUmodnew6"
END
IF (simulate) THEN
  SEQUENCE
    GET signalid "getopc"

    # Receive input variables from the server:
    # MV's
    plant.evap_rate_kg_per_hr := "evap_rate.sp";

    # Receive simulation time from server
    Ts:="Ts";
  END #get

  CONTINUE FOR Ts
  SEND signalid "sendopc"

    "X10.PV" := Plant.prod.quantiles_all_pop(2);
    "X50.PV" := Plant.prod.quantiles_all_pop(5);
    "X90.PV" := Plant.prod.quantiles_all_pop(8);
    "total_mass_evap.pv" := plant.mass_vapour_produced;
    "evap_rate.pv" := plant.evap_rate_kg_per_hr;

  END #send
  END #sequence Simulate
END #if Simulate

IF (SaveState = 1) THEN
  SAVE STATE "stateTUmod"
END
IF (SaveState = 2) THEN
  SAVE "varAndStates"
END

IF (VarInfo) THEN
  SENDMATHINFO signalid "count"
END
IF (Jac) THEN
  SENDMATHINFO signalid "jacobian"
END
IF (Lin) THEN
  LINEARISE SIGNALID "TEST_LIN"

    input
      Plant.evap_rate_kg_per_hr;

    output
      Plant.prod.quantiles_all_pop(2);
      Plant.prod.quantiles_all_pop(5);
      Plant.prod.quantiles_all_pop(8);
      plant.mass_vapour_produced;
### B.4 Controller.csv

#### # Signal Section

<table>
<thead>
<tr>
<th>COMMENT</th>
<th>SIGNAL NUMBER</th>
<th>TAGNAME</th>
<th>ATTR</th>
<th>SIGNAL TYPE</th>
<th>INST #</th>
<th>DISPLAY NAME</th>
<th>DESCRIPTION</th>
<th>SIGNAL SUBTYPE</th>
<th>UNIT</th>
<th>OFF</th>
<th>SET</th>
</tr>
</thead>
<tbody>
<tr>
<td># CV's</td>
<td>S</td>
<td>1</td>
<td>X10</td>
<td>cv</td>
<td>1</td>
<td>X10</td>
<td>X10_CSD</td>
<td>Setpoint</td>
<td>-</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>2</td>
<td>X50</td>
<td>cv</td>
<td>2</td>
<td>X50</td>
<td>X50_CSD</td>
<td>Setpoint</td>
<td>-</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>3</td>
<td>X90</td>
<td>cv</td>
<td>3</td>
<td>X90</td>
<td>X90_CSD</td>
<td>Setpoint</td>
<td>-</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td># MV's</td>
<td>S</td>
<td>4</td>
<td>evap_rate</td>
<td>mv</td>
<td>1</td>
<td>evap_rate</td>
<td>evap_rate</td>
<td>-</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### # --- Input block

<table>
<thead>
<tr>
<th>SIGNAL</th>
<th>ATTR</th>
<th>Data server</th>
<th>Default value</th>
</tr>
</thead>
<tbody>
<tr>
<td>block</td>
<td>signal attribute</td>
<td>Data server tag</td>
<td>Default value</td>
</tr>
<tr>
<td># --- CVs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I</td>
<td>X10</td>
<td>PV</td>
<td>X10.pv</td>
</tr>
<tr>
<td>I</td>
<td>X50</td>
<td>PV</td>
<td>X50.pv</td>
</tr>
<tr>
<td>I</td>
<td>X90</td>
<td>PV</td>
<td>X90.pv</td>
</tr>
<tr>
<td>I</td>
<td>X10</td>
<td>status</td>
<td>X10.status</td>
</tr>
<tr>
<td>I</td>
<td>X50</td>
<td>status</td>
<td>X50.status</td>
</tr>
<tr>
<td>I</td>
<td>X90</td>
<td>status</td>
<td>X90.status</td>
</tr>
<tr>
<td>I</td>
<td>X10</td>
<td>ideal</td>
<td>X10.ideal</td>
</tr>
<tr>
<td>I</td>
<td>X50</td>
<td>ideal</td>
<td>X50.ideal</td>
</tr>
<tr>
<td>I</td>
<td>X90</td>
<td>ideal</td>
<td>X90.ideal</td>
</tr>
</tbody>
</table>

#### # --- MVs

<table>
<thead>
<tr>
<th>SIGNAL</th>
<th>ATTR</th>
<th>Data server</th>
<th>Default value</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>evap_rate</td>
<td>pv</td>
<td>evap_rate.pv</td>
</tr>
<tr>
<td>I</td>
<td>evap_rate</td>
<td>status</td>
<td>evap_rate.status</td>
</tr>
</tbody>
</table>

#### # --- DVs

#### # --- Output

<table>
<thead>
<tr>
<th>SIGNAL</th>
<th>ATTR</th>
<th>Data server</th>
<th>Default value</th>
</tr>
</thead>
<tbody>
<tr>
<td>O</td>
<td>evap_rate</td>
<td>sp</td>
<td>StatusOK</td>
</tr>
</tbody>
</table>

#### # --- Simulator settings (ts=60 sec)

<table>
<thead>
<tr>
<th>P</th>
<th>controlPort</th>
<th>1238</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>processHost</td>
<td>localhost</td>
</tr>
<tr>
<td>P</td>
<td>processPort</td>
<td>1234</td>
</tr>
</tbody>
</table>
## Tuning Section

### Tuning Information

<table>
<thead>
<tr>
<th>SIGNAL</th>
<th>ATTRIBUTE</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>CONTROL</td>
<td>REQUEST: Engine_ctrl.request</td>
</tr>
<tr>
<td>I</td>
<td>CONTROL</td>
<td>INIT: Engine_ctrl.init</td>
</tr>
<tr>
<td>T</td>
<td>CONTROL</td>
<td>STATUS_REASON: UNKNOWN</td>
</tr>
<tr>
<td>T</td>
<td>CONTROL</td>
<td>STATUS: 0</td>
</tr>
<tr>
<td>O</td>
<td>CONTROL</td>
<td>STATUS: ALWAYS: Engine_ctrl.status</td>
</tr>
<tr>
<td>T</td>
<td>CONTROL</td>
<td>SHUTDOWN: 0</td>
</tr>
<tr>
<td>T</td>
<td>CONTROL</td>
<td>NOSPECVAL: 1</td>
</tr>
<tr>
<td>T</td>
<td>DELTAMODE</td>
<td>DELTAMODEON: 0</td>
</tr>
<tr>
<td>T</td>
<td>DELTAMODE</td>
<td>TRAJECTON: 0</td>
</tr>
<tr>
<td>T</td>
<td>DELTAMODE</td>
<td>TRAJECTCOUNT: 0</td>
</tr>
<tr>
<td>T</td>
<td>DELTAMODE</td>
<td>TRAJECTSET: 0</td>
</tr>
<tr>
<td>T</td>
<td>DELTAMODE</td>
<td>NOTRAJECTSP: 0</td>
</tr>
<tr>
<td>T</td>
<td>MODEL</td>
<td>DEPMINGOOD: 1</td>
</tr>
<tr>
<td>T</td>
<td>MODEL</td>
<td>FUTHOR: 2</td>
</tr>
<tr>
<td>T</td>
<td>MODEL</td>
<td>HOTMODELINIT: 0</td>
</tr>
<tr>
<td>T</td>
<td>MODEL</td>
<td>INDEPMINGOOD: 1</td>
</tr>
<tr>
<td>T</td>
<td>MODEL</td>
<td>MAXPREDCYCLE: 5</td>
</tr>
<tr>
<td>T</td>
<td>MODEL</td>
<td>MODELCYCLE: 15</td>
</tr>
<tr>
<td>T</td>
<td>MODEL</td>
<td>NAMEMODEFILE: KemiraMod.mat</td>
</tr>
<tr>
<td>T</td>
<td>MODEL</td>
<td>MODINITREADY: 0</td>
</tr>
<tr>
<td>T</td>
<td>MODEL</td>
<td>MODINITSTARTED: 0</td>
</tr>
<tr>
<td>T</td>
<td>MODEL</td>
<td>NAMEMODEL: KemiraMod</td>
</tr>
<tr>
<td>T</td>
<td>MODEL</td>
<td>NEWMODEL: 0</td>
</tr>
<tr>
<td>T</td>
<td>MODEL</td>
<td>NAMEOBSERVERFILE: Observer.mat</td>
</tr>
<tr>
<td>T</td>
<td>MODEL</td>
<td>PASTHOR: 15</td>
</tr>
<tr>
<td>T</td>
<td>MODEL</td>
<td>STATEESTIMCREATED: 0</td>
</tr>
<tr>
<td>T</td>
<td>MODEL</td>
<td>STATEMAPCREATED: 0</td>
</tr>
<tr>
<td>T</td>
<td>SYNC</td>
<td>INT: 1</td>
</tr>
<tr>
<td>T</td>
<td>PREDICT</td>
<td>HORIZON: 900</td>
</tr>
<tr>
<td>T</td>
<td>DEBUG</td>
<td>TRACE: 0</td>
</tr>
<tr>
<td>T</td>
<td>DYNOPT</td>
<td>SVDTH: 1.00E-06</td>
</tr>
<tr>
<td>T</td>
<td>MVMINGOOD</td>
<td>INT: 0</td>
</tr>
<tr>
<td>T</td>
<td>CVMINGOOD</td>
<td>INT: 1</td>
</tr>
</tbody>
</table>

--- MV 1 [1/s]

<table>
<thead>
<tr>
<th>SIGNAL</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>evap_rate VALLOW: 0</td>
</tr>
<tr>
<td>T</td>
<td>evap_rate ENGLOW: 0</td>
</tr>
<tr>
<td>T</td>
<td>evap_rate OPERLOW: 0</td>
</tr>
<tr>
<td>T</td>
<td>evap_rate IDEAL: 12.92</td>
</tr>
<tr>
<td>T</td>
<td>evap_rate OPERUPP: 15</td>
</tr>
<tr>
<td>T</td>
<td>evap_rate ENGUPP: 20</td>
</tr>
<tr>
<td>T</td>
<td>evap_rate VALUPP: 20</td>
</tr>
<tr>
<td>T</td>
<td>evap_rate USE: 1</td>
</tr>
<tr>
<td>T</td>
<td>evap_rate CRITICAL: 0</td>
</tr>
<tr>
<td>T</td>
<td>evap_rate NORMSTEP: 5</td>
</tr>
<tr>
<td>T</td>
<td>evap_rate SSMAXSTEP: 3</td>
</tr>
<tr>
<td>T</td>
<td>evap_rate ISSZONEWT: 5</td>
</tr>
</tbody>
</table>
evap_rate  IDEALRANK  50
evap_rate  ISSIDEALWT  10
evap_rate  MAXSTEP  0.5
evap_rate  MVPENALTY  8
evap_rate  MOVETIMES  1  2
evap_rate  TRAJECT  0
evap_rate  GAIN  1
evap_rate  DELAY  0
evap_rate  RAMPINGON  0
evap_rate  RAMPLIMUPP  0
evap_rate  RAMPLIMLOW  0
evap_rate  IDLTRACKINGON  0
evap_rate  INTERNALIDEAL  0
evap_rate  MODELCYCLE  5

# --- CV 1 [um]
X10  VALLOW  -100
X10  ENGLow  -100
X10  OPERLOW  -5
X10  IDEAL  205
X10  OPERUPP  450
X10  ENGUPP  500
X10  VALUPP  500
X10  USE  0
X10  CRITICAL  0
X10  SSMAXSTEP  20
X10  ZONERANK  100
X10  ISSZONEWT  10
X10  IDEALRANK  10
X10  ISSIDEALWT  9
X10  IDYNWT  100
X10  TRAJECT  0
X10  GAIN  1
X10  DELAY  0
X10  RAMPINGON  0
X10  RAMPLIMUPP  0
X10  RAMPLIMLOW  0
X10  IDLTRACKINGON  0
X10  INTERNALIDEAL  0
X10  MODELCYCLE  5
X10  INTERMITON  0
X10  NOTMEASON  0
X10  NEWPV  0

# --- CV 2 [um]
X50  VALLOW  -100
X50  ENGLow  -100
X50  OPERLOW  -5
X50  IDEAL  637.5569
X50  OPERUPP  1200
X50  ENGUPP  1400
X50  VALUPP  1400
T X50 USE 1
T X50 CRITICAL 0
T X50 SSMAKSTEP 20
T X50 ZONERANK 100
T X50 ISSZONEWT 10
T X50 IDEALRANK 10
T X50 ISSIDEALWT 9
T X50 IDYNWT 100
T X50 TRAJECT 0
T X50 GAIN 1
T X50 DELAY 0
T X50 RAMPINGON 0
T X50 RAMPLIMUPP 0
T X50 RAMPLIMLOW 0
T X50 IDLTRACKINGON 0
T X50 INTERNALIDEAL 0
T X50 MODEL_CYCLE 5
T X50 INTERMITON 0
T X50 NOTMEASON 0
T X50 NEWPV 0

# --- CV 3 [um]
T X90 VALLOW -100
T X90 ENGLow -100
T X90 OPERRLOW -5
T X90 IDEAL 1243.3
T X90 OPERUPP 1500
T X90 ENGUPP 1700
T X90 VALUPP 1700
T X90 USE 0
T X90 CRITICAL 0
T X90 SSMAKSTEP 20
T X90 ZONERANK 100
T X90 ISSZONEWT 10
T X90 IDEALRANK 10
T X90 ISSIDEALWT 9
T X90 IDYNWT 100
T X90 TRAJECT 0
T X90 GAIN 1
T X90 DELAY 0
T X90 RAMPINGON 0
T X90 RAMPLIMUPP 0
T X90 RAMPLIMLOW 0
T X90 IDLTRACKINGON 0
T X90 INTERNALIDEAL 0
T X90 MODEL_CYCLE 5
T X90 INTERMITON 0
T X90 NOTMEASON 0
T X90 NEWPV 0

65