COORDINATED PRODUCTION
FOR BETTER RESOURCE EFFICIENCY

D1.1 Efficient construction of efficient models

Maarten Nauta
PSE – United Kingdom

May 2019
www.spire2030.eu/copro

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 723575

Project Details

PROJECT TITLE	Improved energy and resource efficiency by better coordination of production in the process industries
PROJECT ACRONYM	**COPRO**
GRANT AGREEMENT NO	**723575**
INSTRUMENT	**RESEARCH AND INNOVATION ACTION**
CALL	**H2020-SPIRE-02-2016**
STARTING DATE OF PROJECT	**NOVEMBER, 1ST 2016**
PROJECT DURATION	**42 MONTHS**
PROJECT COORDINATOR (ORGANIZATION)	**PROF. SEBASTIAN ENGELL (TUDO)**

THE COPRO PROJECT

The goal of CoPro is to develop and to demonstrate methods and tools for process monitoring and optimal dynamic planning, scheduling and control of plants, industrial sites and clusters under dynamic market conditions. CoPro pays special attention to the role of operators and managers in plant-wide control solutions and to the deployment of advanced solutions in industrial sites with a heterogeneous IT environment. As the effort required for the development and maintenance of accurate plant models is the bottleneck for the development and long-term operation of advanced control and scheduling solutions, CoPro will develop methods for efficient modelling and for model quality monitoring and model adaption.

The CoPro Consortium

Participant No	Participant organisation name	Country	Organisation
1 (Coordinator)	Technische Universität Dortmund (TUDO)	DE	HES
2	INEOS Manufacturing Deutschland GmbH (INEOS)	DE	IND
3	Covestro Deutschland AG (COV)	DE	IND
4	Procter & Gamble Services Company NV (P&G)	BE	IND
5	Lenzing Aktiengesellschaft (LENZING)	AU	IND
6	Frinsa del Noroeste S.A. (Frinsa)	ES	IND
7	Universidad de Valladolid (UVA)	ES	HES
8	École Polytechnique Féderale de Lausanne (EPFL)	CH	HES
9	Ethniko Kentro Erevnas Kai Technologikis Anaptyxis (CERTH)	GR	RES
10	IIM-CSIC (CSIC)	ES	RES
11	LeiKon GmbH (LEIKON)	DE	SME
12	Process Systems Enterprise LTD (PSE)	UK	SME

13	Divis Intelligent Solutions GmbH (divis)	DE	SME
14	Argent & Waugh Ltd. (Sabisu)	UK	SME
15	ASM Soft S.L (ASM)	ES	SME
16	ORSOFT GmbH (ORS)	DE	SME
17	Inno TSD (inno)	FR	SME

Document details

DELIVERABLE TYPE	**REPORT**
DELIVERABLE NO	**1.1**
DELIVERABLE TITLE	**REPORT ON EFFICIENT CONSTRUCTION OF EFFICIENT MODELS**
NAME OF LEAD PARTNER FOR THIS DELIVERABLE	**PSE**
VERSION	**0.3**
CONTRACTUAL DELIVERY DATE	**30 APRIL 2019**
ACTUAL DELIVERY DATE	**21 MAY 2019**

Dissemination level		
PU	Public	X
CO	Confidential, only for members of the consortium (including the Commission)	

Abstract

As part of the COPRO project, models for industrial processes are developed and used for optimization of the processes. These models need to constructed in an efficient manner and they need to be sufficiently predictive and accurate for the purpose that they are used for. This report gives an overview of the modelling work performed for the use cases of the COPRO project. On a more general level, this report also discusses recommendations for efficient modelling. This includes considerations on which modelling approach to choose for particular problems. One promising direction for obtaining predictive model in an efficient manner is hybrid modelling. UVA and PSE have investigated hybrid modelling application to chemical process models. PSE has developed a prototype hybrid modelling tool in the framework of this project.

REVISION HISTORY

The following table describes the main changes done in the document since it was created.

Revision	Date	Description	Author (Organisation)
V0.1	02/02/2019	Initial draft by PSE.	K.M. Nauta (PSE)
V0.2	28/03/2019	Extended with emailed contributions from UVA, Divis, TUDO, Lenzing, CSIC, Divis, INEOS.	K.M. Nauta (PSE)
V0.3	08/04/2019	Corrections from CSIC, .UVA	K.M. Nauta (PSE)
V0.3	21/05/2019	Final approval	S. Engell (TUDO)

Disclaimer

Table of contents

1 Executive summary

As part of the COPRO project models for industrial processes are developed, extended and used for optimization of the processes. Having an accurate model is a pre-requisite to be able to predict and optimise industrial processes. Developing and validating any model is commonly both an effort and time intensive process. The associated financial investment has to be made before actually getting any return from the optimisation of the process. Moreover, in many cases (part of) this investment has to be made before actually being able to identify any potential for process improvement for a particular process. Therefore, in a substantial number cases, the requirement to develop an accurate model can lead to the process improvement project being perceived of not having sufficient Return-On-Investment (ROI) or having too much risk. This is the so-called “modelling bottleneck”. The ways it can be alleviated is the topic of this report. One way to reduce this bottleneck is to make sure that the model development and validation step is conducted in a manner that is efficient as possible.

This report for the D1.1 work package first describes how the models in the COPRO project were generated, and which decisions were taken to ensure the modelling process was efficient. Next, the work by UVA and PSE on “hybrid modelling” is presented. This is a modelling technique that has the potential to combine the advantages of data-driven and first-principles modelling, which would lead to a modelling approach that generates models that can be generated efficiently but are predictive outside of the range of the data that was used to generate them. Finally, conclusions and recommendations are presented regarding efficient modelling in general and hybrid modelling in particular.

2 Introduction

2.1 Steps involved in deriving a process model

To identify how models can be generated efficiently first an overview if given of the steps required to derive a model. For each of these steps the (recommended) general approach is described and an indication is given of the effort and the pre-requisites for this step. While these steps are presented as being sequential, in practise some iterations are required between these steps.

2.1.1 Defining the purpose of the model

Models are typically generated with an explicit purpose. In the COPRO project this purpose is to improve energy and resource efficiency by operating processes in a more efficient manner. Therefore, for this project, this effectively means that any model are (ultimately) used as part of an optimization problem formulation. Formulating this optimization problem ahead of any modeling activity can inform:

1. The input – output structure of the model. Which quantities in the process should the model relate to eachother? Which quantities can be manipulated to improve process operation? Which quantities affect the KPI's of the process (e.g. Resource Efficiency Indicators, process economics)?
2. The (expected) ranges of the variables, both input and outputs.
3. The assumptions used for the operation of the process.
4. The desired approximate evaluation time of the model.

All of these aspects, related to the optimization problem are important to inform the next steps of the modelling process. For the models developed as part of the COPRO projects and described in this report the purpose is as following:

- Sterilization plant (FRINSA use case, CSIC/FRINSA/UVA): Optimization of the production schedule for the sterilization plant and online advisory systems for food safety.
- Evaporator network (Lenzing use case/UVA): Optimization of the evaporator load allocation of the evaporator network. The goal of the optimization is to minimize the overall steam consumption of the evaporation network by an efficient load allocation for each evaporator
- Ammonia network (INEOS use case, INEOS/TUDO): Optimisation of the operation of the tanks and of the plants involved in the ammonia network.
- Best-Demonstrated Practises approach (INEOS use case, INEOS): Advisory system for plant performance improvement.
- Naphtha Cracker (INEOS use case, Divis/INEOS): Prediction of cracker coking in order to advise on optimal de-coking schedule.
- Olefins plant (PSE): Optimisation of the operation of the olefins plant giving a state of coking of each furnace.

Most of these uses of the models involve solving a mathematically well-defined optimization problem within the COPRO project, with the exception the Naphtha Cracker coking prediction and the BDP approach. The latter relies on operator expertise to improve operation.

2.1.2 Gathering data / performing experiments

For the modelling studies conducted as part of the COPRO project data was obtained in the following ways:

1. Data from historical plant operation obtained from the plant historian (Evaporator and surface condenser case studies (Lenzing/UVA), Ammonia network case study (INEOS/TUDO), Naphtha Cracker (Divis/INEOS), Best-Demonstrated Practises approach (INEOS))
2. Measurements from experiments conducted explicitly for the purpose of modelling (Sterilizer case study (FRINSA use case), Granulation soft-sensing case study (PSE))
3. Data generated from evaluations of a high-fidelity first-principles model (Olefins plant case study (PSE))

In the first two cases, measurement data is used directly while in the last case measurement data is only used implicitly in the sense that it is assumed that the high-fidelity model was validated using measurement data where required.

For the first option, no effort and resources have to be spent on experimentation. For industrial processes that are well-instrumented and have an IT infrastructure (PI system, historian) obtaining historical data itself can be simple the effort may be in the order of days. However, it is typically not trivial to determine which data needs to be selected and whether measurements can be relied on.

In the second case, when effort/resources have to be spent on experimentation, this effort is highly process specific and its difficult to generalize between different applications / plants / industries.

For the last option, given a validated high-fidelity first-principles model and given the assumption that the first-principles model is valid for the operation range relevant for the purpose of the model, the only effort required is CPU resources. However, even for high-fidelity first-principles models typically a validation stage to the actual process is employed, something that was not done within the context of this project due to confidentiality of kinetic data for Naphtha crackers.

2.1.3 Data pre-processing / Data-Exploratory analysis

An important step in both first-principles modelling and data-driven modelling is exploratory analysis and pre-processing of the data. These pre-processing steps often involve human analysis and decisions and can account for 90% of the time spent on modelling, for cases where historical data is used. During this step, the data is analysed using statistical indicators used for aggregation (mean, variance, correlation measures) and well as using visual inspection using time plots, scatter plots, histograms, correlation plots and other visualisation tools.

The exploratory analysis and the data pre-processing are often done together. Exploratory analysis typically relates to assessing whether the data quality is adequate, whether data is available to model specific parts of the process, which parts of the data relate to normal and abnormal operation. Typically, data-pre-processing steps are data selection, scaling, outlier detection and filtering.

2.1.4 Defining the model structure

After the data analysis, a promising modelling structure can be identified. When first-principles modelling is used an option also is to identify the modelling structure and initial guesses for the parameters a-priori and then use these to inform the design of experiments. In certain machine learning tools (e.g. (Divis Intelligent Solutions GmbH, 2018)) this selection of a structure is performed

automatically as part of the model validation step (see next section) in an approach where candidate structures are selected based on their performance during model validation.

This step is typically not as effort intensive as the data preprocessing step but might require modelling expertise, in particular when parametric models / first-principle models are used and the equations need to be selected/derived based on modelling assumptions and correlations need to be selected from literature sources.

2.1.5 Model validation

During this step the data is used to validate the model, fit model parameters and, for non-parametric models, choose a model structure. This step is discussed in more detail in a separate COPRO deliverable, (K.M.Nauta, COPRO project partners, 2018). Software tools generally provide defined workflows for this step, and when the data has been pre-processed properly the effort is typically reasonable.

3 Modelling work performed for COPRO use cases

3.1 Overview

During the COPRO project modelling work was performed for each use case. This chapter gives an overview of this work, presents the resulting models and

3.2 Sterilizer modelling (FRINSA use case)

3.2.1 Description of the system being modelled

Sterilization process is one of the main bottlenecks in FRINSA plant. The objective of sterilization is to kill harmful microorganisms by the action of heat. To that purpose, the food product is introduced in cans which are sealed. Then, cans are introduced in carts and the carts are introduced into the sterilizer (or autoclave), see Figure 1. The bottom section of the autoclave is filled with water; such water is pumped to a plate heat exchanger (PHE) where it is heated. Heated water is introduced in the autoclave using a large number of showers distributed along the retort. Once heated water reaches a given predefined temperature (typically, in the range of 110-125 ℃) it is kept during a given amount of time. The heat exchanger uses steam to heat the autoclave water. Once the product is save, steam circuit is closed and cold water is introduced in the PHE in order to cool the system down. When the cans reach a given temperature the batch is finished and the carts are removed from the autoclave.

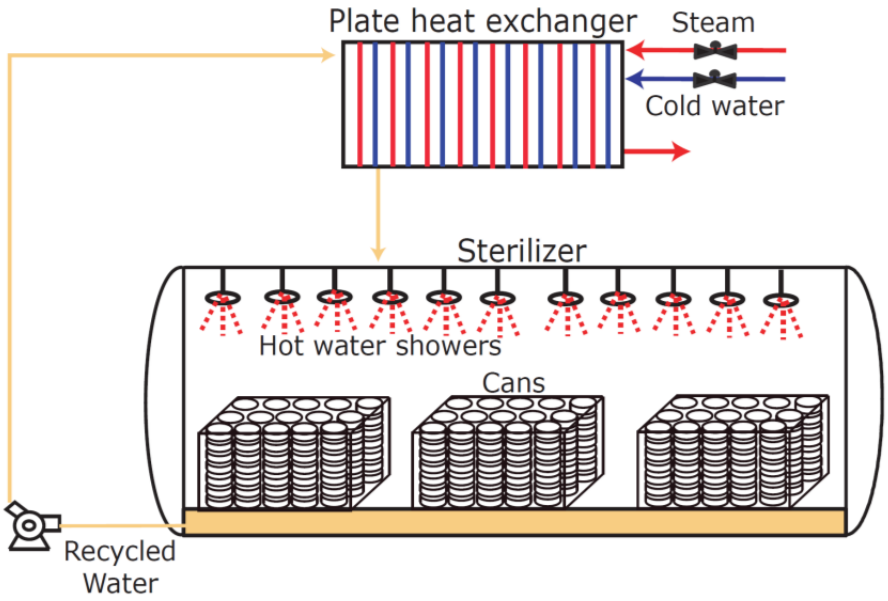


Figure 1. General scheme of the sterilization process.

Sterilization has, however, two drawbacks: (i) product quality is reduced by the action of heat; (ii) it is an energy and time demanding process.

Different models, presented in the following section, are employed to describe the different processes occurring during a sterilization batch. Such models are coupled and they will describe: (i) evolution of microbial lethality (safety parameter); (ii) evolution of can surface color (quality parameter); (iii) temperature distribution and evolution in the food product; (iv) temperature evolution in the sterilizer; (v) temperature evolution in the PHE; (vi) energy consumption.

3.2.2 Types of models used in the FRINSA case study

In this section, models mentioned in the previous section will be briefly described.

3.2.2.1 Safety and quality parameters

Lethality of microorganisms (F_0) is computed using the temperature at the coldest point of the food product (T_c), which is usually located at the center of the product. If the coldest point satisfies the safety constraint, the product is save. Time evolution of F_0 has been widely studied (Abril., 2003), and it can be described by an Ordinary Differential Equation (ODE). In particular F_0 is described by a Thermal Death Time (TDT) kinetic equation of the form (Abril., 2003):

$$\frac{dF_0}{dt} = 10^{\frac{T_c - T_{ref}}{z_{F,ref}}} \quad (1)$$

where $T_{ref}, z_{F,ref}$ are given parameters whose value depend on the bacteria considered.

At the same time, temperature affects food quality (surface color). Different ODE models can be used for describing color degradation in food products. In this project, we have tested: TDT kinetic models (J.R. Banga, 1993); first and second order models with Arrhenius-type temperature dependence; and logistic models (H.S. Ramaswamy, 2002). We concluded that TDT kinetic model of the form:

$$\frac{dC}{dt} = -\frac{1}{D_{ref}} 10^{\frac{T_s(t) - T_{ref}}{z_{C,ref}}} \quad (2)$$

is the best one for describing color evolution of tuna fish during sterilization. In Eq. (2), $T_s(t)$ is the temperature at the top surface of the food product whereas D_{ref} and $z_{C,ref}$ were estimated from experimental data for a given T_{ref}.

Note that, in order to compute the safety and quality parameters, food product temperature evolution and spatial distribution are required (see Section 3.2.2.2).

3.2.2.2 Can temperature distribution

A picture of the real can is presented in Figure 2. If the can is homogeneously heated (as it is usually the case) then the 3D problem can be reduced to a 2D problem in cylindrical coordinates (z, r). Furthermore, symmetry allows us considering just half of the whole spatial domain (shaded region in the Figure).

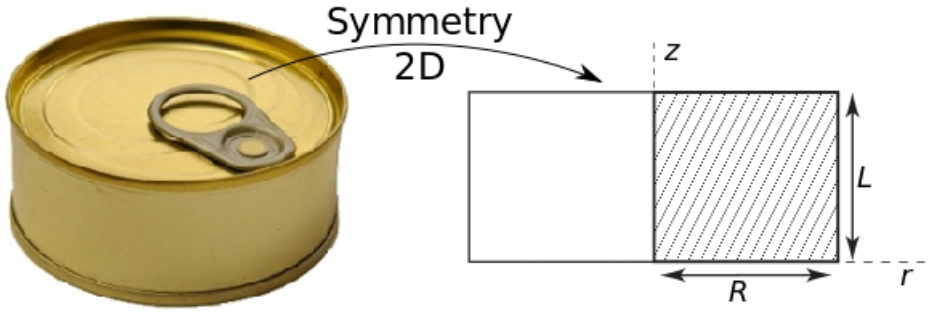


Figure 2. Picture of a real (3D) cylindrical can (left). Symmetry properties allow reducing the dimensionality to a 2D problem (right).

The heat equation is used to describe the evolution of the temperature inside the can. The model consists, therefore, of a Partial Differential Equation (PDE) of the form:

$$\frac{\partial T}{\partial t} = \alpha\left(\frac{\partial^2 T}{\partial z^2} + \frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial T}{\partial r}\right)\right) \tag{3}$$

where α corresponds with the thermal diffusivity of the packaged food (in this case tuna) whereas z, r are, respectively, the height and the radius of the product.

At the product top and bottom, a layer of filling fluid separates the metal cover from the product so heat flux boundary conditions are considered:

$$n \cdot \kappa \nabla \mathrm{T}|_{z=L} = h_t(T|_{z=L} - T_R) \quad \forall r \tag{4}$$

$$n \cdot \kappa \nabla \mathrm{T}|_{z=0} = h_b(T|_{z=0} - T_R) \quad \forall r \tag{5}$$

T_R denotes the temperature of the autoclave (sterilization temperature). At the right boundary the metal cover is in contact with the product so the transfer coefficient will be large and Dirichlet boundary conditions are considered:

$$\mathrm{T}|_{z=0} = \mathrm{T}|_{r=R} = T_R$$

These boundary conditions can be approximated by Robin boundary conditions of the form:

$$n \cdot \alpha \nabla \mathrm{T}|_{r=R} = h_r^*(T|_{r=R} - T_R) \quad \forall z \tag{6}$$

where h_r^* is a large value.

At the left boundary, symmetry boundary conditions are considered:

$$n \cdot \kappa \nabla \mathrm{T}|_{r=0} = 0 \quad \forall z \tag{7}$$

Classical methods for solving PDE systems of the form of (3)-(7) are, in general, computationally demanding. A reduced order model (ROM) has been derived to alleviate such burden. Details about the ROM are given in Deliverable 1.4.

Parameters α, h_t and h_b were estimated from experimental data.

3.2.2.3 Autoclave model

A simple energy balance allows us obtaining the equation describing the evolution of the sterilizer water temperature.

$$m_{W,R} c_{p,W} \frac{dT_S}{dt} = q_R c_{p,W}\left(T_{w,PHE} - T_S\right) - \left(Q_{can} + Q_{amb} + Q_R\right) \tag{8}$$

where $T_{w,PHE}$ is the water temperature at the output of the PHE (input of the autoclave). $c_{p,W}$ is the water specific heat and q_R is the recycled water flow. Q_{can} is the heat absorbed by the cans, which is computed as :

$$Q_{can} = n_c m_c c_{p,c} \frac{dT_{p,n}}{dt}$$

where n_c, m_c and $c_{p,c}$ are the number of cans in the sterilizer, the mass of each can and the specific heat of the can, respectively. $T_{p,n}$ is the mean temperature in the can, which is computed as (C. Vilas M. G., 51-65):

$$T_{p,n} = \frac{\int_0^R \int_0^L T_p(r,z,t) r \, dz dr}{\int_0^R \int_0^L r \, dz dr}$$

Sterilizers heat loses are computed as:

$$Q_{amb} = h_c A_c (T_S - T_{amb}) + \theta \cdot e_r \cdot A_c \left(T_S^4 - T_{amb}^4\right)$$

where T_{amb}, h_c, A_c, θ and e_r are the surrounding media temperature, heat transfer coefficient between the sterilizer and the surrounding media, area of the retort metal cover, Stefan-Boltzmann constant and thermal emissivity. Finally, heat absorbed by the retort metal cover (Q_R) is computed as:

$$Q_R = m_R c_{p,R} \frac{dT_S}{dt}$$

where m_R and $c_{p,R}$ represent the mass and the specific heat of the retort metal cover.

3.2.2.4 Plate heat exchanger model

As shown in Figure 1, PHE contains two different type of plates (plates containing the heating steam and plates containing the autoclave water to be heated). Flow through the steam valve is computed as (A.A. Alonso, 2013):

$$q_A = 3.4 \times 10^{-8} \cdot Cf \cdot Cv \cdot u \cdot P_1 (y - 0.148 y^3); \qquad y = \frac{1.63}{C_f} \sqrt{\frac{P_1 - P_2}{P_1}}$$

Where P_1 and P_2 are the pressures in Pa before and after the valve, respectively. $C_f = 0.9$ is a characteristic valve parameter, $C_v = 12.3$ is the valve size whereas u is the valve opening (between 0 and 1). Flow through the plates containing the autoclave water is constant, $q_R = 25\ kg/s$.

In order to simplify the model, instead of taking into account all the plates separately, we will consider one plate for liquid water and one plate for steam. The contact area will correspond with the contact area of all plates.

Let us denote with sub index "A" those plates containing the steam, i.e those used to heat the retort water. Also, each plate will be divided into a number of compartments ($i = 1,2, \dots, n_c$). Mass balance for water and steam in plates "A", compartment i, results into:

$$\frac{dm_{s,A}^i}{dt} = q_A \left(x_{s,A}^{i-1} - x_{s,A}^i\right) - \psi^i \tag{9}$$

$$\frac{dm_{w,A}^i}{dt} = q_A \left(x_{w,A}^{i-1} - x_{w,A}^i\right) + \psi^i \tag{10}$$

where $x_{s,A}^i, x_{w,A}^i$ are the steam and water mass fractions in compartment i, with $x_{s,A}^0 = 1$ being the steam mass fraction in the flow coming from the boiler. ψ^i represents the condensation/evaporation flow in compartment i.

Energy balance in plates "A", compartment i, results into:

$$\begin{aligned} \left[m_{s,A}^i \left(c_{p,s,A}^i - R_s\right) + m_{w,A} c_{p,w}\right] \frac{dT_A^i}{dt} \\ = q_A x_{s,A}^{i-1} \Delta H_s^i + q_A x_{w,A}^{i-1} c_{p,w} \left(T_A^{i-1} - T_A^i\right) + q_A R_s T_A^i \left(x_{s,A}^{i-1} - x_{s,A}^i\right) \\ + UA \left(T_B^i - T_A^i\right) + \lambda \psi^i \end{aligned} \tag{11}$$

Where the different c_p represent specific heats, ΔH_s^i is the Enthalpy difference between compartment i and $i-1$. R_s is the ideal gas constant divided by the water molecular weight. U is the heat transfer coefficient between plates whereas A is the contact area. Finally, λ is the condensation heat for steam.

Energy balance in the plates containing the water to be heated (water recirculated from the autoclave), denoted by "B", results into:

$$m_{W,B}c_{p,w}\frac{dT_B^i}{dt} = q_R c_{p,w}\left(T_B^{i+1} - T_A^i\right) + UA\left(T_A^i - T_B^i\right) \quad (12)$$

With $T_B^{n_c+1} = T_S$. Note also that $T_{w,PHE}$ in Eq. (8) is the temperature of the first compartment, i.e. $T_{w,PHE} = T_B^1$.

3.2.2.5 Energy consumption

Energy consumption is computed as the amount of steam consumed, i.e.:

$$E = \int_0^{tf} q_A dt \quad (13)$$

3.2.3 Fitting results

3.2.3.1 Color dynamics

Parameters D_{ref} and $z_{C,ref}$ were estimated using experimental data. Five experiments were considered. Four of them were used for parameter estimation whereas the fifth one was used for validation purposes. Each experiment was performed as follows:

- The can was filled with the food product following the FRINSA plant protocol.
- Before the can was sealed, Minolta Chroma Meter CR-200 (J.R. Banga, 1993) was used to measure color (L, a, b). Five measurements at different points of the food product surface were taken.
- The can was filled with the filling liquid and sealed. Then, a sensor was introduced to measure the food product surface temperature.
- The can (including the sensor) was introduced in an autoclave were it was subjected to a given sterilization profile.
- At the end of the process, the can was opened and the filling liquid was drained. Then, color was measured again at five different points in the surface.

Three dimensional color values (L, a, b) where transformed to one dimensional total color difference. To that purpose, the darkest measurement taken in all experiments was used as the reference value (H.S. Ramaswamy, 2002). Then the total color difference was computed as the Euclid distance between the actual value and the reference:

$$C = \sqrt{\left(L - L_{ref}\right)^2 + \left(a - a_{ref}\right)^2 + \left(b - b_{ref}\right)^2} \quad (14)$$

Figure 3 shows the comparison between the simulation and the experimental results for both the estimation (Figure 3 (a)-(d)) and the validation (Figure 3 (e)) experiments. Black line corresponds with can surface temperature whereas blue line corresponds with surface color evolution. Blue dots represent the color experimental measurements at the beginning and at the end of the process. As shown in the figure, the model is able to reproduce the experimental results.

A more quantitative measure of the good of fitness is the RMSE, which, for the estimation experiments is RMSE = 0.283, whereas for the validation experiment is RMSE=0.064.

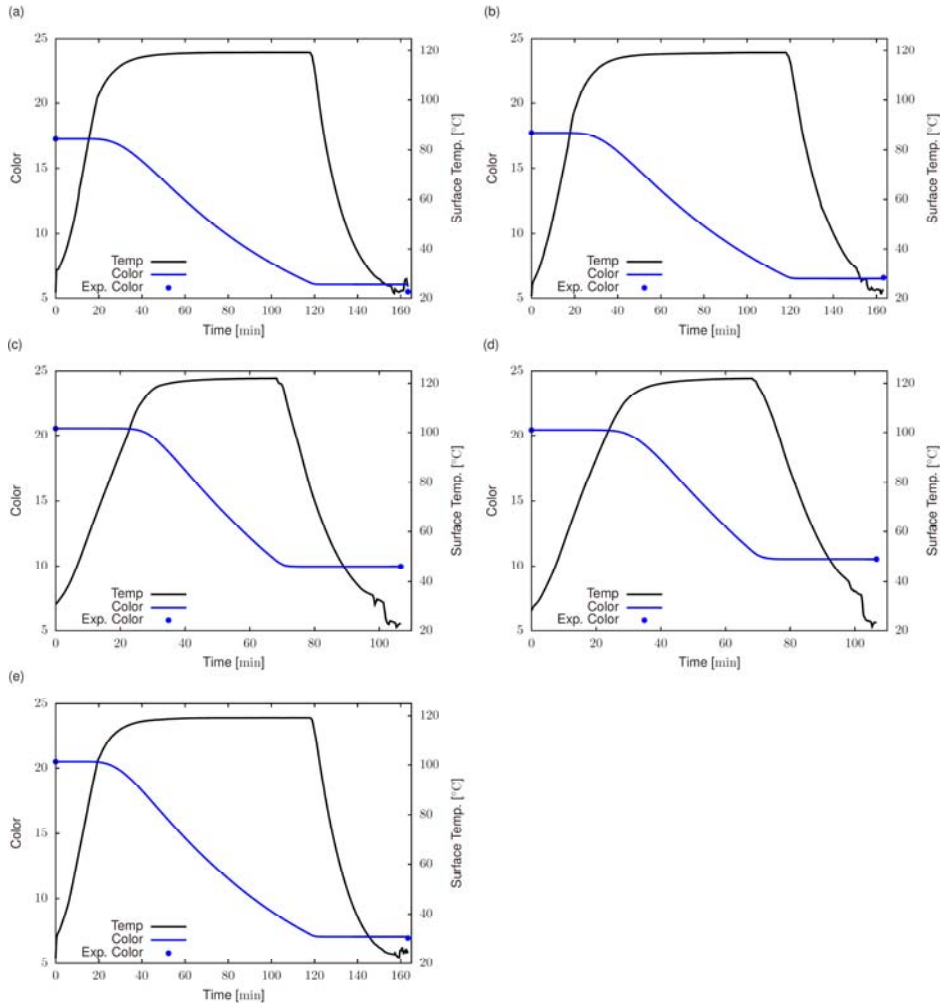


Figure 3. Comparison between model simulation (blue continuous line) and experimental data (blue dots) for color, under different processing conditions (black continuous line). (a)-(d) Estimation experiments and (e) validation experiment.

3.2.3.2 Can temperature distribution

In this case, parameters α, h_t and h_b are estimated from experimental data. Following the procedure described in (C. Vilas A. A.-M.-C., 2018), one can show that measurements at three different positions are required to ensure structural identifiability. Experimental data provided by FRINSA usually consists of the temperature evolution at one point inside the can (the center of the can). However, for cans RO200, using water as filling liquid, measurements at three points (top, center and bottom) are available. Therefore, we have estimated α, h_t and h_b for this type of can.

For the remaining cans (RO200 with oil as filling liquid and RO80 with both water and oil as filling liquid) we only have experimental data at the center of the can. Therefore, we assumed the same α and h_b as in the previous case since the filling liquid layer at the bottom is usually thin. Parameter h_t for these cans was computed using these experiments.

For each type of can we used four experiments for estimation purposes and two for validation. Figure 4 shows one of the validation experiments for each type of can. As shown in the figure, results are satisfactory.

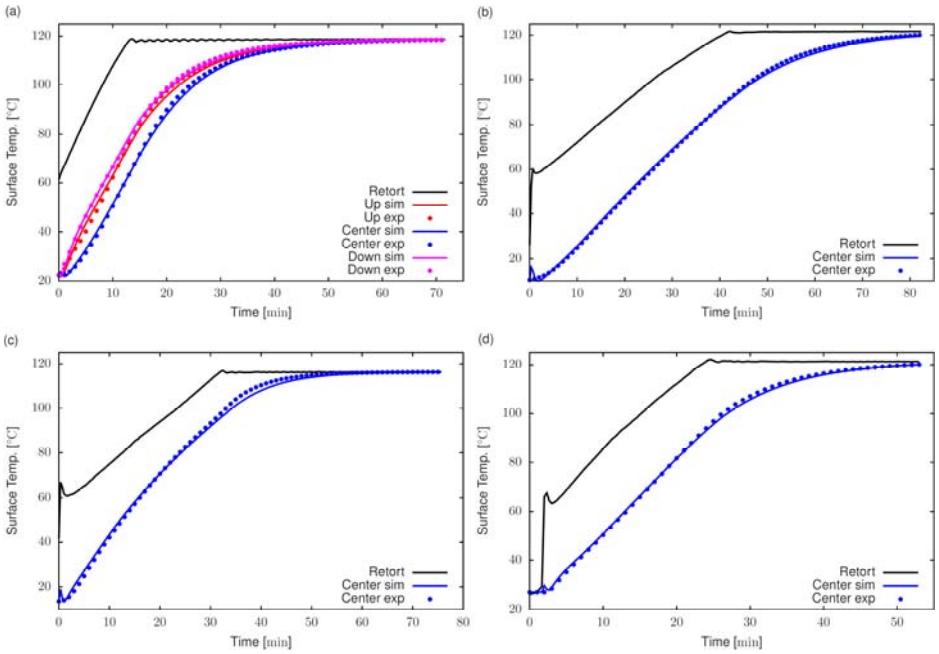


Figure 4. Validation results for the can temperature distribution model. Continuous blue lines correspond with the model results whereas blue dots represent the experimental data. Black continuous lines correspond with the measured autoclave temperature. (a) RO200 can with water as filling liquid ; (b) RO200 can with oil as filling liquid ; (c) RO80 can with water as filling liquid ; and (d) RO80 can with oil as filling liquid.

Table 1 summarizes the RMSE values obtained in each case.

Table 1. RMSE values for the temperature distribution model inside the cans during steriliization. Two types of cans are considered RO200 and RO80. Filling liquid can be either water or oil. RMSE values are presented separately for the estimation experiments (4 per type of can and filling liquid) and for the validation experiments (2 per type of can and filling liquid).

	RMSE Values			
	Estimation		Validation	
	Water	Oil	Water	Oil
RO200	1.262	0.806	1.309	1.095
RO80	1.268	0.736	2.089	1.090

3.2.4 Models in the CoPro project

These models will be used in the CoPro project, together with the scheduling techniques developed by UVa, for Scheduling in the FRINSA plant. Also, they will play a central role in the derivation of software sensors that allow for online assessment and prediction of safety/quality food parameters (see deliverable 2.4 for details).

3.3 Evaporator modelling (Lenzing use case)

3.3.1 Description of the system being modelled

The construction of efficient models focuses in the Lenzing use case mostly on the evaporator units used to re-concentrate the spinbath liquid after the spinning process. The entire evaporation plant consists of 29 different evaporators. Also the evaporators vary in terms of capacity, size and cooling source the mode of operation is the same. In Figure 5 a simplified scheme of an evaporator equipped with a cooling tower as cooling unit is depicted. Some evaporators use surface condensers with river water as cooling unit instead of cooling towers.

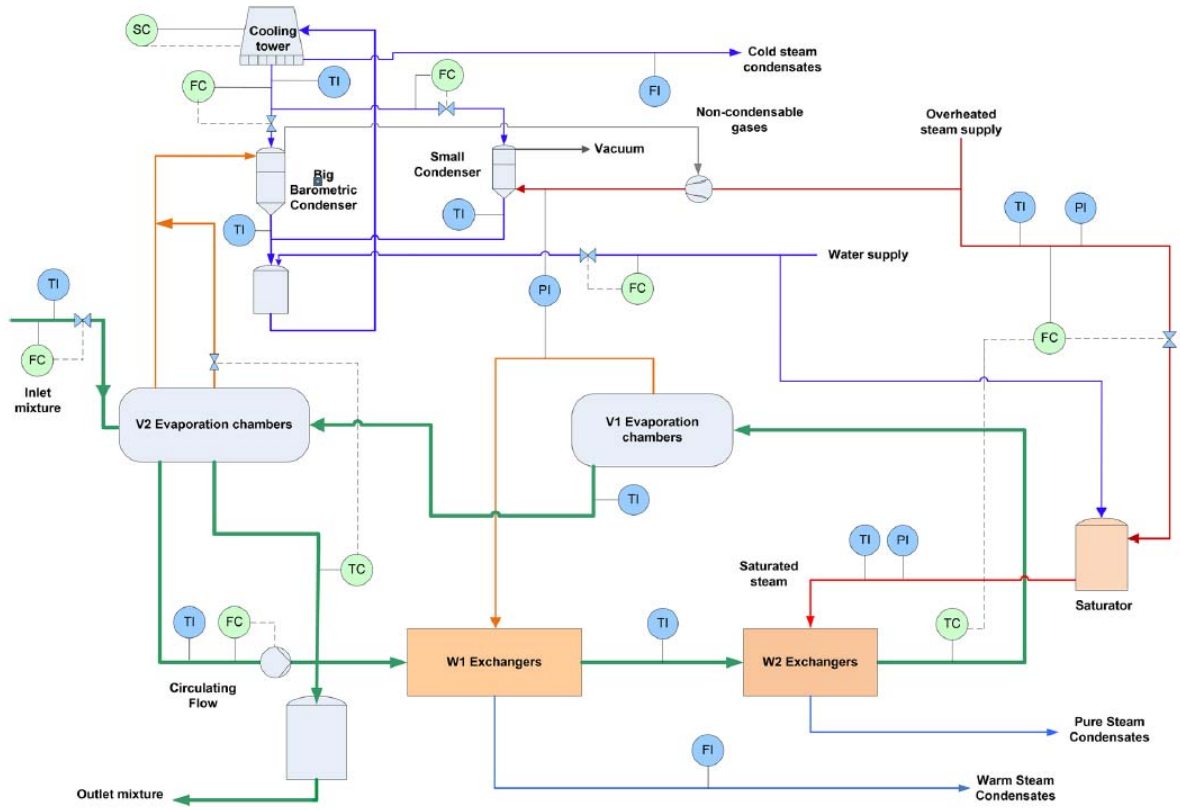


Figure 5: Simplified scheme of a evaporation unit with location of existent instrumentation

The system works as a multiple-effect evaporation, achieved on the one hand thanks to the pressure drop in the chambers V2 created by the condenser, and in the other hand to vacuum pumps connected to the evaporation chambers labelled as V1. The evaporation plant, when connected to the main process, receives an input spinbath liquid, which is a mixture of water with acid and other chemical components plus residual of organic material. The goal is to concentrate the solution by removing certain amount of water. To achieve this, the acid bath is pumped through the line of heat exchangers W1 and W2 in counter current with saturated-steam flows (some coming from the evaporators V1 and other from a fresh steam generated in a boiler) to increase its temperature. Then, the hot mixture enters sequentially the low-pressure chambers V1, which forces a partial evaporation of water. Afterwards, an additional evaporation phase is performed in the last set of chambers V2 thanks to the condenser, which sucks out steam by condensing it with cold water from a cooling tower. Finally, part of the concentrated spinbath liquid leaves the process and the rest mixes with the input, being recirculated through the process.

3.3.2 Data pre-preprocessing

Before the actual construction of evaporator models is carried out the raw data undergoes an intensive data pre-processing. In this procedure the collected raw data is checked, filtered and modified. The reason for the data pre-processing is to make sure that the data used to later create the evaporator model does not contain gross measurement errors, unstable operation points or operation points outside of the model scope. Whether or not data is inside the model scope is up to the specific evaporator and proper process knowledge is necessary to determine the right boundaries for each evaporator. Since the modelling of the evaporators is a recurrent task and has to be individually carried for each evaporator Lenzing has developed in cooperation with TUDO a semi-automatic matlab tool for the entire modelling procedure of the evaporators which also includes the data pre-processing step. The tool filters data that is outside of the relevant operating range, data that is obtained from unstable operation points and outliners. The remaining data is screened for step changes in the evaporation capacity and a set of steady-state operating points before and after changes is obtained. In Figure 6 the interface of the matlab tool is depicted.

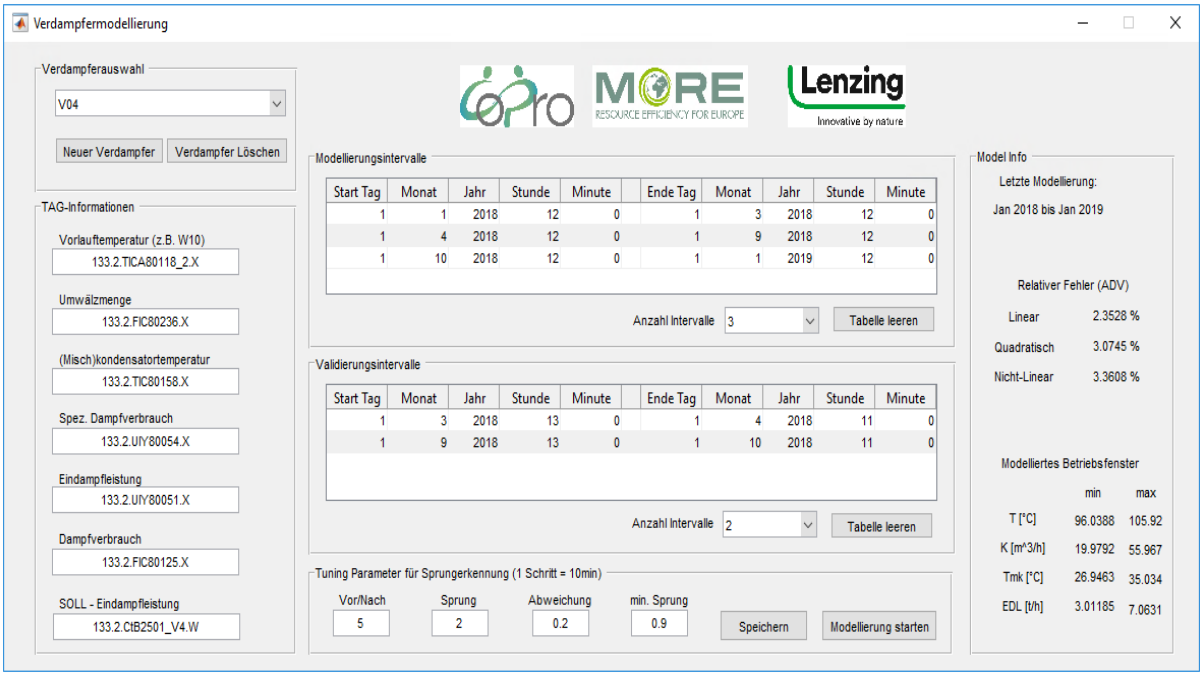


Figure 6: User interface of the matlab tool used for data-based modelling of the evaporators

In the interface the necessary data inputs (PI TAGs) have to be selected as well as the timespan for the data used for modelling and validation. Furthermore some tuning parameters have to be selected in order to define steady state conditions and normal operation points. After the first automatic selection of the step changes and corresponding operation points is performed by the matlab algorithm the step changes are plotted and the user hast to manually select the step changes considered as appropriate for the later modelling and validation task.

3.3.3 Model type

The three key values to describe the operation of an evaporator in the Lenzing use case are the evaporation capacity (EC), the absolute steam consumption (AC) and the specific steam consumption (SC). Experimental data indicated that, the EC and the SC can be described as a linear function of the two main control variables, which are spinbath cycle flow rate (F) and the spinbath temperature (T)

after the last steam heat exchanger, as well as the condenser temperature (T_{mk}) and the fouling state ($f_{fouling}$).

$$EC = a_1 T + a_2 F + a_3 T_{mk} + f_{fouling,1} \quad (15)$$

$$SC = b_1 T + b_2 F + b_3 T_{mk} + f_{fouling,2} \quad (16)$$

$$AC = EO \cdot SC \quad (17)$$

The model parameters (a, b) of the linear regression models are fitted with the pre-processed data using the "LM_DER" solver of the OPTI Toolbox in matlab.

3.3.4 Model validation results

At the end of the modelling procedure of a single evaporator the developed matlab tool chooses randomly three different step changes from the pre-processed data selected for validation. For each of the three target values (EC, SC, AC) the measured data and the model prediction are plotted over the tree step changes. This allows the user a first visual quality check of the only just created models. In Figure 7 the resulting 9 plots from an actual evaporator are illustrated. Additionally to the visual quality check the modelling tool also calculates for each target value the relative Mean Squared Prediction Error (RMSPE) and indicates the results on the right side of the user interface (see Figure 6). The usual RMSPE for the evaporator models is between 1% and 6%. Furthermore since the validity area is crucial for the quality of the model predictions and extrapolation is not desired the tool also indicates the important operation boundaries from the data used for modelling. This way the user can directly recognize if the data used for training the evaporator model is covering all the necessary operation points.

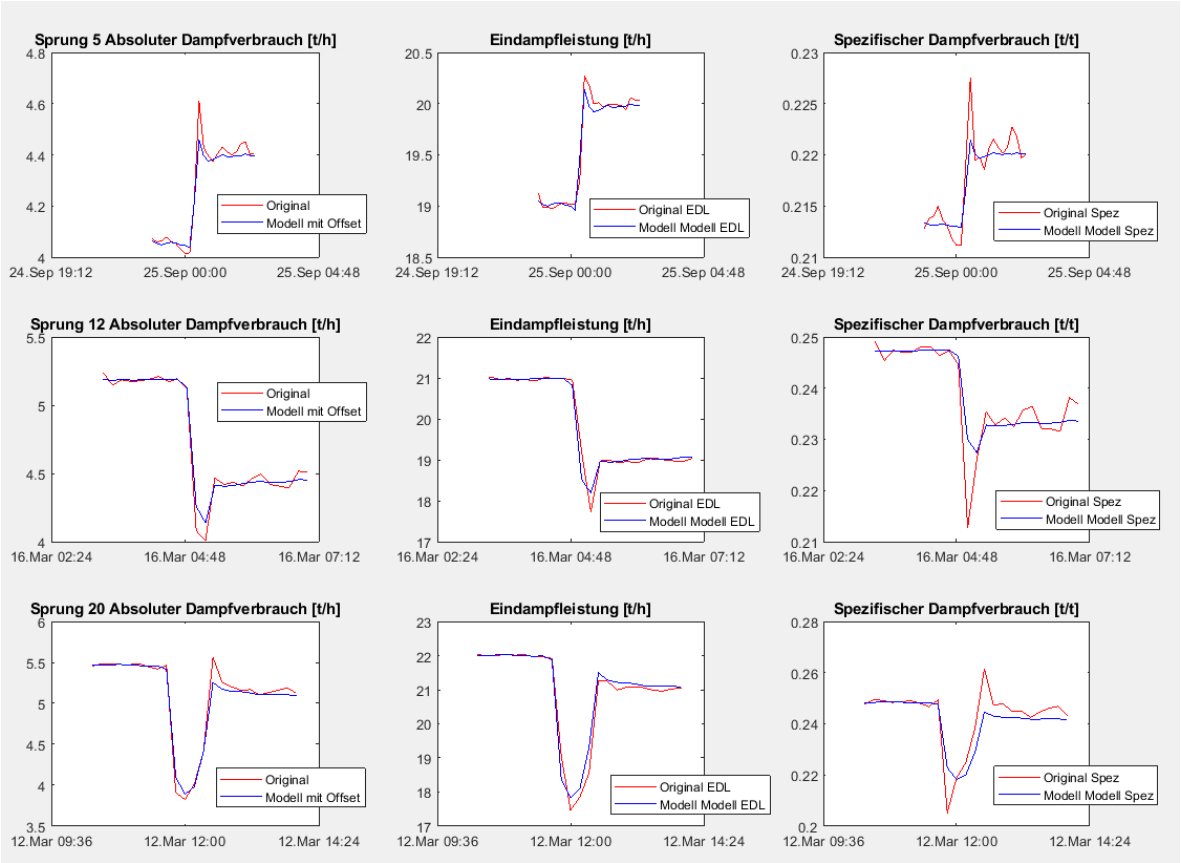


Figure 7: Comparision between the meassured (red) and predicted (blue) target values for three different step changes used for validation

3.3.5 How the models were used in the COPRO project

The predictions from the created models are used as inputs for the optimization of the evaporator load allocation of the evaporator network. The goal of the optimization is to minimize the overall steam consumption of the evaporation network by an efficient load allocation for each evaporator. The optimization problem is formulated as a Mixed-Integer Nonlinear Program. More information about the optimization of the evaporator network can be found in (Kalliski, et al., 2017). The results of the optimization are implemented in a decision support system (DSS) running in a control room at the Lenzing site. The DSS indicates to the operator the results of the optimization and therefore the most efficient load allocation under current production constraints.

3.1 Best Demonstrated Practice models (INEOS use case)

3.1.1 Introduction

The process industry is constantly developing methods for the evaluation of their resource consumption and the identification of possible improvement potentials. For this purpose, INEOS in Köln provides the operators with a performance reference model named Best Demonstrated Practice (BDP), that represents the most resource efficient and stable production at a specific set of non-influenceable circumstances like ambient conditions or feedstock quality. By comparing the current Energy Performance Indicator (EnPI) with its BDP, Operational Improvement Potentials (OIP) can be identified. Figure 8 depicts an illustrative example that represents the concept. The task of the operator is to keep the OIP, defined as the distance between the EnPI and the BDP, as small as

possible. INEOS in Köln has developed and implemented a surrogate modelling approach for calculating the BDP curves from the historical data.

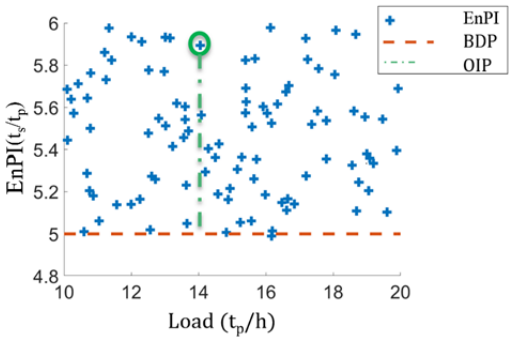


Figure 8 Illustrative example of BDP concept

3.1.2 Best Demonstrated Practice

This section introduces the steps of the BDP modelling approach developed by INOES in Köln briefly. A detailed description of the procedure can be found in (Beisheim, B., Rahimi-Adli, K., Krämer, S., and Engell, S.;, 2018b).

The method comprises 5 general stages, depicted in Figure 9. The goal of the first step, the acquisition of measurement data, is to collect representative and reliable measurements of the performance of the plant. This step deals with aspects like removal of gross errors, stationarity, and removal of abnormal operating windows. Pre-treatment of the data is the next step.

Figure 9. Steps of the BDP modelling procedure

This step is done to remove the outliers, classify the data and to standardize it. Mean centering and unit variance scaling is used for data standardization. The standardization is beneficial for the third step, data clustering, which will be discussed next. Data clustering is introduced to select a few number of representative points from the large amount of the measurement data to be used for the model fitting. In this modelling approach, the kmeans++ algorithm (Arthur & Vassilvitskii, 2007), is an extension of the kmeans algorithm (MacQueen, 1967), is applied. Due to the application of the Euclidean distance as the distance metric for the clustering of the data, the clustering algorithm is sensitive to the magnitude of the data and the assignment of a higher influence on a variable with a higher magnitude is likely. The standardization of the data in the second step is done to avoid such cases. As the outcome of clustering, a center is assigned to each cluster that represents the average of the points of the cluster. Considering that the goal of the BDP model is the calculation of the efficient operational domain, the cluster centers are not the correct representatives. Instead, a percentile analysis for each cluster is performed:

$$\mathrm{r}_j = \frac{1}{|\Re_j|}\sum_{\mathrm{x}\in\Re_j} \mathrm{x} \tag{18}$$

$$\mathrm{x} \in \Re_j \quad \forall \; P_{j,n} \leq \mathrm{EnPI}(\mathrm{x}) \leq P_{j,m} \tag{19}$$

$$\Re_j \subseteq \mathcal{X} \tag{20}$$

$$\mathrm{EnPI}_{\Re_j} = \frac{1}{|\Re_j|}\sum_{\mathrm{x}\in\Re_j} \mathrm{EnPI}(\mathrm{x}) \,, \tag{21}$$

where $\mathfrak{R}_j$ denotes the set of the assigned points to cluster j, $|\mathfrak{R}_j|$ is cardinality of the set. r_j is the calculated cluster center. $P_{j,n}$, $P_{j,m}$denote the lower and upper percentile bounds of the respective cluster. These bounds are used to select the representatives, $\mathrm{EnPI}_{\mathfrak{R}_j}$, as achievable good operation points for each cluster.

The surrogate model development is an adaptation of the ALAMO approach (Cozad, Sahinidis, & Miller, 2014). ALAMO generates simple and accurate models from simulated or experimental data. In order to reduce the shortcomings of linear regression models, ALAMO uses simple basis functions to transform the inputs and chooses a combination of these transformed inputs to fit the responses with an acceptable accuracy. The set of basis functions must be defined by the user and ALAMO selects the most suitable ones and their respective parameters through solving an optimization problem. The details of the implemented adapted version of the ALAMO approach can be found in (Beisheim, B., Rahimi-Adli, K., Krämer, S., and Engell, S.;, 2018b). The model is fitted by solving an optimization problem formulated as:

$$\min_{\beta,y} \sum_{i=1}^{N} e_i \tag{22}$$

$$\text{s.t.} \quad e_i \geq z_i - \sum_{j \in \mathfrak{V}} \beta_j X_{ij} \qquad i = 1, \dots, N \tag{23}$$

$$e_i \geq \sum_{j \in \mathfrak{V}} \beta_j X_{ij} - z_i \qquad i = 1, \dots, N \tag{24}$$

$$\sum_{j \in \mathfrak{V}} y_j = B \tag{25}$$

$$-U_j(1 - y_j) \leq \sum_{i=1}^{N} X_{ij} \left(z_i - \sum_{j \in \mathfrak{V}} \beta_j X_{ij} \right) \leq U_j(1 - y_j) \qquad j \in \mathfrak{V} \tag{26}$$

$$\beta^l y_j \leq \beta_j \leq \beta^u y_j \qquad j \in \mathfrak{V} \tag{27}$$

$$y_j = \{0,1\} \qquad j \in \mathfrak{V}, \tag{28}$$

where z_i are the values of representative points for the clusters calculated in percentile analysis ($\mathrm{EnPI}_{\mathfrak{R}_j}$), X is the matrix of the transformed inputs and β is the coefficient vector of X. $\mathfrak{V}$ is the set of the transformed functions and B denotes the maximum number of terms allowed for the model. The binary variables y_j correspond to the transformed basis functions and are equal to 1 if their respective basis function is selected. β^u, and β^l are the upper and lower bounds for the coefficient vector. N is the number of the representative points and the indices i and j correspond to the representatives and the basis functions respectively.

The optimization is done iteratively by increasing the value of B and the modified corrected Akaike Information Criterion ($\mathrm{AIC_c}$) is used to select the suitable level of complexity of the model. Eq.(26) is an additional constraint, which as described in (Beisheim, B., Rahimi-Adli, K., Krämer, S., and Engell, S.;, 2018b) uses the relaxed bounds U_j to convert the problem formulation into an MILP.

The final BDP model have the following general structure:

$$\mathrm{BDP} = \sum_{i=1}^{n} \sum_{j=1}^{m} \beta_{ij} f_i(x_j),$$

where x_j denotes the model input variable j. These inputs are a specific set of non-influenceable circumstances like ambient conditions or plant load. f_i denotes the basis function i. andβ_{ij} denotes the regression factor for basis function i and model input j.

3.1.3 Application to the production data

The introduced approach in the previous section is actively used at INEOS in Köln for the calculation of the BDPs. The approach described in 3.1.1 is used by defining $[1,x^{\pm[1;2;3]},\exp(x)]$ as the basis functions. An example of this application for the ethylene oxide plant of INEOS in Köln is presented. The load of the plant is defined as the influencing factor and the model is calculated. Figure 10 depicts the resulting model and the data used for its fitting (Rahimi-Adli, Schiermoch, Beisheim, Wenzel, & Engell, 2019). The x-axis presents the load of the plant, and the y-axis shows the EnPI of the specific steam consumption, which is defined as amount of the steam divided by the amount of the product. The colored data represent the clustered data. The red circles and the green + signs represent the calculated average and best operation for the clusters respectively. Considering that the model is data based, extrapolation of the model should be avoided and that is the reason that the validity range of the model is defined as the region between the minimum and the maximum of the load range used for the model fitting.

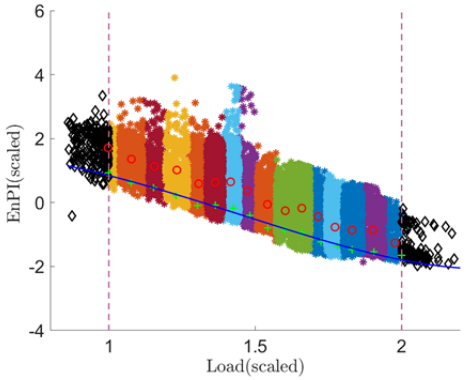


Figure 10 Surrogate model of the product processing section of the ethylene oxide plant

The developed algorithm is generic and can be used to fit models with more than one influencing factor. Furthermore, the developed algorithm and modelling toolbox is not tailored to be used only at INEOS in Köln, and can be implemented in other manufacturing sectors as well.

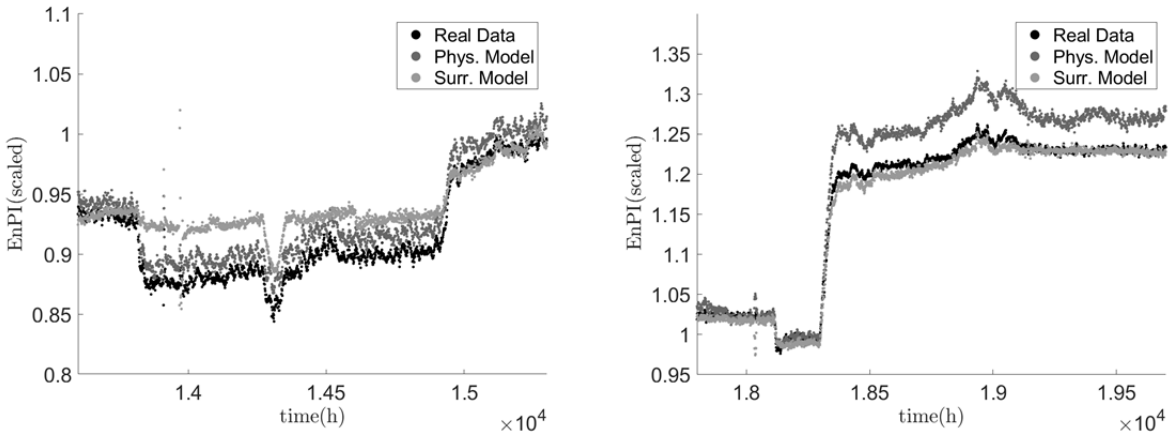
3.1.4 Comparison of the results with a rigorous approach

In order to compare the advantages and possible shortcoming of the developed approach, INEOS in Köln developed and compared a surrogate model and a physical model for the steam production of the reactor of the ethylene oxide plant. This reaction is exothermic and the resulting produced heat is removed by a cooling fluid in the reactor jacket. The cooling fluid is subsequently cooled down against water in a heat exchanger, producing steam.

The surrogate model for this case is developed by taking the load of the reactor and the selectivity of the catalyst as influencing factors. The physical model is developed using the detailed energy balance of the reactor and the cooling system, where the steam is being produced. The details to the developed models can be found in (Rahimi-Adli, Schiermoch, Beisheim, Wenzel, & Engell, 2019).

The comparison between the results of these two models is presented in Figure 11. The figures present the trajectory of the EnPI of the steam production against time in two interesting operating windows. For the operating windows presented in Figure 11a the surrogate model predicts a larger amount of steam production compared to the data and to the physical model. After investigating the possible reasons for this difference, it was concluded that the reactor has a lower conversion value

for the ethylene in this time period, which is not considered in the surrogate model and thereby the results of it deviate from the reality. In Figure 11b however, the predicted amount for the steam production by the physical model is significantly higher. Investigations concluded that the reason for this event is the limited heat removal capacity of the heat exchanger that is used to cool down the cooling fluid of the reactor. Thereby, in case of a large heat production in the reactor, the heat exchanger is not able to convert all of the energy into steam, and the cooling fluid is cooled down against a large stream of cooling water and the heat is lost. Since the surrogate model is based only on the production data, it is not able to capture such information in contrast to the physical model.



a. Comparison 1

b. Comparison 2

Figure 11 Comparison of the physical and the surrogate BDP models against production data (Rahimi-Adli, Schiermoch, Beisheim, Wenzel, & Engell, 2019)

As a result of the developed models and the comparisons done in this work, it was observed that the developed surrogate modelling approach shows a good performance in modeling the BDP curves from the production data. The modelling effort of this method is significantly lower when compared to the physical models. The time spent for development of the surrogate model is approximately 20% of the time required for the development of the physical model. Nevertheless, the range of validity and power of interpretation of the surrogate model is constrained to the range of the training data, and its results beyond these limits are not to be trusted. Furthermore, as presented in Figure 11, a physical model is a better choice, when the goal is to identify the possible improvement potential through possible structural changes in the plants, which is not possible using the surrogate modelling approach.

3.2 Overall plant network modelling (INEOS ammonia network optimization use case)

For the INEOS in Cologne Use Case of the ammonia network optimization, the systems listed in Table 2 and their linking topology shown in Figure 12 had to be modelled efficiently. In the following, the generic models used for each subsystems are described briefly. The details for every model can be found in (Wenzel et al. 2019).

3.2.1 Plant models

For the purpose of optimal scheduling of the network, only stationary models for the input-output relations are considered. In addition, the allowed dynamics of operation are captured in ramping constraints. The plant models in the network are either modelled as single plants, or as plant with multiple units. The plants of type single plant depend on single product stream that determines all other streams of material and energy that are attributed to this plant model:

$$\dot{m}^{(j)}_{(r,s,s')} = \left(a_{r,p} + b_{r,p} \cdot \vartheta^{(j)}_a\right) \cdot y^{(j)}_{p,on} + \left(c_{r,p} + d_{r,p} \cdot \vartheta^{(j)}_a\right) \cdot \dot{m}^{(j)}_{(prod,p)}, \forall (r,s,s') \in \Gamma_{in,p} \cup \Gamma_{out,p}, p \in P, j \in J, \quad (1)$$

$$P^{(j)}_{(el.,p)} = \left(a_{el.,p} + b_{el.,p} \cdot \vartheta^{(j)}_a\right) \cdot y^{(j)}_{p,on} + \left(c_{el.p} + d_{el.,p} \cdot \vartheta^{(j)}_a\right) \cdot \dot{m}^{(j)}_{(prod,p)}, \forall\, p \in P, j \in J, \quad (2)$$

where (1) describes the affine relation of the produced amount of product to the mass streams of all other streams and (2) describes the consumption or production of electric power. In addition, there is a load independent term that is assigned to the plant model. The load independent term accounts for the consumption of material and energy if the plant does not operate (see Figure 13).

A plant with multiple parallel units is modelled similarly, but the individual product streams result from a summation of the multiple parallel units in the plant

$$\dot{m}^{(j)}_{(prod,p)} = \sum_{e_u \in E_{U,p}} \dot{m}^{(j)}_{(prod,p,e_u)}, \forall p \in P_{mU}, j \in J. \quad (3)$$

A block diagram of this model structure is shown in Figure 13. The parameters of the model equations were either taken from the internal planning models that are used by INEOS in Cologne today, or they have been identified by regression from polished production data (see Chapter **Fehler! Verweisquelle konnte nicht gefunden werden.**).

3.2.2 Tanks models

The tank model have been formulated using simple mass balances around the respective tank. In addition to the mass balances around the tanks, the operating modes of the plants play a crucial role. The necessary identified modes of the tanks are shown in Table 2.

$$m^{(j+1)}_{(r,t)} = m^{(j)}_{(r,t)} + \Delta t \cdot \left(\sum_{(r,s,s') \in \Gamma_{in,t}} \dot{m}^{(j)}_{r,s,s'} - \sum_{(r,s,s') \in \Gamma_{out,t}} \dot{m}^{(j)}_{r,s,s'}\right) \forall r \in R, t \in T, j \in J. \quad (4)$$

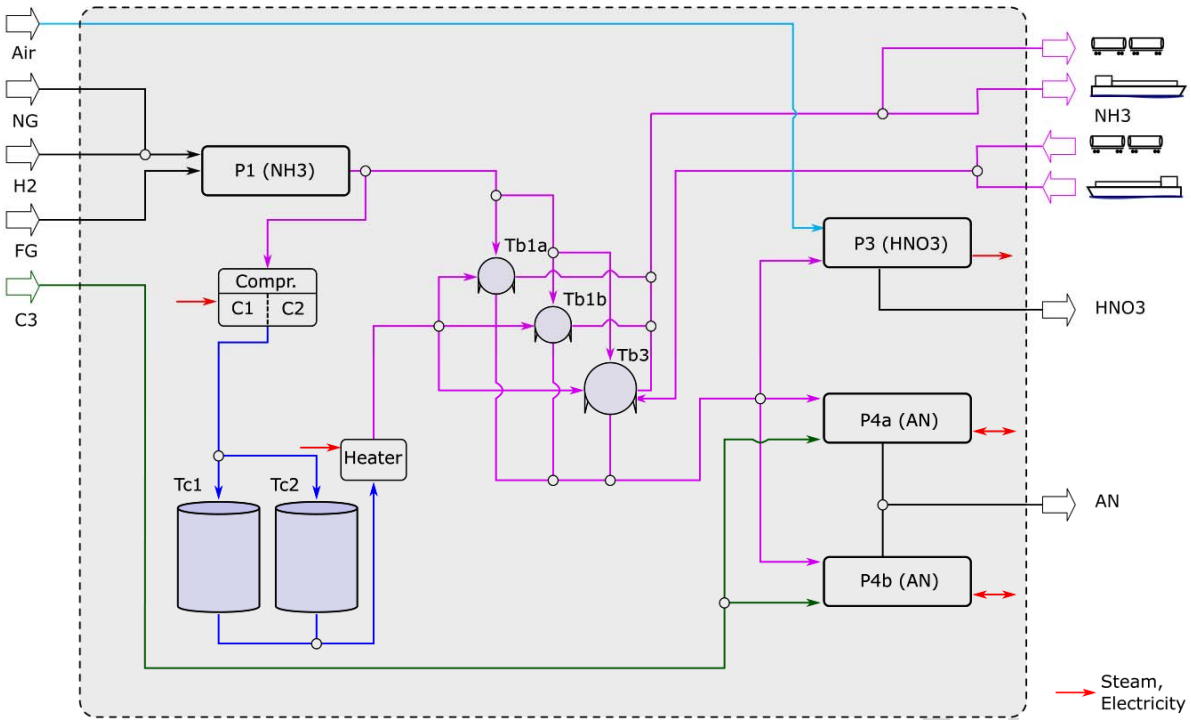


Figure 12: Schematic topology of the INEOS in Köln ammonia distribution network with its processing plants P1, P3, P4a, and P4b (Wenzel et al. 2019)

Table 2: Systems considered in the ammonia distribution network (Wenzel et al. 2019)

System	Symbol	Modes	Comment
Ammonia plant	P1	On, off	NH3 producer
Nitric acid plant	P3	On, off	NH3 consumer
Acrylonitrile plant	P4a	On, off	NH3 consumer
Reactor 1	P4aReac1	On, off, shutting down, starting up	NH3 consumer
Reactor 2	P4aReac2	On, off, shutting down, starting up	NH3 consumer
Acrylonitrile plant	P4b	On, off	NH3 consumer
Reactor 1	P4bReac1	On, off, shutting down, starting up	NH3 consumer
Reactor 2	P4bReac2	On, off, shutting down, starting up	NH3 consumer
Reactor 3	P4bReac3	On, off, shutting down, starting up	NH3 consumer
Reactor 4	P4bReac4	On, off, shutting down, starting up	NH3 consumer

Deep-cooled tank	Tc1	Fill, disc, idle	NH3 deep-cooled storage
Deep-cooled tank	Tc2	Fill, disc, idle	NH3 deep-cooled storage
Buffer tank	Tb1a	Fill, idle	NH3 warm storage
Buffer tank	Tb2a	Fill, idle	NH3 warm storage
Buffer tank	Tb3	Fill, idle, discs, disct, discp	NH3 warm storage, handles import/export
Compressor	C1	On, off	State change
Compressor	C2	On, off	State change
Heater	H1	On	Energy consumption linear with processed amount

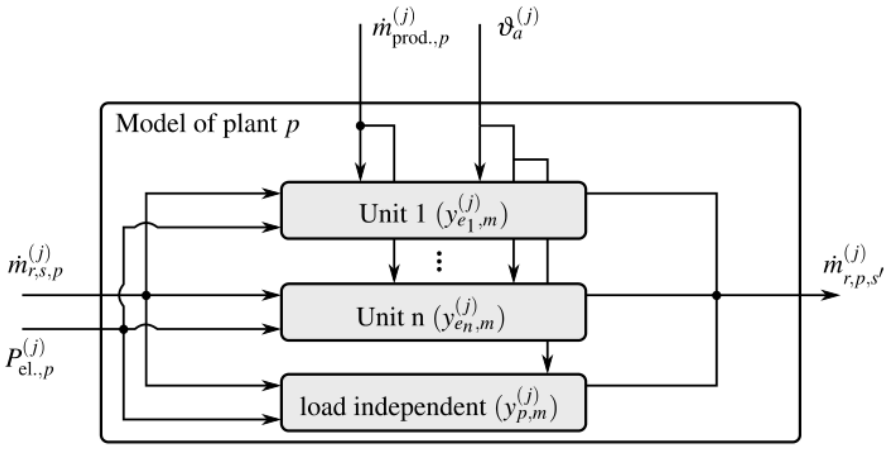


Figure 13: Modelling of plants with multiple parallel units. The continuous input to the model is the desired product stream (Wenzel et al. 2019)

3.2.3 Models for compressors and heaters

The models for the compressors and heaters were identified similarly to the model of the production plants as linear models.

3.2.4 Data pre-processing

In the cases where the models were present, obviously not data preprocessing needed to be done. In the cases where affine models needed to be identified from production data, first necessary sets of data needed to be found that excluded shutdown and maintenance periods. This was done by iteratively searching for times with no production (see Figure 14).

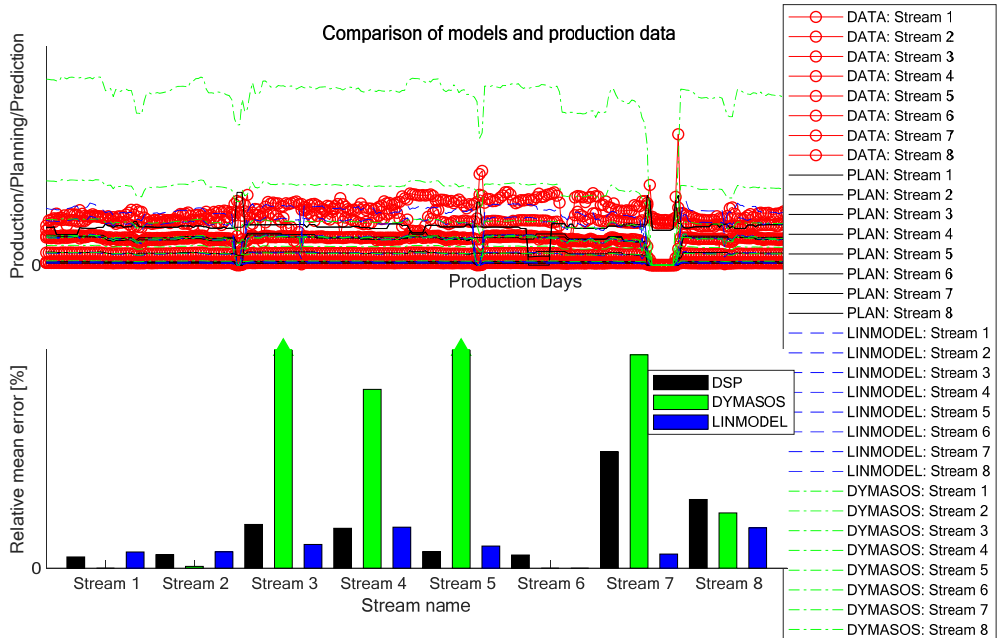


Figure 14: Analysis of the model quality in comparison to production data. One can visually identify times with no production. These times were eliminated before fitting the model parameters.

3.2.5 Model type (linear, piece-wise affine etc), which parameters were fitted from data and why the particular modelling type was chosen

The model type is a mixed-integer linear programming problem, where the model equations of the plants and tanks themselves are affine model equations with discrete decisions incorporated. This particular model type has been chosen, because the dynamic effects of the processes themselves can be neglected due to the length of the optimization horizon. The dynamics in the overall problem results from ramping constraints and minimal or fixed stay time constraints.

3.2.6 Efficient modelling strategy

In order to quickly model the overall optimization problem and in order to be able to quickly extend to problem and the scope of the optimization or to optimize structurally similar use cases a tabular data-based modelling strategy was employed. The core of the implementation is a generic Julia code that formulates the optimization model in JuMP syntax, which can then be passed over to any commercial or open source solver for which an interface to JuMP exists.

The core code opens a database where for each type of model (plant, tanks, etc.) a data sheet with the models that need to be considered is stored. An example for such a data sheet can be found in Figure 15. The example shows a data sheet where each row corresponds to a single tank. The core code of the tool then identifies the entries and writes down the model equations necessary for this type of equipment.

The topology of the overall problem and the linking streams are modelled in a separate stream table that contains the exact routing of the pipes. The usage of this strategy enables the engineer working on this problem to quickly add, remove, and review the constituent systems without the necessity to write a single line of code. The use of the data-based approach for the modelling facilitates further

development of interactive GUIs and interfaces to the tool to build interacting tools, if demanded by the customer.

Name	Resource	lb	Unit (lb)	ub	Unit (ub)	ub_disc	Unit (ub_disc)	ub_fill	Unit (ub_fill)
Filtern	Filtern	Filtern	Filtern	Filtern	Filtern	Filtern	Filtern	Filtern	Filtern
Tb1a	NH3	100	t	1000	t	12	t/h	12	t/h
Tb1b	NH3	100	t	1000	t	12	t/h	12	t/h
Tb2	NH3	150	t	1400	t	20	t/h	20	t/h

Figure 15: The values in the table have been altered for the sake of confidentiality.

3.2.7 Fitting of data-driven models

In addition to the already existing models of INEOS in Köln, TU Dortmund University derived data-driven affine models from production and planning data. Although, the models do not capture the nonlinearities of the operating systems over the complete operating range, the precision of affine models has been considered to be suitable for planning purposes.

The model quality has been assessed and compared to existing models (see Deliverable D1.2 for Details). Figure 16 and Figure 17 show an exemplary comparison of different models. It can be seen that the affine models have the smallest deviation from the production data.

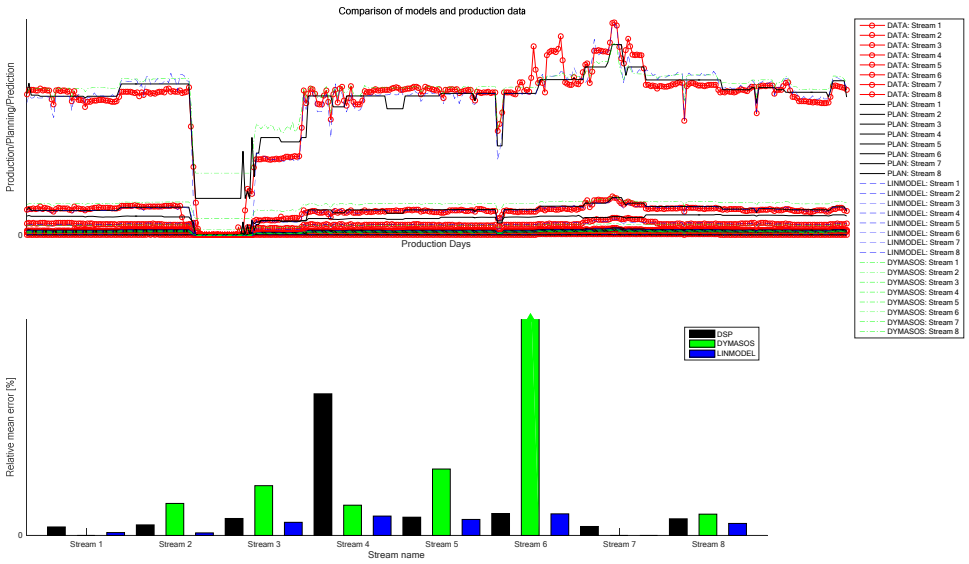


Figure 16: Comparison of different models created for one of the plants at INEOS in Köln.

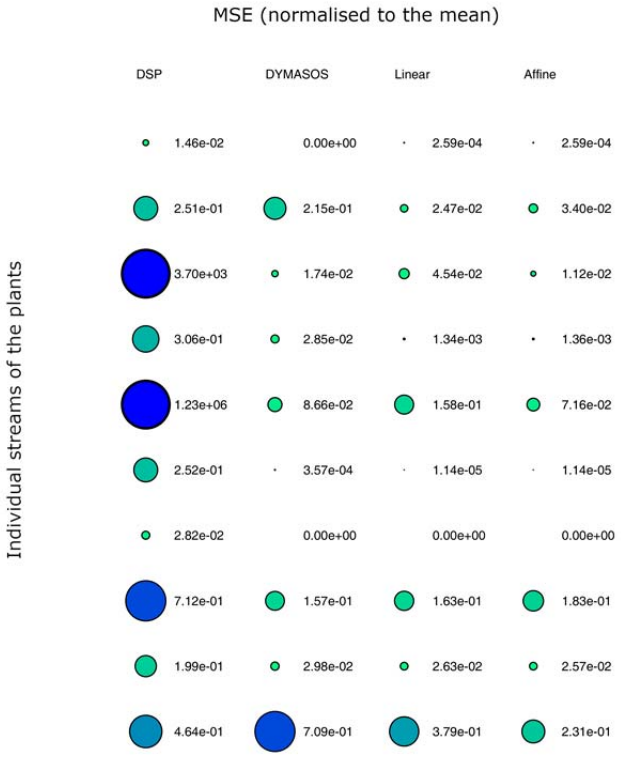


Figure 17: Circle plots for the assessment of prediction quality. On the vertical axis a number of stream for a particular production plant are listed. In the horizontal direction, a comparison is possible between the different models that are available. Smaller circles denote smaller mean squared error. This dashboard enables a fast assessment of the quality of the models and indicates the streams with the largest mismatch.

The error is calculated as follows

$$e = \overline{\left(\frac{y_{plant} - y_{model}}{\overline{y_{plant}}}\right)^2}$$

3.2.8 Use of the models in the COPRO project

The models were used for the development of an optimisation model. The optimisation model can be used to derive an optimal schedule for the operation of the overall network. The results of the optimization model that were inferred by using the described models above are plausible and validated by the industrial partner INEOS in Köln. In Figure 18 the comparison of the optimized tank levels with the recorded data at INEOS in Köln is shown and in Figure 19 the operating level of the major plants is shown. With the derived models in the project, the operation of the tanks and of the plants could be optimized satisfactorily.

The use of the derived models now enables the industrial partner to employ the derived tool in what-if-scenarios to investigate the performance of the network and to detect bottlenecks in the schedule resulting from the interplay of operations, demand side response and handling the logistics. Using the models significant saving potential could be revealed. For more details on the optimization results see (Wenzel et al. 2019).

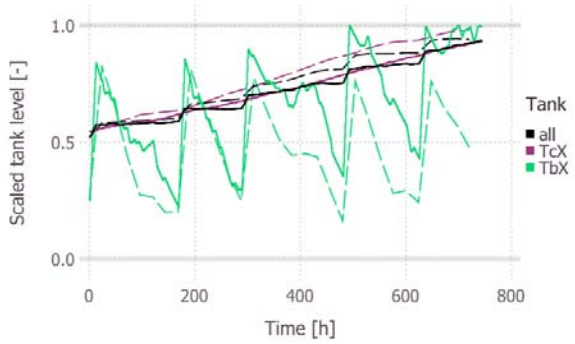


Figure 18: Integrated amount in the tanks grouped by type of the tank (TcX = Tc1 + Tc2, TbX = Tb1a + Tb1b + Tb3). Dashed lines correspond to the recorded data (Wenzel et al. 2019).

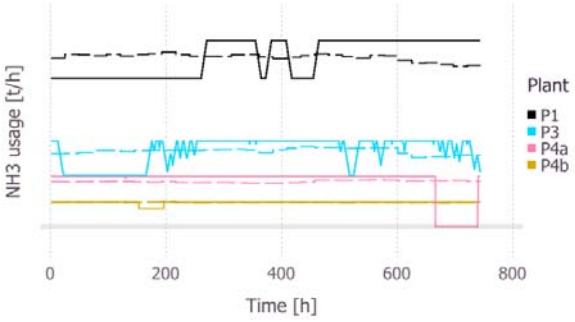


Figure 19: Comparison of the optimized and recorded plant production levels w.r.t. the usage of ammonia (dashed lines represent recorded data). The exact numbers of the usage are confidential (Wenzel et al. 2019).

3.3 Cracker coking modelling (INEOS cracker use case)

A data-driven model was developed by Divis to predict coking in the INEOS cracker. The development for this model was already described in detail in a COPRO deliverable. (COPRO partners, 2019). An affine model was used to predict a the future for a timeseries based on past values for a transformed variable, namely the *coking index*.

The data analysis and pre-processing proved to be a large part of the effort for this use case. The data quality and data accessibility were the most important factor for data-driven modelling. Since machine learning algorithms assume that the used data is correct much effort was put into data pre-processing. This step includes the selection of relevant variables, as well as removal of erroneous, irrelevant or conflicting information in the data set such as downtimes, maintenance operations etc..

For the INEOS in Köln cracker use case the pre-processing step included the calculation of the Naphtha composition from different Naphtha sources as well as the cleaning of the data for the decoking periods.

Next to general considerations about the data quality the format in which the data is used requires attention. This can include smoothing, feature and pattern extraction for time series data as well as the selection of the suited modelling approach. After this a regression approach was used to predict the coking index. While several smoothing techniques and black-box-modelling approaches were tested, the simplicity of a linear model was noted as the most suitable. During CoPro the close collaboration from data providers and analysts became a crucial aspect to the success of data-driven modelling.

4 Hybrid modelling

4.1 Hybrid model

Hybrid modelling has been recognized as a promising approach for process systems (M. von Stosch, 2014). In hybrid modelling both parametric and non-parameteric models from different modelling sources are combined. A typical application is for the combination of first-principle and data-driven modeling. The parametric part of the model is derived from first-principles using a parameter-fitting approach which the data-driven part is derived using a machine learning / non-parametric fitting algorithm.

In this section of the report we present a systematic approach to building hybrid models. The hybrid modeling approach is applied by UVA to the surface evaporator unit operations from the Lenzing use case. PSE has implemented a prototype hybrid modelling tool that allows fitting of data-based and hybrid models using the Python sk-learn toolkit or a PSE implementation of the ALAMO algorithm (Wilson & Sahinidis, 2017) coupled with model prediction in the gPROMS kernel using a plug-in function.

4.2 Approach for hybrid modelling

The preferred option in order to develop efficient and reliable models for supporting operators and managers in their decisions is building models that combine as much physical information as possible/acceptable with relationships obtained from experimental data collected from the plant (Zorzetto, Maciel Filho, & Wolf-Maciel, 2000). In this way, these *hybrid* or *grey-box* models get a high level of matching with the actual plant and, importantly, they get improved prediction capabilities, as their outputs will fulfil, at least, the considered basic physical laws in the whole region of operation required.

Here in CoPro we propose the following methodology for systematic building of grey-box models (Pitarch, Sala, & de Prada, A systematic grey-box modeling methodology via data reconciliation and SOS constrained regression, 2019).

Stage 1 – First-principles building. In this stage, the modeller states a set of first-principles equations

$$\frac{dx}{dt} = f(x(t), u(t), z(t), \theta), \quad h(x(t), u(t), z(t), \theta) = 0, \tag{1}$$

that must be certainly fulfilled by the actual process. Here $x \in \mathbb{R}^n$ are the system states, $u \in \mathbb{R}^m$ are the *known* process inputs (manipulated variables or measured disturbances), $z \in \mathbb{R}^q$ are algebraic variables (internal states whose dynamics is neglected or arbitrary unknown inputs), $\theta \in \mathbb{R}^\rho$ are model parameters assumed constant, and $f(\cdot) \in \mathbb{R}^n, h(\cdot) \in \mathbb{R}^l$ can be nonlinear functions.

The number of equations in (1) and their complexity is decided by the modeller attending to : 1) the physical knowledge on the process, 2) the computational demand required by the application where the model is to be used (e.g. control, real-time optimization or scheduling) and 3) the expected level of accuracy in the predictions.

Let us now assume that model (1) is incomplete, meaning that all process variables cannot be fully determined with the current number of equations and the actual inputs u, i.e., there are $q - n - l -$

$m > 0$ variables $z^* \subset z$ that are "free", i.e. arbitrary time varying. Therefore, (1) is just a backbone of basic physical laws that need to be completed with some extra equations $r(x,u,z) = 0$ obtained from plant experimental data.

Stage 2 – Data reconciliation and estimation. With the partial model (1) and a set of process input-output data measured from the plant, solve the following dynamic data reconciliation (DDR) problem to get coherent estimations of all variables $(\hat{x}, \hat{u}, \hat{z})$ over time, and for the parameters $\hat{\theta}$:

$$\min_{u,z,w,\theta} \int_H K^2 \sum_{i=1}^{s} \left(\frac{|\epsilon_i(t)|}{K} - \log\left(1 - \frac{|\epsilon_i(t)|}{K}\right)\right) + \sum_{j=1}^{r} w_j(t)^2 \, dt$$

$$\text{s.t.:} \quad \frac{dx}{dt} = f(x(t), u(t), z(t), \theta), \qquad x(0) = x_0 \tag{2}$$

$$\frac{dz^*}{dt} = \omega_c \cdot z^*(t) + \kappa \cdot w(t), \qquad z^*(0) = z_0^*$$

$$h(x(t), u(t), z(t), \theta) = 0, \qquad g(x(t), u(t), z(t), \theta) \geq 0$$

Where $\epsilon_i := (y_i - \hat{y}_i)/\sigma_i$, being y the process measurable variables (either internal states, process inputs or outputs) and σ their corresponding sensor standard deviations; $K > 0$ is an user-defined parameter to tune the fair estimator for insensitivity to gross errors (Llanos, Sanchéz, & Maronna, 2015); and z^* are supposed to vary conforming a wide-sense stationary process w whose power spectral density is limited by bandwidths $\omega_c > 0$, normally chosen by the modeller/engineer.

The vector of additional constraints $g(\cdot) \geq 0$ is stated to force any physical insight on the model variables and its variation over time that the engineer might have, e.g. upper and lower bounds, some variables being always larger than others, etc. Moreover, the initial states x_0 and z_0^* may be either assumed known from the estimations provided at the previous run, or also left decision variables with some penalty w.r.t. such previous estimations in the objective function.

Problem (2) is normally solved in a moving-horizon fashion, discretising the time horizon H, and following either a sequential or simultaneous approach. Implementation details are omitted here, but the reader is referred to the review on DDR (Pitarch & de Prada, D2. 1–Report on Dynamic Data Reconciliation of Large-Scale Processes, 2018) for such a detailed description.

Stage 3 – Constrained regression. Once estimations for the unknown inputs z^* are available, their hidden relations with other variables $r(x,u,z) = 0$ will be sought via suitable machine-learning approaches to complete model (1). The literature on machine learning is rather broad, but not any approach can take advantage of the partial knowledge that the engineer may have about z^*. Therefore, *extra* (local or global) conditions on the regression candidate models are to be enforced in order to guarantee reliable interpolation, but also extrapolation, to allow z^* taking values outside the range where experimental data was collected. This feature is key in further optimisation of the plant.

More formally, the problem to solve is: Given a data set of N estimated values for some $\hat{z}^*$, and considering some variables in the model $v \subset [x,u,z]$ as inputs, build a regression model $z^* = p(\beta, v)$ with $\beta \in \mathbb{R}$ regression coefficients, such that a measure $J(\hat{z}^*, \beta, z^*)$ of the fitness to data (e.g. L_1-regularised error or Least Squares error) is minimised over a set of constraints on the parameter space $\beta \in \wp$, on the inputs $v \in \mathfrak{V}$ and on the model response $c(\beta, v) \geq 0$.

$$\min_{\beta} \; J \coloneqq \sum_{i=1}^{N} \|\hat{z}^* - p(\beta, v)\|_L \qquad (3)$$
$$\text{s.t.:}\; \Omega \coloneqq \{\beta \in \wp | c(\beta, v) \geq 0 \; \forall \; v \in \mho\}$$

Two recently developed approaches, which are able to tackle the resolution of this constrained regression, are proposed here. The first one, implemented in the algebraic modelling environment ALAMO (Wilson & Sahinidis, 2017) and adopted in the modelling and simulation software gPROMS (Barton & Pantelides, 1993), uses mixed-integer programming (MIP) and global optimisation algorithms to automatically select among a set of user-provided potential basis functions, a linear combination of those that provide the best fit taking into account such extra constraints to guarantee physical coherence. These modelling environments offer a good support for this task, though the usually nonconvex problems to be solved and the adaptive-sampling procedures required to ensure constraint satisfaction are computationally demanding, even in the case where the MIP problem is restricted to be linear in decision variables β.

Instead of the "ALAMO approach", we also proposed an alternative way to handle the constrained-regression problem via sum-of-squares (SOS) programming (Pitarch, Sala, & de Prada, A systematic grey-box modeling methodology via data reconciliation and SOS constrained regression, 2019), (Pitarch, Sala, & de Prada, A Sum-Of-Squares Constrained Regression Approach for Process Modeling, 2019). In this approach, the potential set of basis functions for regression are limited to be polynomial, which may be a limitation. However, the resulting optimisation problem is convex, that is a clear advantage, and the extra constraints on the model response $c(\beta, v) \geq 0$ are naturally introduced and enforced with full guarantee of satisfaction within a desired input-output region, no matter how many samples are to be fitted or which region was covered by the experiments. This is a nice feature of the SOS approach, as performing experiments in industrial plants is often restricted and the amount of available data in plant historians is usually recorded with the plant around the same operating points so, in the end, the measured data is quite limited in process information.

In summary, with these proposed approaches, nonlinear and/or high-order polynomial regressors can be used with guarantees of well-behaved resulting function approximators, compared to most machine-learning options in prior literature. Section 4.5.2 present the application of these ideas to get suitable models for optimisation purposes in the Lenzing case study.

4.3 Generating data-driven models for complex unit operations

To determine whether data-driven modelling approaches are able to generate sufficiently accurate models of complex unit operations in chemical process modelling, PSE has done a meta-modelling (or "surrogate modelling") studyTable 3 Estimation pipeline for PLS fitting based on the original inputs to investigate whether data-driven models can be used to fit complex first-principles unit operation models over a representative operating range to a significant degree of accuracy. For this study, two complex unit operation models were selected:

1. Solid-Oxide Fuel Cell model
2. Olefins cracking furnace model

In both cases the Global System Analysis (GSA) functionality in PSE gPROMS process modelling software was used to perform a Monte-Carlo type simulation to generate solutions to the model for a number of input parameters. The input parameters on a uniform grid within a hypercube determined from the likely operation range for each individual input. Note that the input parameters are all related to operation and not to the design of the unit.

4.3.1 Solid-Oxide Fuel Cell model

Solid-Oxide Fuel Cells can be used to produce electricity from syngas by oxidizing this (Badwal, 2014). These fuel cells are often used in stationary applications and operate a high temperature. PSE has a high-fidelity model of this type of fuel cells as part of its gFUELCELL product. The fuel cell was included in a flowsheet with inflow and outflow streams for air and fuel (see Figure 20). The factors and responses are shown in Table 4 and Table 5 respectively. An overview of the GSA simulation key characteristics is shown in Table 6.

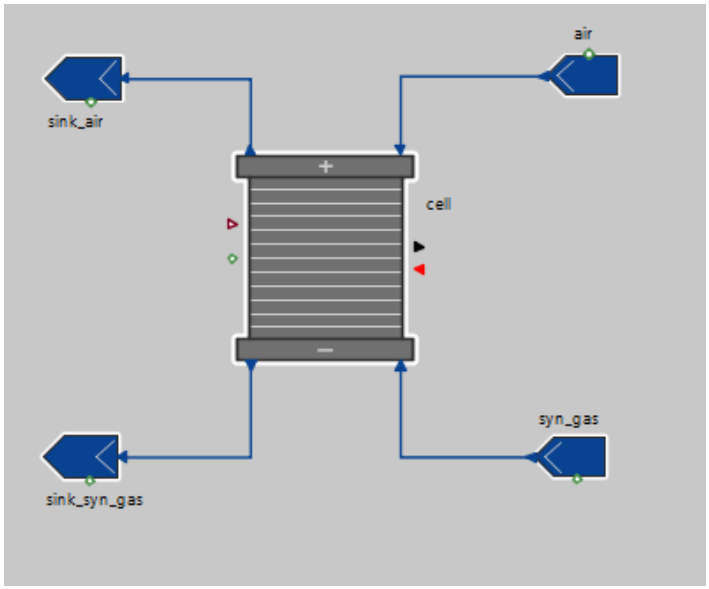


Figure 20 Flowsheet with the gPROMS model of a Solid-Oxide fuel cell

Table 4 Factors used for GSA simulation of the SOFC

Factor	Unit	Lower bound	Upper bound	Distribution
Air flow rate	Mol/s	0.004	0.04	Uniform
Air pressure	bar	1	2	Uniform
Air Temperature	dC	650	800	Uniform
Current density	A/cm2	0	0.75	Uniform
Syngas molar fractions [CO,H_2,H_2O]	Mol %	[20,20,3]	[80,80,20]	Uniform
Syngas flowrate	Mol/s	0	0.001125	Uniform
Syngas pressure	bar	1	2	Uniform

Table 5 Responses used for GSA simulation of the SOFC

Response	Unit
Cathode molar fractions [O_2,N_2,H_2O]	mol %
Cathode temperature	dC

Anode molar fractions [CO,CO_2,H_2,H_2O]	mol %
Anode temperature	dc

Table 6 GSA simulation characteristics for SOFC model

Global System Analysis (GSA) simulation results	
Samples requested	44019
Samples failed	0
Samples succeeded	44019
CPU time	22.5 hr

Model generation was conducted with cross validation in accordance with the procedure described in (K.M.Nauta, COPRO project partners, 2018). Both a linear PLS regression using the original inputs was performed as well as a PLS regression using a feature space generated using both polynomial and reciprocal (ratio) transformations. The estimation pipelines in scikit-learn for both are given in Table 7 and Table 8 respectively.

The results are shown in terms of validation curves, based on 10 fold CV, in Figure 21. This figure shows that, since there is no noise on this data, higher model orders give better results. For the linear case the Q2 results closely match the R2, which is expected given the large amount of data in each fold. For the nonlinear case, the average Q2 tracks the R2 fairly closely but the minimal Q2 for all fold can be significantly worse. This is likely because of the reciprocal transform, which can show large gradients and errors close to 0. The average Q2 is significantly higher (0.998) for the nonlinear feature space compared to that of the linear feature space (0.776). This is also reflected in the scatter plots for the key response variables, where for the linear feature space most samples are outside of a 5% relative error margin, see Figure 22, and for the nonlinear one they are inside, see Figure 23.

In summary, while the non-linear

Table 7 Estimation pipeline for PLS fitting based on the original inputs

Data-processing and model fitting pipeline for fitting with linear feature space		
1	StandardScalar	Scaling of each variable with the max-min range
2	PLSRegression	Partial Least Squares fitting

Table 8 Estimation pipeline for PLS fitting based on a transformed feature space

Data-processing and model fitting pipeline for fitting with non-linear feature space		
1	StandardScalar	Scaling of each variable with the max-min range
2	ReciprocalTransform	Adding of 1/x to feature space for each input
3	PolynomialFeatures	Adding of polynomial terms up to order 2 to the feature

		space
4	PLSRegression	Partial Least Squares fitting

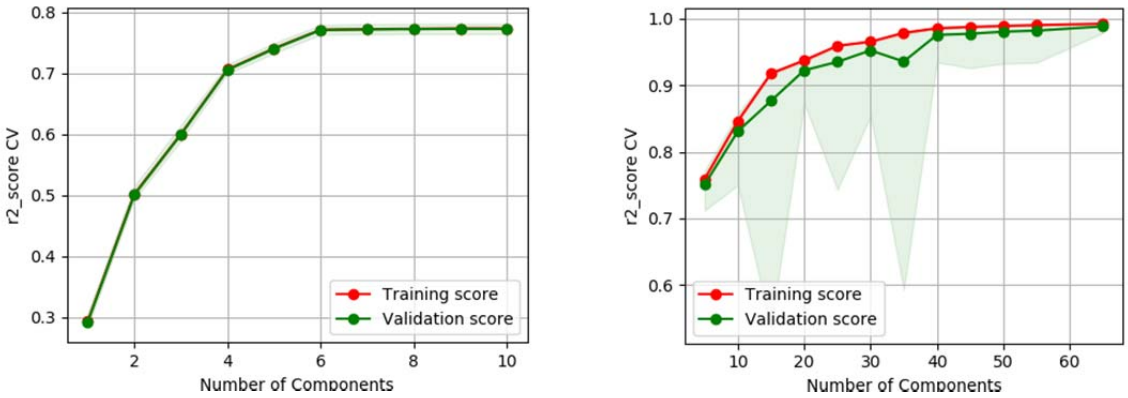


Figure 21 Comparison of validation curves with R2/Q2 scores for PLS model fitting based on original inputs (left) and on transformed feature space (right) for SOFC surrogate model. The red line is the training (R2) score, the green line the validation (Q2) score. The green band is generated from the maximal and minimal cross validation Q2 scores.

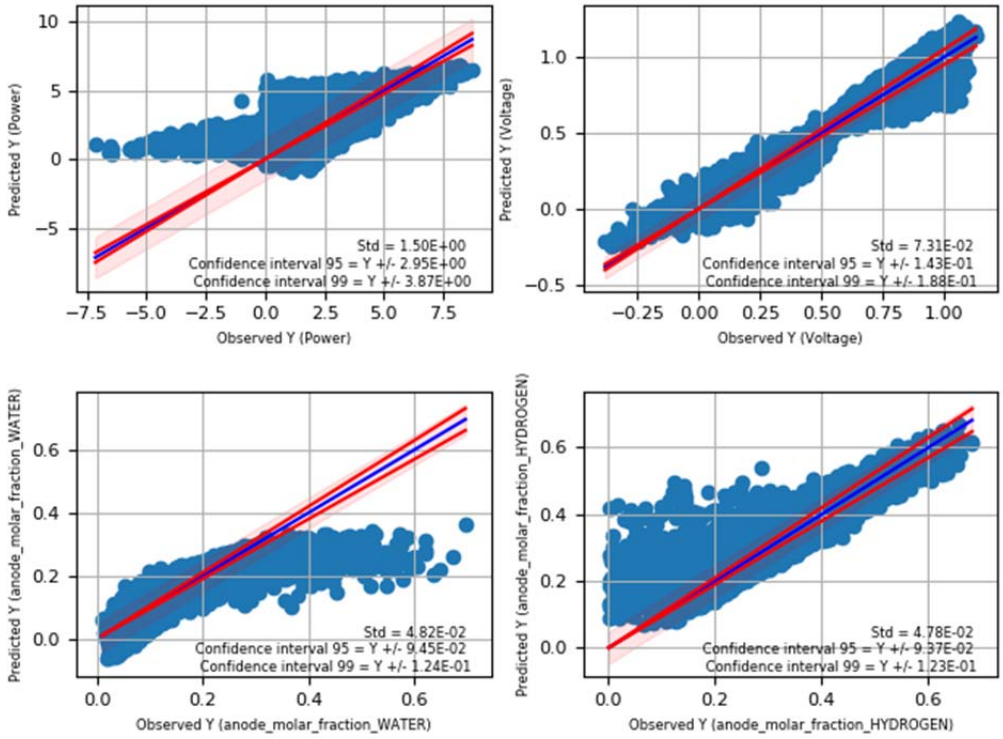


Figure 22 Scatter plot for PLS model fitted to original inputs for key response variables (Power, Voltage, Anode water fraction, Anode hydrogen fraction)

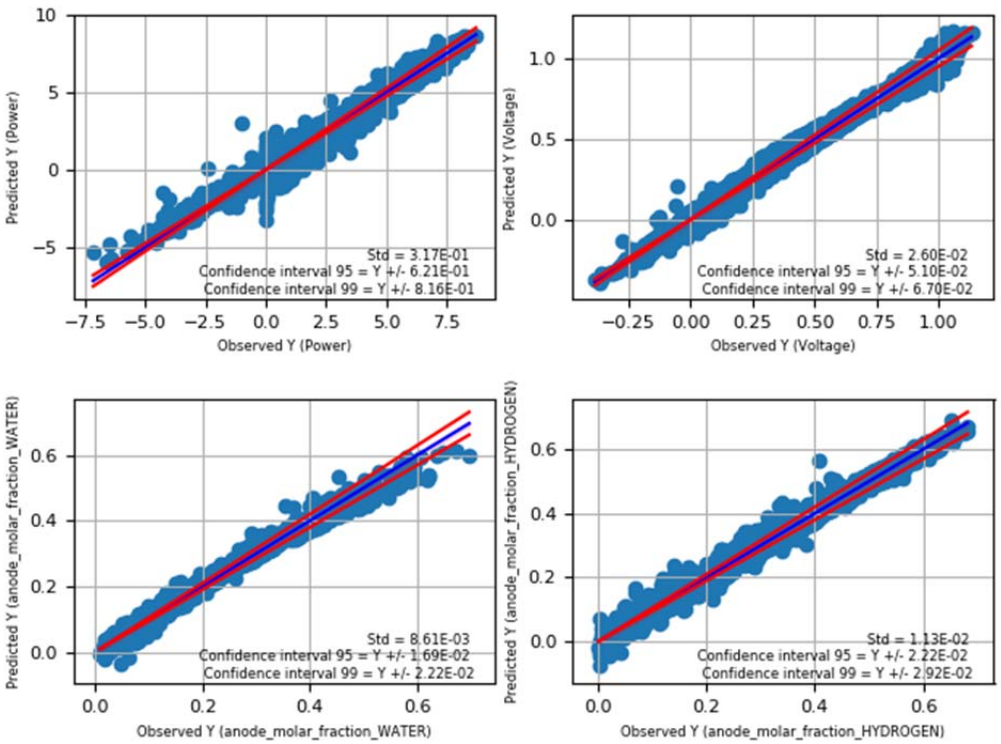


Figure 23 Scatter plot for PLS model fitted to nonlinear feature space for key response variables (Power, Voltage, Anode water fraction, Anode hydrogen fraction)

4.3.2 Cracking furnace

In the framework of this project PSE has chosen to use an olefins cracking furnace case study based on a standard example in the gPROMS ProcessBuilder product to get an indication of the potential of hybrid modelling for the INEOS cracker use case. Due to reasons of confidentiality of kinetic models and the re-modelling effort, the INEOS use case cracker was not used directly.

PSE has a detailed model library for ethylene processes, “gML Olefins”. This library includes a first-principles model of a cracking furnace. The model predicts the cracked gas composition and the coil outlet temperatures of the furnace given feedstock composition and flowrate and the furnace operating conditions. This is achieved by using detailed calculations of the cracking kinetics along the length tubes of the furnace as well as calculations to determine the temperature along the tube.

The complexity of the model depends on the chosen number of discretisation points along the length of the cracking tubes. For a typical choice of this number of points, the model includes around 16K variables and equations and it takes around 30 sec to find a first solution on i7 processor.

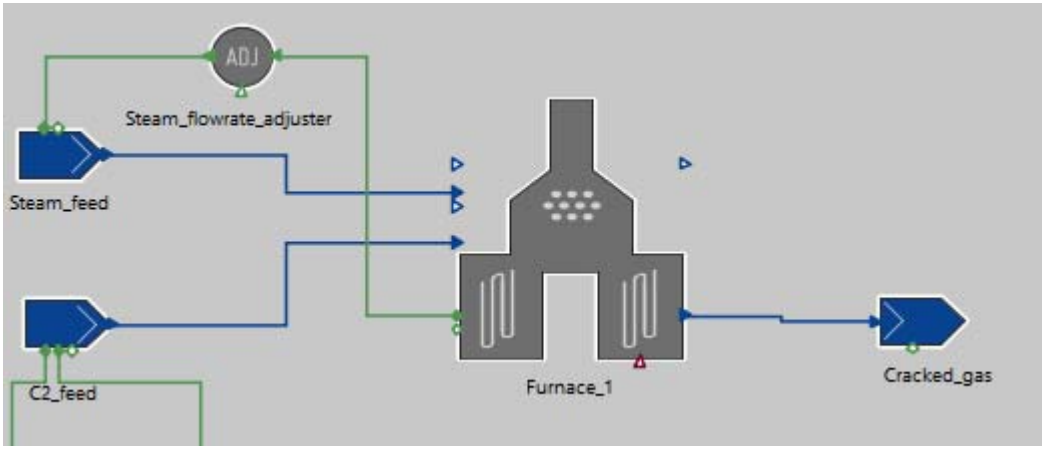


Figure 24 First principles model of Olefins furnace as used in the gML Olefins product

Table 9 Factors used for GSA simulation of furnace model

Factor	Unit	Lower bound	Upper bound	Distribution
CIT	dC	525	600	Uniform
Conversion	M	1.00E-13	0.008	Uniform
Conversion_PROPANE	%	75	95	Uniform
Coil outlet pressure (COP)	bar	1.65	2.25	Uniform
Feedstock mass flowrate	kg/s	9.7	20	Uniform
Ethane mass fraction in	kg/kg	0	0.5	Uniform
Propylene mass fraction in	kg/kg	0	0.001	Uniform

Table 10 Responses used for GSA simulation of furnace model

Response	Unit
Coil inlet pressure (CIP)	bar
Tube metal temperature (TMT)	dC
COT	dC
Mass fractions [11 components]	kg/kg

Table 11 GSA simulation characteristics

Global System Analysis (GSA) simulation results	
Samples requested	15000
Samples failed	923
Samples succeeded	15000
CPU time	8447 sec

For this example both linear PLS estimation was done, see Table 12, as well as a PLS estimation based on non-linear feature space with polynomial terms, see Table 13. The results, shown in Figure 25, indicate that both linear input space and nonlinear feature space based models validate very well due to abundance of data and lack of noise. With a nonlinear feature space higher Q2 scores can be obtained.

Table 12 Estimation pipeline for PLS fitting based on the original inputs

Data-processing and model fitting pipeline for fitting with linear feature space		
1	StandardScalar	Scaling of each variable with the max-min range
2	PLSRegression	Partial Least Squares fitting

Table 13 Estimation pipeline for PLS fitting based on a transformed feature space

Data-processing and model fitting pipeline for fitting with non-linear feature space		
1	StandardScalar	Scaling of each variable with the max-min range
3	PolynomialFeatures	Adding of polynomial terms up to order 2 to the feature space
4	PLSRegression	Partial Least Squares fitting

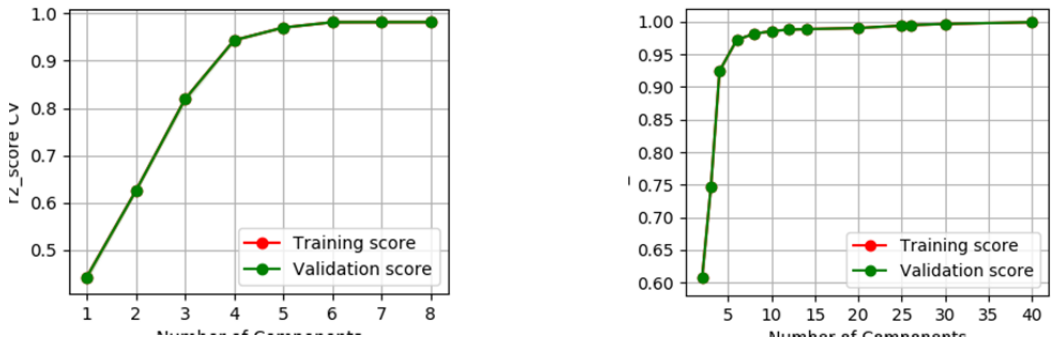


Figure 25 Comparison of validation curves with R2/Q2 scores for PLS model fitting based on original inputs (left) and on transformed feature space (right) for cracking furnace surrogate model. The red line is the training (R2) score, the green line the validation (Q2) score. The green band is generated from the maximal and minimal cross validation Q2 scores.

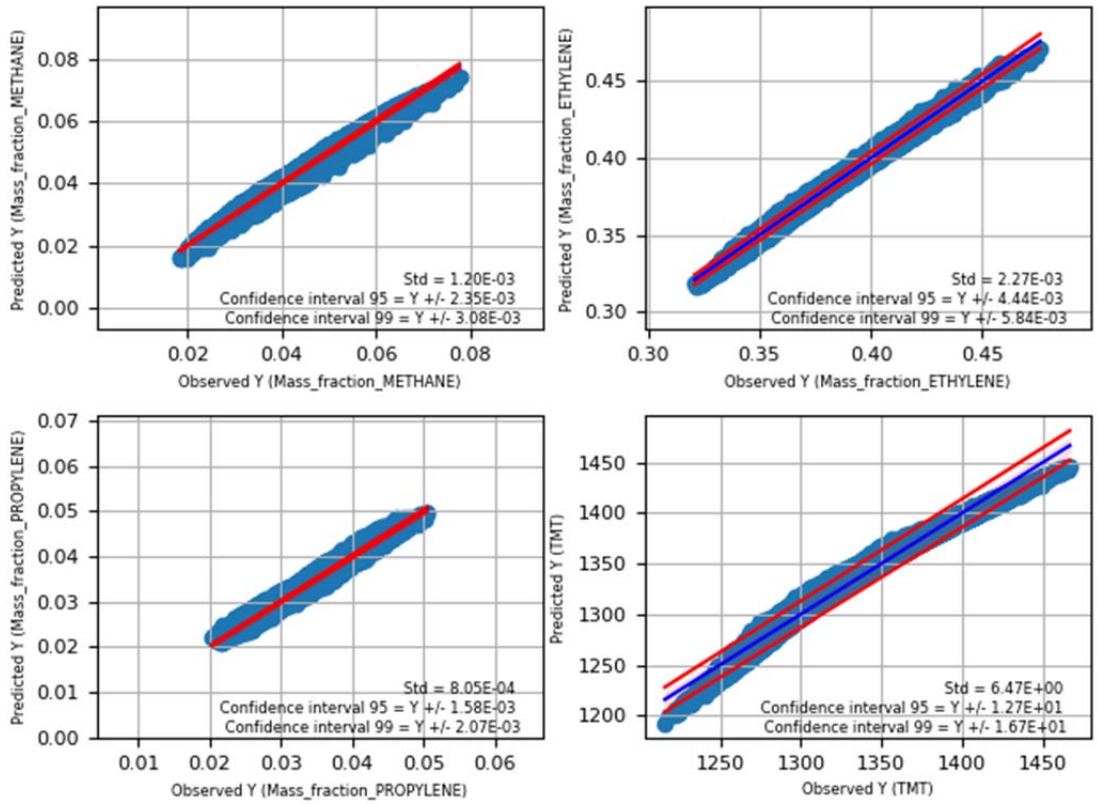


Figure 26 Scatter plots for key variables for the cracking furnace PLS fit using the original input space

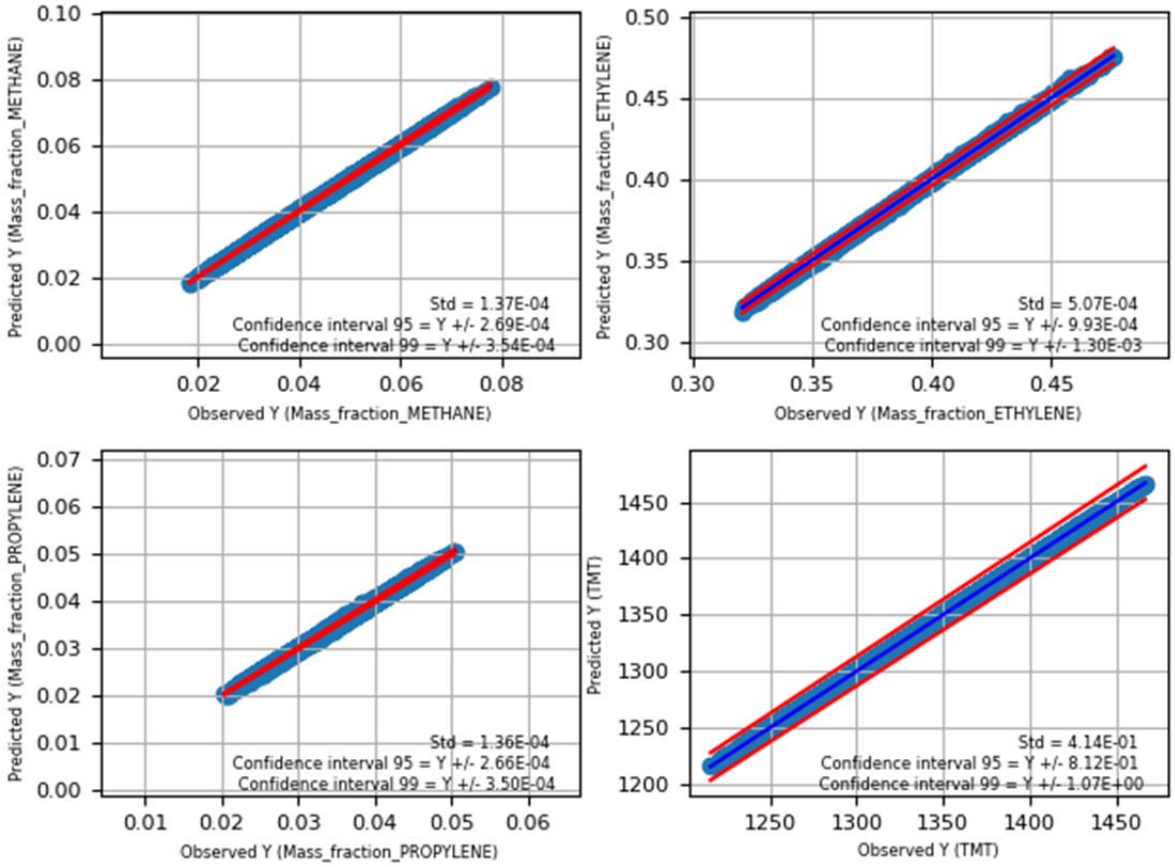


Figure 27 Scatter plots for key variables for the cracking furnace PLS fit using the nonlinear feature space

4.4 Hybrid modelling tool

Within the framework of the COPRO project, PSE has developed a prototype hybrid modelling capability to be used in conjunction and as part of its gPROMS process modelling platform. The gPROMS process modelling platform (Barton & Pantelides, 1993) permits the development of models from first-principles, both using a custom modelling language and well as using drag and drop flowsheeting. The intention of this tool is to allow user to select and compare first-principles modelling, data-driven modelling, or a combination of both (hybrid modelling) for any application.

This tool consists of a number of components. These components are develop to various Technology Readiness Levels (TRL's) . The main reason for this is that in particular for data pre-processing the requirements for functionality and hence UI design are not clear and require more application experience. For the fitting the requirements are quite clear and for the prediction they are clear as well, hence an implementation in C++ for a higher TRL is used. A detailed functional overview of this tool will be given in COPRO deliverable D1.2, here just a outline of the tool is given, in terms of its individual components.

4.4.1 Components

4.4.1.1 Prototype python wrapper for derivation of data-driven and hybrid models using the sk-learn toolbox

The scikit-learn toolbox in Python is a commonly used toolbox for machine learning that implements a number of fitting algorithms and classes for setting up pipelines for data-processing and fitting. Based on this toolkit a number of classes have been set up that streamline the following workflow:

- Data import from HDF5 and CSV
- Setting ranges for operating variables and scaling data
- Performing a fit using linear PLS regression with on nonlinear transformations of the original input space
- Analyzing results using standard plots (scatter plots, learning curve, validation curve)
- Selecting the optimal value of hyper-parameters based on validation statistics
- Writing the resulting model to an XML format for data-driven models

4.4.1.2 Tool for derivation of data-driven models using the ALAMO algorithm

The ALAMO algorithm (Wilson & Sahinidis, 2017) was identified as a promising algorithm to productize as part of the hybrid modeling tool. A prototype tool has been developed in C++ and tested on a range of examples included those in the original ALAMO paper. Currently work is under way to productise this tool and integrate it into the gPROMS product code-base. The tool uses the Xpress Solver Engine as a MIQP solver. It supports commonly used basis functions and, in line with the original ALAMO paper, uses the AIC criterion as a model fitness metric to select the model complexity. It can be configured using a .json configuration file. It also writes the resulting model to an XML format for data-driven models.

4.4.1.3 Foreign Object (FO) for prediction using data-based models in gPROMS ProcessBuilder

When a data-driven model has been derived and serialized in the DataBasedModel XML format using either the python wrapper or the prototype tool based on the ALAMO algorithm, it can be used in

the gPROMS kernel. To facilitate this, a gPROMS Foreign Object (plug-in function) has been developed and coded in C++. This function takes inputs from the kernel (the inputs to the data-driven model) and returns outputs and their derivatives from the data-driven model.

4.4.1.4 Prototype flowsheeting library for data-driven and hybrid modelling in gPROMS ProcessBuilder

When a purely data-driven or a hybrid model model has been generated to represent a particular unit operation its important that this model can be integrated rapidly in a gPROMS flowsheet. Therefore a library of unit models have been developed that facilitate the inclusion of data-driven models in first-principle model flowsheet using drag and drop flowsheeting.

Model name	Description
General_model_data_based_gML	General model that can represent a unit operation, plant section or plant. Any number of process streams can enter or leave the unit. No mass balance constraint is enforced. All outlet streams are predicted from a data-based model.
Calc_data_based_gML	General calculation from a data-based model. This model can be connected to other models to select internal variables from these models and perform a calculation on them.
General_model_mass_balance_data_based_gML	General model that can represent a unit operation, plant section or plant. Any number of process streams can enter or leave the unit. Mass balance constraints are enforced. A data-based model is used to predict conversion and split factors.
Heat_exchanger_data_based_gML	Heat exchanger model. Mass and energy balances are enforced. The heat transfer “UA” value can be predicted using a data-based model.
Conversion_reactor_data_based_gML	Model that can respresent a reactor unit operation as a conversion reactor whereby the conversion is predicted using a data-based model.
Component_splitter_data_based_gML	Model that can represent a separation unit operation as a conversion reactor whereby the split factor for each component is predicted using a data-based model.
Source_data_based_gML	Model that represents a process steam source. Stream quantities (fractions, temperature, etc) can be calculated using a data-based model.

Splitter_data_based_gML	Model that represent a stream splitter. Overall split fraction can be calculated using a data-based model.

4.5 Application examples

4.5.1 Hybrid ethylene plant model optimisation

As part of the examples included with PSE's gML Olefins library in gPROMS ProcessBuilder, a simplified ethylene plant is included. This plant includes 7 furnaces, a simplified model of the quench section, first-principles models of the compression section and simplified models for the different stages of the separation section. The plant model features recycle streams for both ethane and propane.

The furnaces in this plant are all assumed to be identical. Four furnaces are operated with a predominantly ethane feed and the three remaining ones with a propane feed.

The furnaces are assumed to all have a different coking layer thickness (1-7mm) to reflect the fact that they are in different stages after a de-coke. The coking layer is assumed to be of uniform thickness along the coil length.

To predict material properties the "RKS (Advanced)" equation of state was used.

This model consists of around 115K variables and equations and it takes around 350 sec to find a first solution on an i7 processor.

Using this model the operation of the ethylene plant can be optimised. A simplified optimisation problem has been defined in which the product and feedstock prices are used to calculate an approximation of the plant profit. The COT settings for each furnace need to be optimised as well as the feed flows to each furnace.

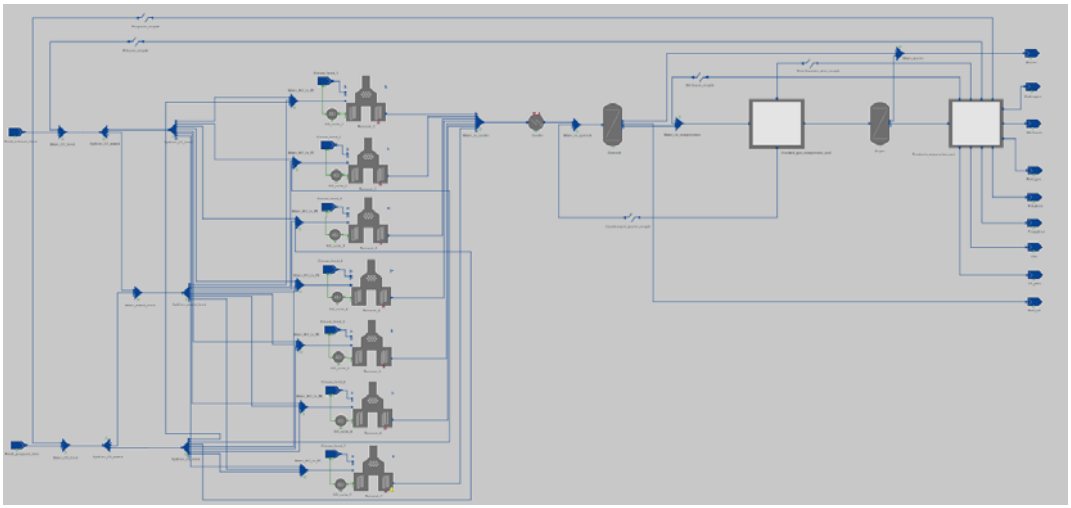


Figure 28 gPROMS flowsheet for an olefins plant with 7 furnaces

The PLS model for the furnace, derived in 4.3.2, was used to replace the first-principles furnace models in the ethylene plant flowsheet. This was done using the gPROMS hybrid modelling tool, with the "Unit_operation_data_driven_gML" model replacing the "Furnace_gML" first-principles model (see Figure 29, Figure 30). This reduces the number of equations in the model substantially, see Table

14. The results for the optimisation are compared in Table 15. What is clear is that while the prediction from the hybrid model at the optimal point is fairly close to that of the original model, there is still a non-negligible difference in particular with relation to the maximum load constraints. This will have to be investigated further.

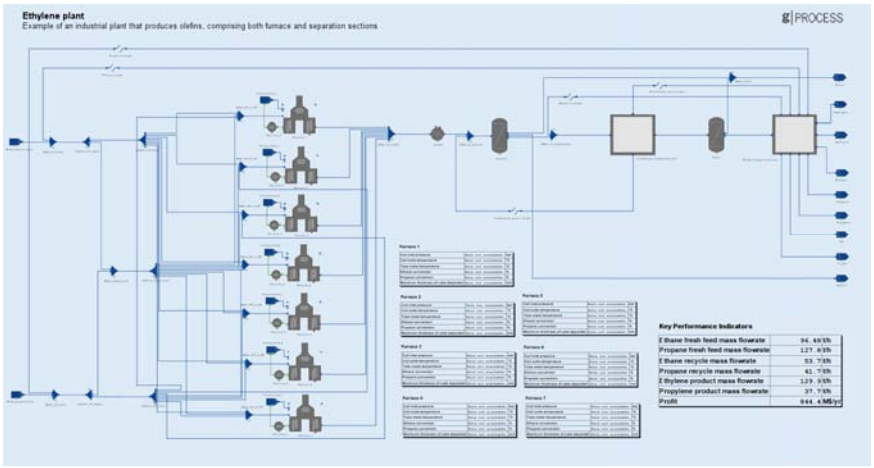


Figure 29 First-principles Olefins plant model

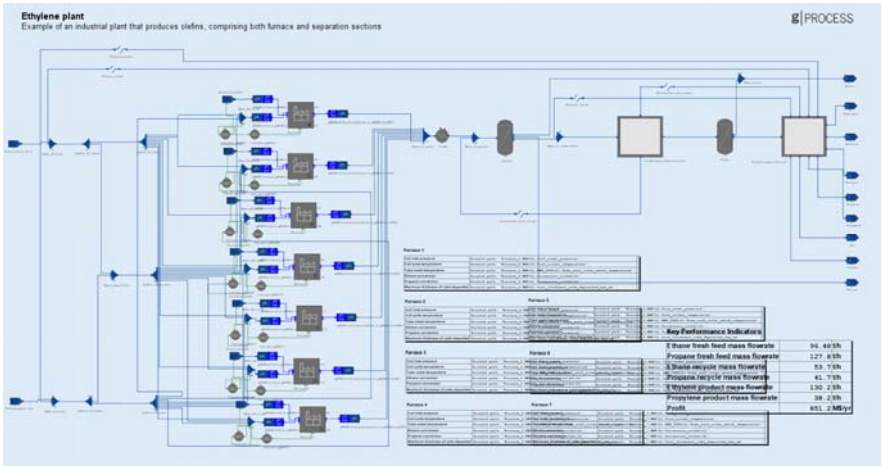


Figure 30 Hybrid Olefins plant model with the cracking furnace modelled using PSE's hybrid modelling toolbox

Table 14 Comparison of model complexity for first-principles and hybrid model

Parameter	**First-principles model**	**Hybrid model**
Number of equations	112,865	9,132

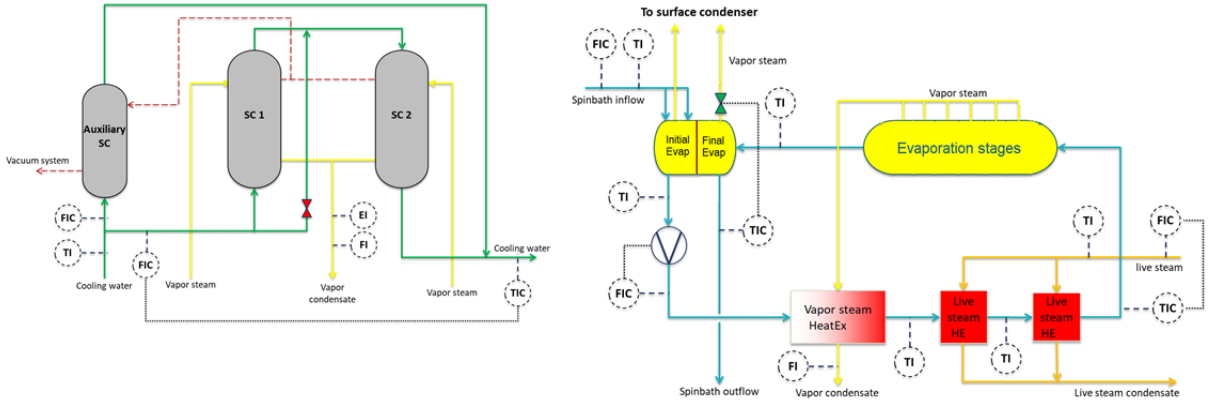
Table 15 Comparison of optimisation results for first-principles and hybrid model

Parameter	**Unit**	**Initial point**	**First-principles model**	**Hybrid model**	**First-principles model from hybrid model solution**
Profit	M$/yr	847.981	844.406	851.190	845.1043
Maximum	%	4.8%	<0.01%	<0.01%	1%

constraint violation (furnace load)					

4.5.2 Modelling the surface-condensers performance

The evaporation plants in Lenzing AG have one main source of energy consumption, live steam from boilers, which is used to heat the spinbath up to a set point to start a partial evaporation of water. The specific-steam consumption (SSC) in a plant is strongly dependent on the performance of the cooling system (the secondary evaporation effect). Some of the plants are equipped with cooling towers, that run independently, but others are cooled with river water through surface condensers (see Figure 31). Thus, the more cooling water is send to the surface condenser, the less specific-steam consumption is achieved in the plant. However, the operation of such condensers is not independent, as the water that is taken from the river is a limited and shared resource through a distribution network.



a) Surface-condenser system b) Multi-effect evaporation plant

Figure 31. Spinbath evaporation plant with surface condenser.

Hence, with the aim of optimising the water distribution among plants, the effect of the operation of the surface condensers (SC) in the steam consumption is studied.

In a rigorous way, in order to predict the change in the SSC due to a variation in the cooling water through, we would need to include the complete nonlinear model of the whole plant. Since this could be possible for one plant, it would be impractical for the aim of optimising simultaneously the whole network due to the excessive model complexity. Therefore, some experiments have been performed with the SCs onsite and, from them, experimental surrogate models are developed to predict the variation on the steam consumption according to the performance of the SCs.

4.5.2.1 Designed experiments

Lenzing AG has collected data from all plants by the following tests:

- **Outlet water temperature versus water flow through the SCs.**

With fixed evaporation capacity (plant load), records of temperatures have been collected at different water flows to the SC, covering the usual operation range. Therefore, low-order polynomial curves can be fitted to the raw data as shown in Figure 32.

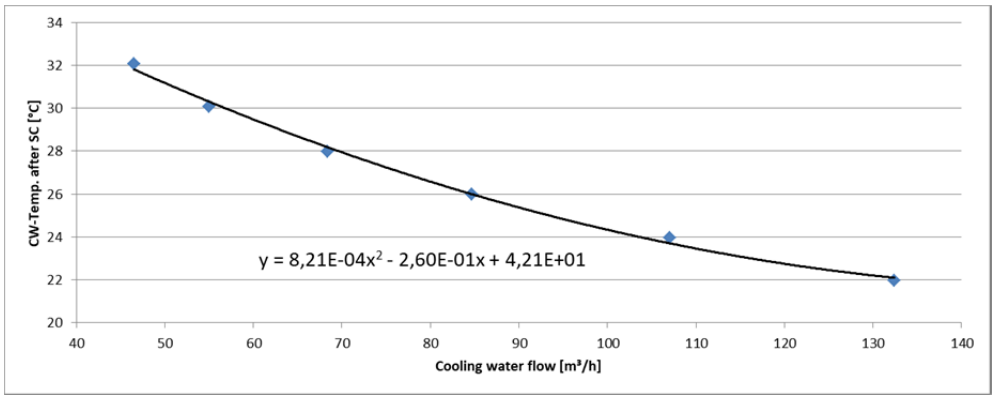


Figure 32. Water outlet temperature VS cooling water flow.

- **Specific-steam consumption versus cooling capacity.**

From the measurements on the water side (inlet/outlet temperatures and volumetric water flow), the actual cooling capacity in the SC can be computed by the formula:

$$C_{pow} = 4.18 \cdot F \cdot \frac{T_{out} - T_{in}}{3600} \quad (4)$$

Hence, by recording the live steam consumption of the evaporation plant in the above test, we can depict the specific steam consumption (SSC) versus the available cooling power in the SC system and fit a model for it too. See Figure 33.

The above relationships will allow us to set a model relating the variation of the SSC in the plant due to the cooling water flow through the SC.

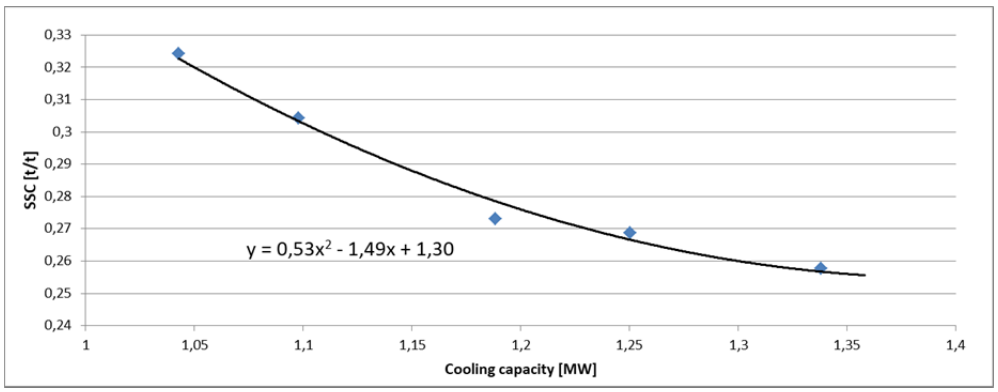


Figure 33. SSC VS cooling power.

- **Conductivity versus cooling water flow.**

Conductivity in the steam side of the SC is an indicator of the so-called spinbath entrainment effect. This unwanted behaviour needs to be avoided to keep the materials life. Lenzing AG has also recorded the conductivity values for the test performed in the plants. The results are conductivity values for different water flows, similar to Figure 34.

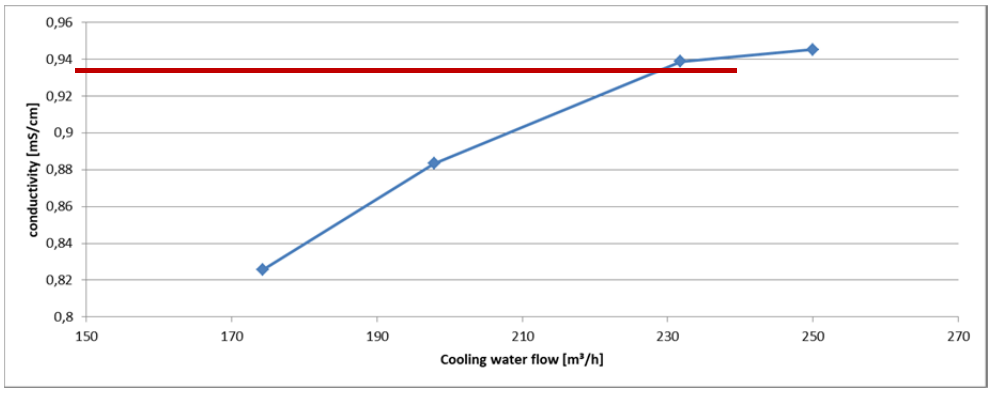


Figure 34. Conductivity VS cooling water flow.

From these tests we can define new limits for the maximum water flow to the SC permitted before starting with spinbath entraintment (when conductivity approaches 1 mS/cm).

4.5.3 Type of models

In order to adjust a model for predicting the cooling water outlet temperature w.r.t. the water flow through the SC, $\Delta T \coloneqq T_{out} - T_{in} = f(F_{cw})$, we propose a polynomial model with candidate monomial basis functions up to degree 3:

$$\Delta T \coloneqq T_{out} - T_{in} = \beta_0 + \beta_1 F_{cw} + \beta_2 F_{cw}^2 + \beta_3 F_{cw}^3 \tag{5}$$

In this way, given a measured water inlet temperature T_{in}^{m}, the outlet temperature T_{out} could be computed by (5). Now, assuming the tests were carried out with the SCs fully clean, the fouling effect will provoke that measurements of actual outlet temperatures T_{out}^{m} lie below the above *best demonstrated efficiency* curve (reduced heat transfer due to fouling), as shown in Figure 35. Thus, a bias parameter K_f is added to (5) in order to adjust the model in real time with each measurement T_{out}^{m}:

$$T_{out} = \Delta T + T_{in}^{\mathrm{m}} - K_f, \quad K_f = \Delta T + T_{in}^{\mathrm{m}} - T_{out}^{\mathrm{m}} \tag{6}$$

In this way, the current state of fouling in the SC is taken into account in the model.

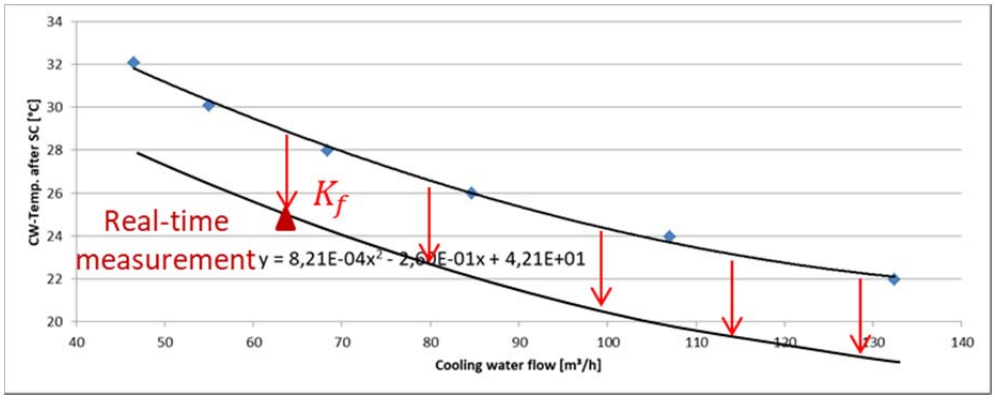


Figure 35. Adapting the model to the current fouling state.

Moreover, this approach allows to isolate the effects by the fouling in the SC system from the effects by the fouling in the spinbath heating line, which also affect the overall SSC.

Analogous to the above temperature model, we set a polynomial candidate model $\Delta\mathrm{SSC} = g(C_{pow})$ to predict the variation of the SSC when the cooling capacity in the SC varies:

$$\Delta\text{SSC} \coloneqq \text{SSC} - K_B = \beta_4 + \beta_5 C_{pow} + \beta_6 C_{pow}^2 + \beta_7 C_{pow}^3 \tag{7}$$

Where C_{pow} is computed by (4) and K_B is a bias required to remove the dependency on the operating point (load) from the experimental data for regression. To do so, the simplest idea is to compute the *best specific steam consumption*, $K_B = \text{Best SSC}$, as shown in Figure 36. Such Best SSC value can be taken as the lowest SSC recorded in the plant tests. For that to be true, two assumptions are made: 1) the tests were carried out with clean SC and 2) the model for ΔSSC (i.e., the shape of the curve in Figure 36) does not vary significantly from one operation point to another (evaporation loads).

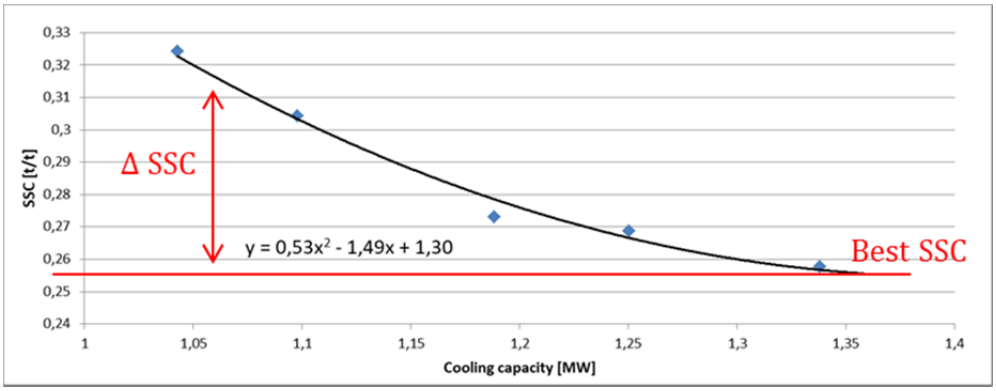


Figure 36. Normalization w.r.t the best possible operation.

Finally, the increment on the plant absolute steam consumption due to the loss of efficiency in the SC system can be trivially computed with the current evaporation load E_F, by: $\Delta\text{SC} = \Delta\text{SSC} \cdot E_F$

4.5.3.1 Models fitting

Now it only remains fitting the parameters β from (5) and (7) to the data recorded form the experimental tests. Although this task may look trivial looking at the previous figures, using a standard regression technique like regularised LS implies serious risks, as we are going to demonstrate.

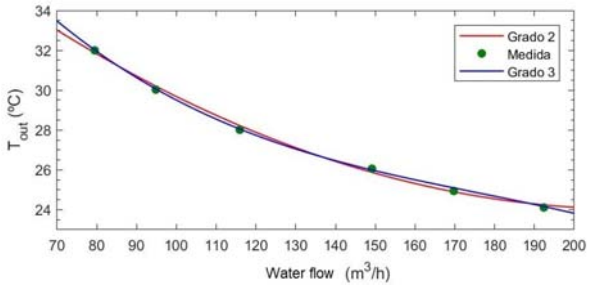
The decision variables for optimizing the water distribution in the evaporation network are the cooling-water flows F_{cw} send to each SC, details in (Marcos, Pitarch, de Prada, & Jasch, 2018). Thus, the optimiser will use the combination of (4), (5) and (7) to predict the variation of the SSC achieved with a proposed value for the flow:

$$\Delta\text{SSC} = \sum_{i=0}^{3} \beta_{i+4} \cdot \left(\frac{1.18}{3600} F_{cw} \cdot \left(\sum_{j=0}^{3} \beta_j F_{cw}^j + T_{in}^{\text{m}} - K_f \right) \right)^i \tag{8}$$

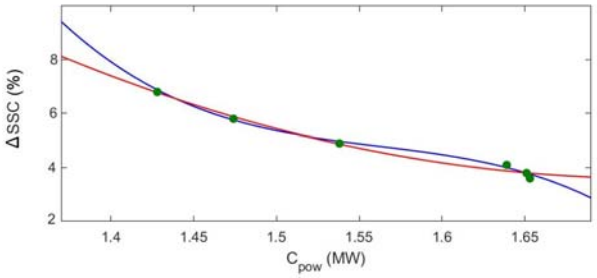
Note that, although low-degree polynomials were fit independently for (5) and (7) with negligible fitness deviations to data, model (8) is a much higher degree polynomial in F_{cw} (note that if $\deg(5) \leq \phi$ and $\deg(7) \leq \gamma$, $\deg(8)$ is less or equal than $\gamma \cdot (\phi + 1)$). Consequently, little deviations due to noise in the experimental data may derive in incoherent and unexpected responses of (8).

For example, if standard LS with regularization (Neumaier, 1998) is used to fit (5) and (7) to the data independently, Figure 37 shows that a better *local* fit (blue curves in figures a,b) results in an unacceptable response of (8), depicted in Figure 37d. In particular, the region highlighted in a dashed box shows how the cooling capacity reduces at high cooling-water flows, with the

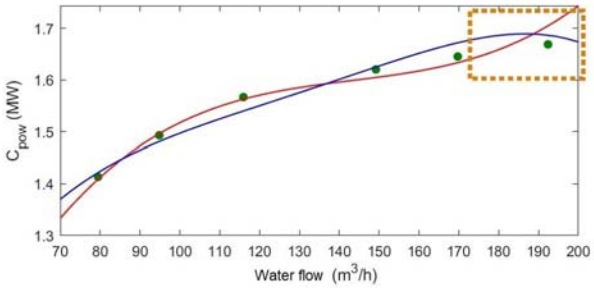
corresponding increase in the SSC. That behavior is impossible by the process physics and, in addition, this model would create a *false local minimum* in the optimisation.



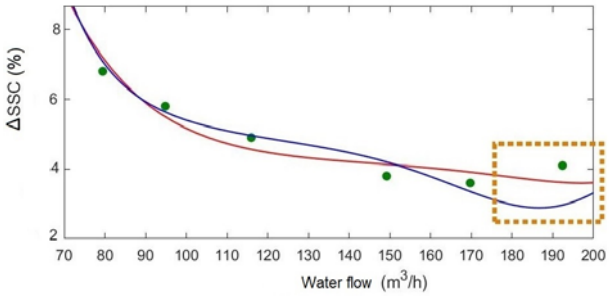
a) Model for the outlet temperature



b) Model for the SSC variation.



c) Cooling power predicted VS water flow



d) SSC predicted VS water flow.

Figure 37. Comparison of models achieved by regularised LS (blue) and by the ALAMO approach (red).

Remark. Note that this unexpected behavior wouldn't be easily detected because the raw experimental data to fit the models in Figure 37a and Figure 37b do not show outliers. However, the point inside the highlighted dashed box resulted to be an outlier when depicted in Figure 37d.

Here comes the need of using *constrained regression* to force additional constraints on the model response (monotonic behaviour in this case). As (8) is nonlinear in decision variables β, we employed the "ALAMO approach" in this case, summarised next:

1. **Model fit.** Given a set of N datapoints for $\{\Delta T, \Delta\text{SSC}, C_{pow}, F_{cw}\}$, suitably centered, scaled and ordered from lowest to highest flow F_{cw}, find the best fit for (8) solving the following mixed-integer nonlinear optimisation problem, where the objective function balances the fitness to data with the model complexity (limited by D by the Akaike information criterion):

$$\min_{\substack{\beta, D \in \mathbb{R} \\ y \in \{0,1\}}} \; N \log\left(\frac{\|T_{out}^{\mathrm{m}} - T_{out}(\beta, F_{cw})\|_2^2 + \|\Delta\text{SSC}^{\mathrm{m}} - \Delta\text{SSC}(\beta, F_{cw})\|_2^2}{N}\right) + 2D + \frac{2D(D+1)}{N-D-1} \tag{9}$$

$$\text{s.t.:} \quad y_1 + y_2 + y_3 + y_4 \leq D \tag{10}$$

$$y_1\beta^{\mathrm{L}} \leq \beta_2 \leq y_1\beta^{\mathrm{U}}, \;\; y_2\beta^{\mathrm{L}} \leq \beta_3 \leq y_2\beta^{\mathrm{U}}, \;\; y_3\beta^{\mathrm{L}} \leq \beta_6 \leq y_3\beta^{\mathrm{U}}, \;\; y_4\beta^{\mathrm{L}} \leq \beta_7 \leq y_4\beta^{\mathrm{U}} \tag{11}$$

$$T_{out}\left(\beta, F_{cw}^{[i+1]}\right) - T_{out}\left(\beta, F_{cw}^{[i]}\right) < 0 \;\forall\, i \in \{1, \dots, N-1\} \tag{12}$$

$$\Delta\text{SSC}\left(\beta, F_{cw}^{[i+1]}\right) - \Delta SSC\left(\beta, F_{cw}^{[i]}\right) < 0 \;\forall\, i \in \{1, \dots, N-1\} \tag{13}$$

Where notation $F_{cw}^{[i]}$ represents the value of the flow at sample i in the dataset.

2. **Constraint validation.** With the parameters β fixed to a value β^* got from the above fitting stage, check analytically the numerical conditions (12) and (13) solving the following nonlinear optimisation problems respectively:

$$\min_{F_{cw}} \; T_{out}(\beta^*, F_{cw}) \quad \text{s.t.}: \mathrm{F^L} \leq F_{cw} \leq \mathrm{F^U} \tag{14}$$

$$\min_{F_{cw}} \; \Delta\mathrm{SSC}(\beta^*, F_{cw}) \quad \text{s.t.}: \mathrm{F^L} \leq F_{cw} \leq \mathrm{F^U} \tag{15}$$

   Denote by F^* to the solution of any of the above two optimisation problems. Then, if $F^* \equiv \mathrm{F^U}$ for both, the obtained model is proven monotonic and both T_{out} and SSC decreases as F_{cw} increases, coherent with the process physics. Otherwise, add each point $F^* \neq \mathrm{F^U}$ found in this stage to the regression dataset and go back to Stage 1.

By applying this constrained-regression approach, suitable models for predicting the SC effect on the SSC have been obtained, see red curves in Figure 37. Indeed, the obtained models (5) and (7) are lower in complexity (2-degree polynomials) than those obtained by standard regularised LS (degree 3).

4.5.4 Modelling the heat-transfer in an evaporation plant

In this second case study, we make use of the proposed methodology to build up a grey-box model for the multiple-effect evaporation plant depicted in Figure 31b. The plant receives a spinbath input, mixture of water with chemical components and leftovers of organic material, and the goal is to concentrate the liquid by removing certain amount of water.

4.5.4.1 First-principles backbone and variables estimation

The modelling phase starts from a nonlinear set of equations of the plant in steady state, obtained by first principles. These equations have been omitted here for brevity, but the reader is referred to (Pitarch, Palacín, de Prada, Voglauer, & Seyfriedsberger, 2017), (Pitarch, Palacín, Merino, & de Prada, 2017), for a detailed description. Then, after suitable data pretreatment to remove out-of-range measurements, the *estimation* phase (Stage 2 of the proposed methodology) uses data reconciliation (2) to "clean" the process data from incoherent sensor values and to get suitable estimates for all algebraic variables and parameters. In particular we focus on the heat-transmission coefficient $UA(t)$, as this time-varying parameter includes the conduction and convection effects plus the exchange surface, values that are not precisely known in the lumped sets of heat exchangers.

Moreover, because an accurate modeling of the long-term fouling dynamics in the heat-exchangers pipes is key for a realistic optimisation of the operation as well as the right scheduling of the maintenance tasks (Pitarch, Palacín, de Prada, Voglauer, & Seyfriedsberger, 2017), (Pitarch, Palacín, Merino, & de Prada, 2017). Indeed, this issue is arises in other industrial systems like coking in cracking furnaces or catalyst deactivation in chemical reactors. All have in common a system-efficiency degradation, which may be palliated or worsened by the way the equipment is operated.

Thus, a set of experiments where performed on site to collect data, running the plant in different operating conditions, setting different values for the main control variables: the spinbath flow F and the temperature setpoint. Moreover, in order to get significant information from the actual fouling dynamics, the plant historian for several months of operation (including some stops for cleaning) has been also provided as experimental data (sampling performed each 5 min.) for reconciliation. Figure

38 shows the estimated $\widehat{UA}$ for the first set of lumped heat exchangers over 7 months of plant operation.

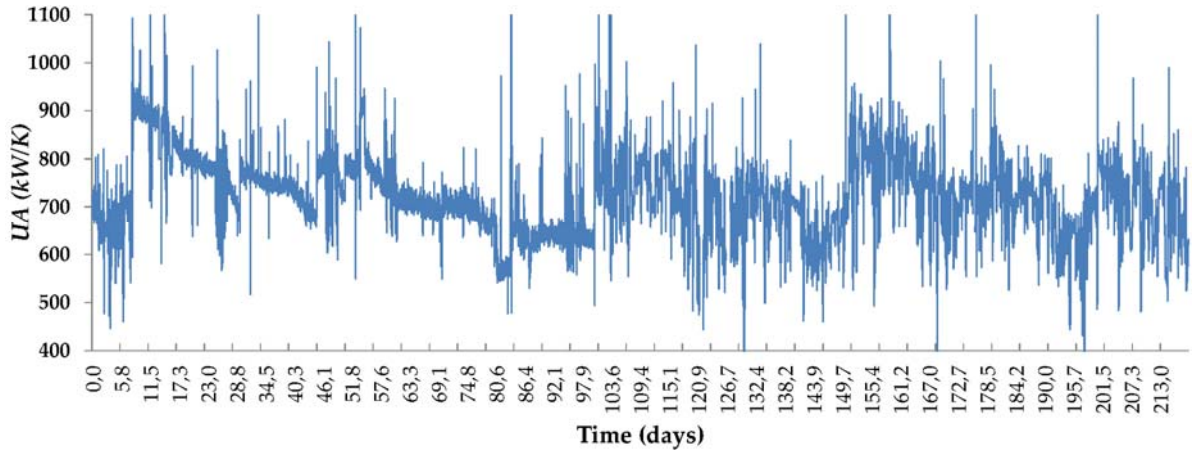


Figure 38. Estimated heat-transmission coefficient.

4.5.4.2 Constrained regression

The objective now in the *regression* (Stage 3 of the proposed methodology) is to build up a polynomial regression model $UA = p(F, t)$ to link/predict the heat-transmission coefficient with the spinbath flow through the exchangers F and with the time t that the plant is operating since last cleaning.

The first issue to face is selecting appropriate samples for training and validation. Although the recorded dataset of 7 months with a sampling time of 5 minutes may look huge, the quality of the collected data is under discussion: the plant was usually operating at high circulating flows, except in the few experiments executed on purpose and in particular situations (product changeovers). Therefore, many samples with the plant operating in a local region are available, but significant information of the convection and fouling behaviors at medium/low flows is missing.

Remark. Although there is no major computational issue in performing regularised LS or SOS constrained regression with hundreds of data, if many samples are agglomerated around the same operating point, the fitted model might specialize too much in such region, as the model structure for regression will not likely contain the same non-linearities that the actual plant which generated the data. Hence, prediction capabilities out of this region can be seriously compromised with such a model. Therefore, the data points must be "triaged" according to their degree of uniqueness (data containing almost-redundant information should get lower weights in the regression, or directly removed from the training set) in order to prevent this possible model bias due to strong non-uniform data densities.

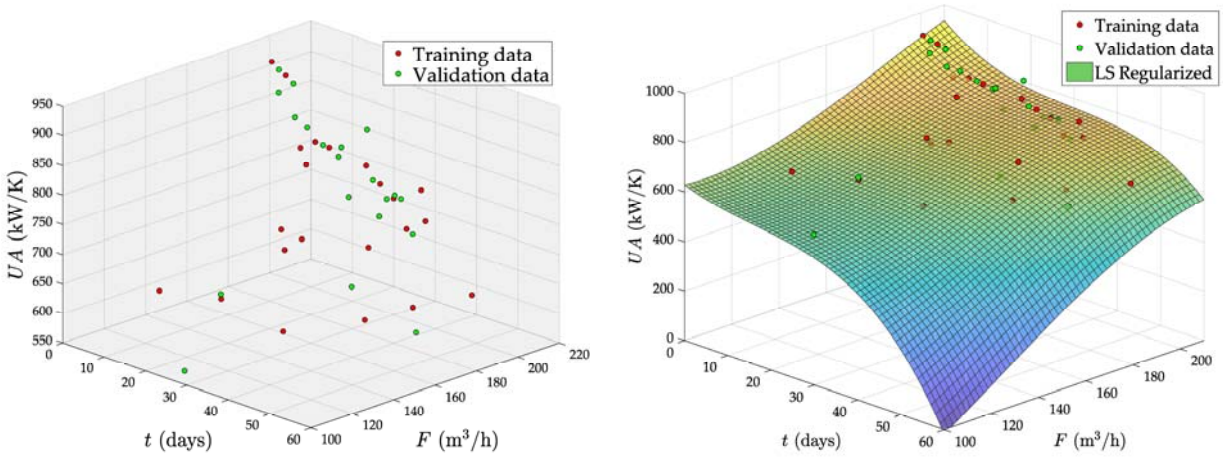
Consequently, after inspecting and analyzing the plant historian, we ended up with a selected subset of 22 samples $\{\widehat{UA}, F, t\}$ for training plus 20 additional samples for validation. These samples, depicted in Figure 39a, contain nearly all the information available in the desired region of operation:

$$\mathfrak{V} \coloneqq \left\{ F, t \in \mathbb{R}^{+} \middle| 100 \leq F \leq 200 \frac{\mathrm{m}^3}{\mathrm{h}}, t \leq 60 \text{ days} \right\} \tag{16}$$

As it can be observed by simple visual inspection, there are enough samples covering $\mathfrak{V}$ at high flows, but there is a significant lack of information at lower flows, especially after a cleaning and when the plant is in operation for more than 40 days.

After centering and scaling the data with the standard deviation, an LS identification was initially tested with exponential regularization in the coefficients corresponding to the higher-degree monomials of $p(F,t)$, see (Pitarch, Sala, & de Prada, A systematic grey-box modeling methodology via data reconciliation and SOS constrained regression, 2019). The best fit (lower total error with the training plus validation sets) is achieved with a polynomial model of coordinate degree at most 3 (Figure 39b):

$$UA(F,t) = -2.5335\mathrm{e}^{-4}F^3 - 7.0692\mathrm{e}^{-4}F^2t + 2.0131\mathrm{e}^{-3}Ft^2 - 5.5415\mathrm{e}^{-3}t^3 + 0.13823F^2 + 0.14058Ft + 0.066824t^2 - 21.0228F - 13.8979t + 1602 \quad (17)$$



a) Datasets for regression

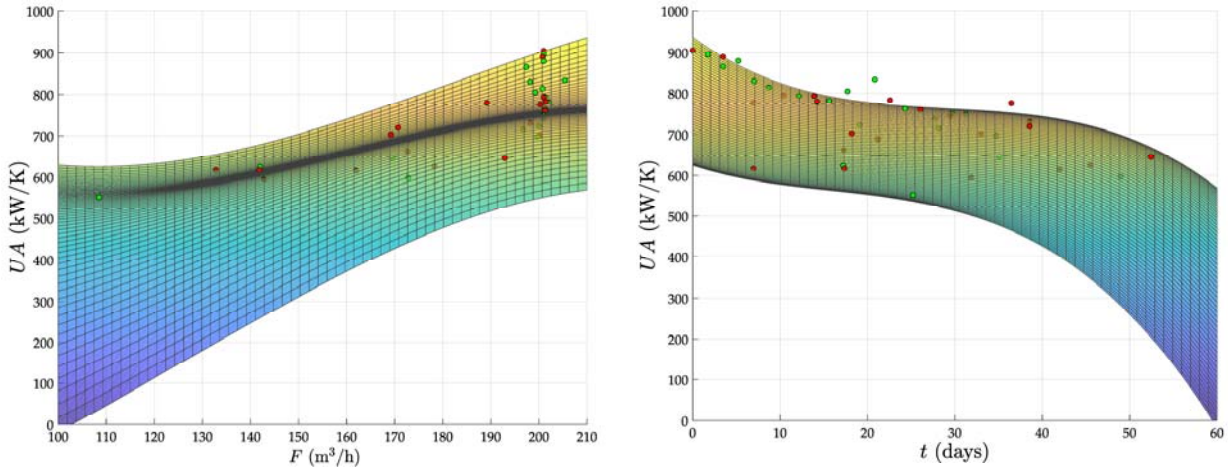
b) Model using regularised LS.

Figure 39. Selected experimental data and first modelling attempt.

There are two aspects in this model which are unacceptable form the physical perspective:

1. *The abrupt falling of the UA from the day 30 onwards is not possible*. Indeed, the predicted UA even reach zero and negative values after two months of operation with low flows. A pump in this plant fixes the flow. Therefore, the fouling due to deposition of organic material must tend to a saturation limit with the time. This is because the flow speed increases as the effective pipe area reduces by fouling and, from basic physics, the deposition of organic particles in the pipes must always decrease with the flow speed.
2. *The mild increase of UA observed at low F when the exchangers are fully clean* (see Figure 40a) *is also physically impossible*: With nearly constant exchange area, UA always must decrease as F does by convective thermodynamics.

Therefore, SOS-constrained regression (Pitarch, Sala, & de Prada, A Sum-Of-Squares Constrained Regression Approach for Process Modeling, 2019) is recalled here to incorporate the above physical insight in the data-driven modelling phase. Hence, the standard LS regression problem is augmented with polynomial constraints to enforce individual monotonic responses with respect to F and t, plus local bounds on the model derivatives, as follows:



a) UA versus the flow.

b) UA versus the operation time.

Figure 40. Partial 2D views of the model got by regularised LS.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 723575

$$\min_{\alpha,\beta,\tau} \quad \sum_{i=1}^{22} \tau_i \tag{18}$$

$$\text{s.t.:} \quad \begin{bmatrix} \tau_i & \widehat{UA} - p(\beta, F^{[i]}, t^{[i]}) \\ \widehat{UA} - p(\beta, F^{[i]}, t^{[i]}) & I \end{bmatrix} \succcurlyeq 0 \; \forall \, i: 1, \dots, 22 \tag{19}$$

$$\frac{\mathrm{d}p(\beta, F, t)}{\mathrm{d}F} - \epsilon - s_1(\alpha_1, F, t) \cdot (60 - t)t - s_2(\alpha_2, F, t) \cdot (200 - F)(F - 100) \in \Sigma_{F,t} \tag{20}$$

$$-\frac{\mathrm{d}p(\beta, F, t)}{\mathrm{d}t} - \epsilon - s_3(\alpha_3, F, t) \cdot (60 - t)t - s_4(\alpha_4, F, t) \cdot (200 - F)(F - 100) \in \Sigma_{F,t} \tag{21}$$

$$\frac{\mathrm{d}p(\beta, F, t)}{\mathrm{d}t} - \lambda_t - s_5(\alpha_5, F, t) \cdot (60 - t)(t - 30) - s_6(\alpha_6, F, t) \cdot (200 - F)(F - 100) \in \Sigma_{F,t} \tag{22}$$

$$\lambda_F - \frac{\mathrm{d}p(\beta, F, t)}{\mathrm{d}t} - s_7(\alpha_7, F, t) \cdot (60 - t)(t - 30) - s_8(\alpha_8, F, t) \cdot (200 - F)(F - 100) \in \Sigma_{F,t} \tag{23}$$

$$s_j(\alpha_j, F, t) \in \Sigma_{F,t} \;\; \forall \, j: 1, \dots, 8 \tag{24}$$

Where symbol $\succcurlyeq$ means that the real matrix in (10) is constrained to be positive semidefinite; $\Sigma_{F,t}$ denotes that polynomials $(19)(20)$-(24) are constrained to be SOS in variables F and t; $\epsilon > 0$ is a small tolerance; $\lambda_t, \lambda_F \in \mathbb{R}^+$ lower and upper bounds on the model partial derivatives w.r.t. t and F, respectively; and α_j are extra decision variables corresponding to the coefficients of the polynomial multipliers s_j, whose highest coordinated degree is chosen to be two[1].

In the above SOS optimisation problem, the linear objective (18) together with the linear matrix inequality (19) are equivalent (by Schur complement) to the usual unconstrained LS fitting. Then, the remaining SOS constraints enforce *local* bounds on the model partial derivatives: (20) and (21) force $\mathrm{d}p/\mathrm{d}t < 0$ and $\mathrm{d}p/\mathrm{d}F > 0$ on $\mho$ respectively, i.e. monotonic responses; whereas (22) and (23) locally limit the slope in each direction (F or t) to the user-defined values λ, in order to avoid an excessive falling in the model response beyond the day 30 of operation.

Note that, although SOS programming is *convex* optimisation, an appropriate coding of (18)-(24) would use the data normalised to zero mean and $\sigma = 1$ (and so would be the region $\mho$ accordingly) to avoid numerical problems in the resolution, analogous to regularized LS or any other numerical approach. For more details on the implementation, on the SOS programming, on proving local positivity of polynomials, etc., the reader is referred to (Pitarch, Sala, & de Prada, A Sum-Of-Squares Constrained Regression Approach for Process Modeling, 2019), (Pitarch, Sala, & de Prada, A systematic grey-box modeling methodology via data reconciliation and SOS constrained regression, 2019) and references therein.

[1] By Positivstellensatz theorem **Invalid source specified.**, $\deg(s_j)$ must be chosen such that $\deg(s_j) \cdot \deg(\mho) \geq \deg(\mathrm{d}p/\mathrm{d}t)$.

With the stated SOS-constrained regression, a well behaved model for UA (see Figure 41a) is obtained with only a ~0.7% fitting degradation (goodness of fit in Table 16) with respect to model (17):

$$\begin{aligned} UA(F,t) = {} & 7.066\mathrm{e}^{-8}F^4 + 2.9544\mathrm{e}^{-6}F^3t + 1.6325\mathrm{e}^{-6}F^2t^2 - 2.4195\mathrm{e}^{-6}Ft^3 + 1.0012\mathrm{e}^{-4}t \\ & - 1.9868\mathrm{e}^{-4}F^3 - 1.5847\mathrm{e}^{-3}F^2t + 5.0898\mathrm{e}^{-5}Ft^2 - 0.013865t^3 + 0.08888I \\ & + 0.23223Ft + 0.62707t^2 - 10.8758F - 22.7836t + 1000 \end{aligned} \quad (25)$$

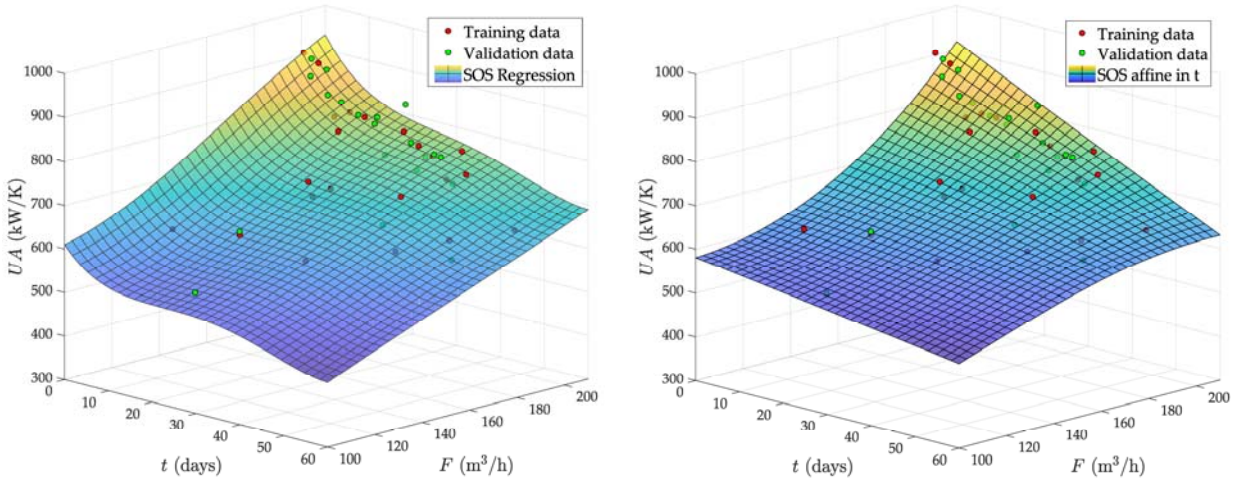
Analysis and comparison with previous work in CoPro

As reported in (Pitarch, de Prada, Wenzel, & Misz, 2018), we assumed the hypothesis that the increase of specific-steam consumption in the plant due to fouling was linear with the operation time. This was done based on direct measurements of the SSC, in order to facilitate the resolution of the maintenance-scheduling problem formulated in (Palacín, Pitarch, Jasch, Méndez, & de Prada, 2018). Now we analyse whether this assumption was reasonably true.

For this aim, the polynomial model $p(\beta, F, t)$ is forced to be affine in t. This requirement can be easily achieved by a constraint on the model convexity:

$$\frac{\mathrm{d}^2 p(\beta, F, t)}{\mathrm{d}t^2} = 0 \; \forall \, F, t \in \mho \quad (26)$$

This equality can be trivially enforced by a couple of opposite-sign inequality SOS constraints, intersecting in zero, which replace (22) and (23) in the fitting problem, details omitted for brevity. Doing this, effectively the obtained model is affine in t and nonlinear in F, as Figure 41b shows.



a) Model fulfilling bounds on the slopes.

b) Model affine in t (restricted curvature).

Figure 41. Models computed via SOS-constrained regression.

This model incurs in an ~13% fitting degradation w.r.t. the "optimal" (17). Nevertheless, what is more relevant than the fitting error in this case is the observed variation of the slope in t at different flows. This indicates that *the assumption in* (Palacín, Pitarch, Jasch, Méndez, & de Prada, 2018) *is acceptable as long as F remains nearly constant*. Indeed, as the plant was normally operating at high flows when the data was collected from the historian, we did (could) not realise of this varying behaviour with the flow.

Table 16. Goodness of fits for the tested approaches.

Method	MSE Training	MSE Validation	Total RMSE	Deterioration
Regularised LS	13448	14282	166.52	-
SOS constrained	14751	13362	167.67	0.69%
SOS affine	20147	15131	187.82	12.79%

5 Conclusions and recommendations

5.1 Conclusions related to modelling for the COPRO use cases

At the start of the COPRO project, it was already known from the experience of the project partners that data gathering, data exploratory analysis and pre-processing is very effort intensive and can consume in the order of 90% of the modelling effort.

Another conclusion that arises from this project is that context to data is crucial. While it is not too effort intensive to obtain data from historians in plants, determining normal from abnormal operation, assessing data quality, and understanding the relation of sensors to the process equipment is non-trivial and requires process knowledge.

Even before data is, collected information/context is required about the relevant influencing factors and structure of the plant.

For this reason close collaboration between process engineers, operators, data-providers (research and planning departments of operating companies) and consultants/researchers is required. When this human chain gets too long, or priorities are such that response times for parts of this communication chain get too long, this context is not communicated well.

Regarding model structure one conclusion from this project is that for a significant number of case studies (INEOS Naptha cracker, INEOS coking, INEOS ammonia network, Lenzing evaporator network) either simple affine models are sufficient or the data-driven ALAMO approach (Cozad, Sahinidis, & Miller, 2014) is used.

A semi-empirical / first-principle model was used for the FRINSA sterilization use case. In order to increase model efficiency, reduced order techniques were applied for the modelling of can temperature distributions. This model was reasonably simple and in contrast to other case studies in this project validation was done based on data from experiments. The models obtained are appropriate for real-time optimization tasks (Vilas and Alonso, 2018) as well as for the development of software sensors. On the other hand, the computational load associated with plant scheduling is too high which makes difficult to embed these models in real-time scheduling. The solution proposed consists of using the models to derive tables that relate decision variables (such as process temperature or autoclave can load) with other variables of interest (such process time, energy consumption or product quality). If the grid used for the decision variables is small enough, the scheduling algorithm can use interpolation to take into account intermediate values.

For the INEOS use case, a comparison has been made between first-principles and data-based modelling using a surrogate modelling approach. It was found that the first-principles model had significant added-value in particular if the purpose would include improvements through design changes rather than operation changes only. It was also found that time spent for development of the surrogate model is approximately 20% of the time required for the development of the physical model.

A conclusion that arises from this project is that software, tooling and expertise is very important during the stage of data pre-processing, in particular when multiple organizations are involved. Basic algorithms like for example outlier detection, steady-state detection, scaling, smoothing are often re-

developed for each modeling endeavor. This also holds for more advanced algorithms like feature and pattern extraction. While toolboxes and purpose built data-driven modelling packages (e.g. (Divis Intelligent Solutions GmbH, 2018)) offer some pre-processing functionality, typically many applications require custom steps for selecting and removing data that are developed in spreadsheets (MS Excel) scripting languages (Python) or special purpose languages for data processing (Matlab, R). It is common that development for pre-processing steps is repeated for each application, adding overhead. This pre-processing can also require expertise in IT/signal processing/data analytics that modelers and chemical engineers might not possess in sufficient degree. In addition, it might be difficult to review/share the pre-processing steps between different organizations as they might be implemented in commercially licensed tools (Matlab) or rely on expertise in particular programming languages.

5.2 Conclusions related to hybrid modelling

UVA has proposed a two step approach for hybrid modelling: first data-reconciliation was done on a first-principles partial model in order to obtain coherent estimates for the inputs and outputs of the data-driven part of the plant complete hybrid model. This data-driven part is then fitted from these estimated inputs using a machine learning / function approximation algorithm. To enforce physical constraints on the data-driven part of the hybrid model, constrained fitting should be employed to include any physical insight that the modeler may have on the process, providing thus reliable and coherent prediction capabilities. The approach was illustrated through the modelling of an evaporation plant and its attached cooling system from the Lenzing's use case. These models became the basis for the real-time optimisation developments on the evaporation, water-distribution and heat-recovery networks, conducted within the work in WP3.

The experience gained through this work made us understand the potential dangers of just trust blindly in machine learning approaches for process modelling: One needs to be very careful with the data used for regression and needs to spend sufficient time in the validation stage, in order to be sure that the obtained models behave correctly in all the situations/operation region where the model is used for prediction. Otherwise, any deployed decision support system will be not reliable. To avoid (or reduce) these kind of issues is key for the efficient construction of models (hence saving precious engineering time), so we recommend the presented hybrid-modelling approach which uses the wide physical knowledge on process-systems literature as the basis to get both reliable process data and models.

PSE has conducted a meta-modelling study for 2 complex unit operations to determine whether data-driven models are able to accurately represent these models. This would be a pre-requisite to model plants that include these types of unit operations using a hybrid modelling approach. It was found that this was possible, but that for most input-output relationships non-linear terms where required and that linear modelling is not sufficient to cover the behavior of models over a convex hull derived from the operating range of each individual input.

A hybrid modelling tool was developed that integrated with the gPROMS first principle modelling environment. An outline of this tool was presented within this report and a detailed specification will be presented in the upcoming COPRO deliverable D1.2.

Using the hybrid modelling tool, the data-driven model in an cracker optimization use case, inspired by the current optimization approach at INEOS, it was found that the hybrid model was able to reach a solution relatively close to that of the original model.

Regarding hybrid modelling in gPROMS, the recommendation is proceed the hybrid modelling tool development so that in particular the prediction part reaches a high enough TRL level where it can be productized and licensed as part of the gPROMS product.

6 Bibliography

A.A. Alonso, A. A.-M.-C. (2013). Real time optimization for quality control of batch termal sterilization of prepackaged foods. *Food Control*(32), 392-403.

Abril., A. C. (2003). Procesos de conservación de alimentos. *Tecnología de alimentos. Mundi-Prensa*.

Arthur, D., & Vassilvitskii, S. (2007). k-means++: The advantages of careful seeding. *In Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, Society for Industrial and Applied Mathematics*, pp. 1027–1035.

Badwal, S. S. (2014). Review of Progress in High Temperature Solid Oxide Fuel Cells. *Journal of the Australian Ceramics society, 50*(1).

Barton, P., & Pantelides, C. (1993). gPROMS - A combined discrete/continuous modelling environment for chemical processing systems. *Simulation Series, 25*, 25-34.

Beisheim, B., Rahimi-Adli, K., Krämer, S., and Engell, S.;. (2018b). Energy performance analysis of continuous processes using surrogate models. (Manuscript in preparation).

C. Vilas, A. A.-M.-C. (2018). Toward predictive food process models: A protocol for parameter estimation. *Critical Reviews in Food Science and Nutrition*(58), 436-449.

C. Vilas, M. G. (51-65). Desarrollo de una librería de componentes en EcosimPro para la operación de plantas de procesamiento térmico de alimentos. *Revista Iberoamericana de Automática e Informática Industrial*(5), 2008.

COPRO partners. (2019). D2.5.

Cozad, A., Sahinidis, N. V., & Miller, D. (2014). Learning surrogate models for simulation-based optimization. *AIChE Journal, 60(6)*, pp. 2211-2227.

Divis Intelligent Solutions GmbH. (2018). ClearVu Analytics. Retrieved from http://www.divis-gmbh.de/en/about.html

H.S. Ramaswamy, C. C. (2002). Color and Texture Change Kinetics in Ripening Bananas. *Lebensm.-Wiss. u.-Technol*(35), 415–419.

J.R. Banga, A. A.-M. (1993). Kinetics of thermal degradation of thiamine and surface colour in canned tuna. *5Z Lebensm Unters Forsch*(197(2)), 127–131.

K.M.Nauta, COPRO project partners. (2018). *D1.3 report: Model Quality, model uncertainty and model maintenance.* COPRO project.

Kalliski, M., Voglauer, B., Seyfriedsberger, G., Jasch, C., Röder, T., & Engell, S. (2017, October). Resource efficient operation of an evaporator network. *Symposium on Computer Aided Process Engineering – ESCAPE 27*.

Llanos, C., Sanchéz, M., & Maronna, R. (2015). Robust Estimators for Data Reconciliation. *Industrial & Engineering Chemistry Research, 54*(18), 5096-5105.

M. von Stosch, R. O. (2014). Hybrid semi-parametric modeling in process systems engineering: Past, present and future. *Computers & Chemical Engineering*, 86-101.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, pp. 281-297.

Marcos, M., Pitarch, J., de Prada, C., & Jasch, C. (2018). Modelling and real-time optimisation of an industrial cooling-water network. *22nd IEEE Inter. Conf. on System Theory, Control and Computing (ICSTCC)*, (pp. 591-596). Sinaia.

Neumaier, A. (1998). Solving ill-conditioned and singular linear systems: A tutorial on regularization. *SIAM Review, 40*, 636-666.

Palacín, C., Pitarch, J., Jasch, C., Méndez, C., & de Prada, C. (2018). Robust Integrated Production-Maintenance Scheduling for an Evaporation Network. *Computers & Chemical Engineering, 110*, 140-151.

Pedregosa, F. e. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 2825-2830.

Pitarch, J., & de Prada, C. (2018). *D2. 1–Report on Dynamic Data Reconciliation of Large-Scale Processes.* Deliverable, SPIRE.

Pitarch, J., de Prada, C., Wenzel, S., & Misz, Y. (2018). *D3. 1–Preliminary report on optimisation methods for large plants with discrete and continuous degrees of freedom.* Deliverable, SPIRE.

Pitarch, J., Palacín, C., de Prada, C., Voglauer, B., & Seyfriedsberger, G. (2017). Optimisation of the Resource Efficiency in an Industrial Evaporation System. *Journal of Process Control, 56*, 1-12.

Pitarch, J., Palacín, C., Merino, A., & de Prada, C. (2017). Optimal operation of an evaporation process. In H. Bock, H. Phu, R. Rannacher, & J. Schlöder (Eds.), *Modeling, Simulation and Optimization of Complex Processes HPSC 2015* (pp. 189-203). Springer, Cham.

Pitarch, J., Sala, A., & de Prada, C. (2019). A Sum-Of-Squares Constrained Regression Approach for Process Modeling. *12th IFAC Symposium on Dynamics and Control of Process Systems, Including Biosystems (DYCOPS)*, (p. Accepted). Florianopolis.

Pitarch, J., Sala, A., & de Prada, C. (2019). A systematic grey-box modeling methodology via data reconciliation and SOS constrained regression. *Processes, SI: Process modelling and simulation*(In Press), 23.

Rahimi-Adli, K., Schiermoch, P. D., Beisheim, B., Wenzel, S., & Engell, S. (2019). A model identification approach for the evaluation of plant efficiency. *In Proceedings of the 29th European Symposium on Computer Aided Process Engineering*, p. In Press.

Silva, A. d. (2018 (expected)). *Hybrid modelling / machine learning for soft-sensing and process modelling.* Lisbon: Lisbon Technical University.

T. Back, P. C. (n.d.). *Automatic Meta-modelling of CAE Simulation Models.* Divis Intelligent Solutions GmbH.

Wilson, Z., & Sahinidis, N. (2017). The ALAMO approach to machine learning. *Computers & Chemical Engineering, 106*, 785-795.

Wold, H. (1985). Partial least squares. *Encyclopedia of statistical sciences, 6*, pp. 581–591.

Zorzetto, L., Maciel Filho, R., & Wolf-Maciel, M. (2000). Processing modelling development through artificial neural networks and hybrid models. *Computers & Chemical Engineering, 24*(2-7), 1355-1360.