Simulation Modelling Practice and Theory 30 (2013) 1–20

Contents lists available at SciVerse ScienceDirect

Simulation Modelling Practice and Theory

journal homepage: www.elsevier.com/locate/simpat

Production modelling for holistic production control

Miha Glavan [a,*], Dejan Gradišar [a], Stanko Strmčnik [a], Gašper Mušič [b]

[a] Jožef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia
[b] Faculty of Electrical Engineering, University of Ljubljana, Tržaška 25, 1000 Ljubljana, Slovenia

ARTICLE INFO

Article history:
Received 13 April 2011
Received in revised form 9 May 2012
Accepted 19 July 2012
Available online 15 October 2012

Keywords:
Holistic production control
Production performance indicators
Black-box modelling
Neural networks

ABSTRACT

Holistic production control is a concept that introduces production optimisation by employing model-based, closed-loop control of the principal production Performance Indicators (pPIs). The concept relies on the development of a simple black-box model that describes the relation between the main pPIs and the most influential input (manipulative) variables. In this article the modelling aspects of the holistic production control implementation are presented. The main steps of the production modelling procedure are described, such as data preprocessing, the definition of pPIs, the selection of input variables and the derivation of black-box models. Particular emphasis is given to a modelling approach based on neural networks and a corresponding modelling assistant tool, which has been developed to support the modelling procedure. The approach is illustrated on the Tennessee Eastman benchmark process, where neural network models for three main production performance indicators, i.e., costs, quality and production rate, are derived.

1. Introduction

For advanced manufacturing it is important that the production process is efficient, adaptable and flexible to quickly adjust to customers' demands and at the same time minimises the production costs. Therefore, during the production process, fast and accurate actions are needed at all management levels in a factory.

Information technology (IT) has a considerable impact on the efficiency and quality of manufacturing, especially in terms of enabling better control and optimisation. However, in the past its introduction was, for several reasons, more oriented to the control of manufacturing processes (control of particular machines, reactors, devices, etc.) and to the support of business functions (see the functional hierarchy model of a production enterprise in Fig. 1). It was only some 15 years ago that IT started to penetrate into the so-called production control level, where scheduling, dispatching, plant-wide optimisations and local optimisations are typically performed.

Nowadays, one can say that the different hierarchical levels in a factory are relatively well supported by IT tools, which should enable the effective supervision and control of particular subsystems. The available models and standards are merging traditionally disparate functions and systems across the enterprise. The corresponding information-technology solutions allow access of the right information, in the right place, at the right time and in the right format. However, there are at least two important challenges that remain. The first one is the lack of integration and coordination among different control levels, which results in the suboptimal operation of the entire plant. And the second one is the vast amount of collected data, and the few ideas about how to cope with it and how to use it for appropriate decisions and production optimisation.

Over the years some research areas have emerged that face these issues. One of them is the area of *plant-wide control*, which was developed within the process engineering community. The entire flow-sheet, not only the lower levels, are

* Corresponding author. Tel.: +386 1 477 36 84.
E-mail address: miha.glavan@ijs.si (M. Glavan).

1569-190X/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.simpat.2012.07.010

Nomenclature

$K_u, u(\cdot)$	vector of manipulative variables
R	available input space
K	vector of production performance indicators (pPIs)
K^*, r	reference vector from business level
$K\|\|M$	model prediction of production performance indicators
N	number of samples
Z^N	measured data set
$u(t)$	input signal
$y(t)$	output signal
$\hat{y}(t), g(\cdot)$	predictor of $y(t)$
$\varphi(t)$	regression vector
Θ	parameter vector
n_a	number of delayed outputs in $\varphi(t)$
n_b	number of delayed inputs in $\varphi(t)$
n_k	time delay
D	weight decay factor
y_s, y_m	setpoint and minimisation oriented pPIs
Q_s, Q_m	control objective weights for y_s and y_m

Abbreviations

RTO	real-time optimisation
HPC	holistic production control
pPIs	production performance indicators
MES	manufacturing execution system
MPC	model predictive control
TE	Tennessee Eastman
NARX	nonlinear autoregressive model with exogenous inputs
MSE	mean-square error
OBS	optimal brain surgeon
NPE (NCE)	normalised prediction (control) error
IAE	integral of absolute error

covered and the emphasis is on control structure design. In this frame, an important part of research is dealing with a hierarchical decomposition of the original control design problem based on heuristic rules. The heuristic logic is developed so as to keep the process variability and therefore the operational plant objectives within acceptable limits for a given set of disturbances (see [1] or [2]). The plant-wide control is also studied in discrete manufacturing, where the existing challenges require a form of technical intelligence that goes beyond simple data, through information to knowledge [3]. The integration of data-mining methods [4–6] into the decision making on the production control level is required. The resulting knowledge, in the form of static and dynamic models, will facilitate new opportunities for collaboration throughout the plant, and across the supply chain. This will enable us to meet the increasing demands on flexibility and reactivity within the Intelligence in Manufacturing (IIM) paradigm.

In recent years, *closed-loop control* is also being used at the so-called production control level – the level between the business and process control levels. One promising approach here is Real-Time Optimisation (RTO) [7]. RTO is defined as a model-based, upper-level, control system that is operated in a closed loop and provides set-points to the lower-level control systems in order to keep the process operation as close as possible to the economic optimum. A summary of the recent developments and applications of dynamic real-time optimisation is given by Kadam and Marquardt [8]. Aspects of how to use Model Predictive Control within the RTO structure are presented by Rawlings and Amrit [9]. Similar concepts could also be found in the field of discrete manufacturing (e.g. [10]). On the other hand, there are mathematically oriented approaches based on the solution to a given large-scale, mixed-integer, nonlinear-programming, dynamic-optimisation problem, which in the limit should be able to simultaneously determine the optimal size of the process units and their interconnections as well as the optimal control scheme configuration (see [11] for a review).

The main disadvantage of these approaches is the huge complexity, the need to cope with an enormous number of details and the low robustness of solutions to changes in the production. Another approach is represented by *holistic production control* (HPC), which introduces the concept for production optimisation, with detailed analysis of the historical production process data. HPC can be viewed as an optimisation and control upgrade for the manufacturing execution systems (MES), employing the advanced analysis of process data to support the production manager with the decisions needed to realise business objectives.

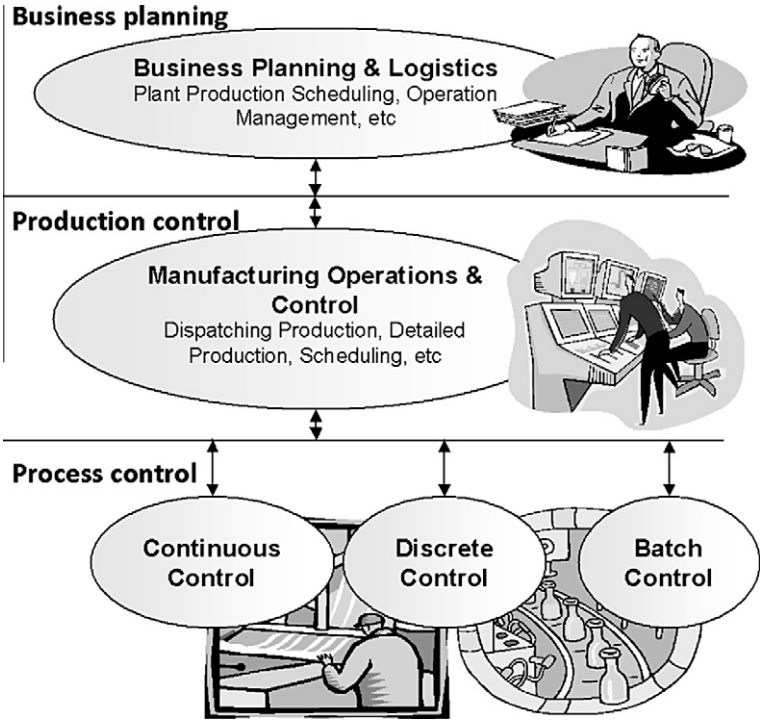


Fig. 1. Process hierarchy.

The idea is based on the work of Zorzut et al. [12], where closed-loop control is implemented on the production process for a polymerisation plant. Key performance indicators (KPIs) of production, referred to as ''production performance indicators (pPIs)'' are introduced to reduce the extensive amount of measured process data and to aggregate the important information concerning process economic efficiency. The optimisation and control of the production is based on the closed-loop control of a few pPIs, using a simple black-box model. The usage of pPIs for production control is justified also by ISO standard [13], which states that key performance indicators for manufacturing operations tend to be in the 20% of the total number of process and resource measurements being observed, while they have 80% or more impact on the processes being tracked.

Modelling approaches with key performance indicators can be noticed in the literature and some commercial products. However, these solutions are mostly focused only on the business planning level. Commercial business platforms mainly enable monitoring and managing of business performance indicators for historical data and in real-time (e.g. [14]), while other solutions offer guidelines on how to formulate and use business oriented KPI models to improve organisation performance and how to describe the complex interrelationships between different parts of the business process. Historical data oriented approaches have also been reported. Here interactions among the process parameters and the goal KPI are directly identified to support business KPI monitoring and business process redesign (e.g. [15–17]).

On the other hand, there are also some commercially available MES oriented solutions for knowledge extraction on the production level (e.g. [18,19]), which can be used for data collection and to some level also offer the functionality to review some performance indicators. Basics for these tools are mostly oriented to descriptive data-mining or static production analysis. Therefore, users of these tools must be very knowledgeable and have a deep theoretical background to extract useful information to be used to correct production.

In this article we present a concept which extends the usability of such support systems, as dynamical modelling of the production oriented PIs is introduced and the decision-support is realised with the employment of solutions from the process control field.

As the extensive experimentations on processes are usually severely restricted, the main problem is how to obtain an appropriate model for the principal pPIs, using historical records of the production process data only. Therefore, an effective empirical dynamic modelling approach is required, which would be able to extract the relevant information from the vast amount of production data. In the engineering community the most common approaches are focused on the parametric models [20]. Another approach is to use nonparametric modelling principles, where the sizes of the models are allowed to grow with the data size (e.g. [21]).

The article addresses a problem, how to efficiently synthesise methods widely used for data-mining, system identification and process control, in order to support decisions on the production control level. A solution with the closed loop control of

the main pPIs is presented, where appropriate steps of the production modelling procedure are specified and illustrated on the Tennessee Eastman benchmark process. Particular emphasis in this article is given to the parametric modelling approach, based on neural networks and to a corresponding software tool that was developed to support the modelling procedure. This modelling-assistant tool tends to automate and to simplify the modelling steps for HPC, in order to ease the model identification also for non-modelling experts, like the production managers. Furthermore, input variable selection, which is often neglected in a modelling procedure, is identified as a crucial modelling step in HPC design. Some preliminary results of this work can be found in [22].

This article is structured as follows. Section 2 introduces the holistic production concept. Section 3 presents the steps needed to identify the appropriate model, with the employment of black-box modelling principles. Section 4 introduces the developed modelling assistant. In Section 5, a modelling procedure for HPC is demonstrated for the case study of the Tennessee Eastman benchmark plant. Finally, the conclusions are given in Section 6.

2. Holistic production control

The concept of holistic production control can be best explained by the scheme depicted in Fig. 2. The process we would like to control is indicated by the block *Production process*. Note that this block also covers the low-level process control. Different inputs (K_u) are available to manipulate the production process. These inputs are actually the reference values for the process control loops and/or other manipulating variables not used within the stabilisation loops. On the other hand, there are many measurable disturbances (d) and outputs (y), which are used to calculate on-line the pPIs (K) – *pPI calculation block*. The pPIs are the production variables that are used by the production manager to determine the appropriate input values (K_u) in order to optimise the production process. The demands from the business control level are given as reference pPIs (K^*). The attempt of the concept described here is to help the production manager with the decision-making process, which would close the loop for the introduction of the *Production controller* & *Optimiser*.

One of the possible solutions with this approach is to apply model-based control and optimisation. To enable this, an appropriate model describing the behaviour of the process projected on pPIs is required (the *pPI model*). The model can be updated online and can provide the controller with the predicted outputs $K|M$. The model can also consider the measurable disturbances (d). Based on current values K, the predicted outputs $K|M$ and the reference values K^* (i.e., the planned business goals) the *Production controller* & *Optimiser* determines the appropriate input values K_u and in this way supports (or substitutes) the production manager (Eq. (1)).

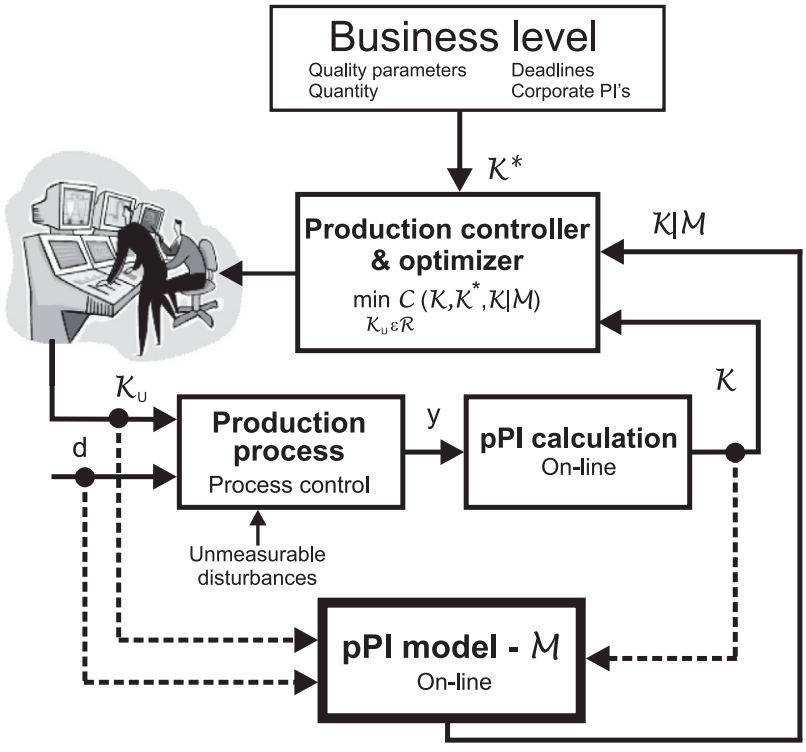


Fig. 2. Concept of holistic production control.

$$\widehat{K_u} = \arg\min_{K_u \epsilon R} C(K, K^*, K|M) \quad (1)$$

How to derive an appropriate pPI model represents the main challenge of the HPC approach.

3. Production modelling for holistic production control

Modern manufacturing systems are in many cases and for various reasons too complex to be accurately described analytically from first principles. Instead, one can assume that the relationship between the inputs and the outputs can be described by a stochastic, high-dimensional model from a class of generally nonlinear model structures.

The production model has to include enough details of the production process to reflect the dynamics for production control. This model should be relatively simple in comparison to the models used for the process control level, yet because of the overall complexity and the limitations of testing the process, this task is extremely complex. Production control usually requires that the model is easy to adapt online as well. Therefore, the main objective is the development of the concept of identifying a relatively simple, input–output model of the production.

The main steps of the production modelling are shown in Fig. 3. More detailed discussion about each of these steps and a short overview of the potential methodologies are given in the following subsections.

3.1. Data preprocessing

Special attention is needed when data from a historical production database are used. From the vast amount of data, the informative portions need to be identified. These data segments should cover the interesting dynamics of the pPIs, for which we would like to determine the future behaviour. Furthermore, any outliers or missing data need to be properly analysed, and to cover all the process operating conditions an uneven data distribution is needed.

Data-cleaning procedures can be applied to detect and remove any outliers present in the data. As pointed out in [20], nonlinear data-cleaning procedures are recommended. We can find many filters in the literature proposed for this task: the Martin–Thomson filter [23,24], the FIR-median hybrid (FMH) filter [25], the Hampel filter [26], etc.

3.2. Definition of production performance indicators – pPIs

The research field of performance measurement systems (PMSs) is becoming increasingly important for industry and academics [27]. Performance indicators (PIs) are commonly used by organisations to evaluate their overall economic success and many recommendations have been presented on how to specify such indicators [27]. These indicators should cover the relevant business aspects of the specific process, where the status of the process should be evaluated with relatively short-term indicators, since longer-scale business estimations (e.g., an annual profit report) are useless for quick process adjustments [28].

As mentioned, the production objectives are usually aggregated in production performance indicators (pPIs). The selection of these pPIs should be performed manually, with extensive consideration of the production expert's knowledge of the process.

3.3. Input variables selection

To simplify the model and to enhance model's accuracy only the most relevant manipulative variables need to be identified. On the basis of the historical process data, an extensive analysis is needed to evaluate which inputs have the biggest impact on the selected pPIs.

Variable selection is already a widely applied methodology in the field of data mining. But, as noted by some authors, like Smits et al. [29], in modelling projects it is mostly assumed that true inputs are a-priori known or all the available inputs are used in a model. To avoid the so-called curse-of-dimensionality, which essentially limits the robustness of a data-based model, only the most relevant inputs need to be selected. This represents an especially important step for HPC design, as in real-world production processes many potential variables are available. Furthermore, as aggregated pPIs are connected with many process variables it is often found that some a-priori excluded inputs are later identified as significant, and vice versa.

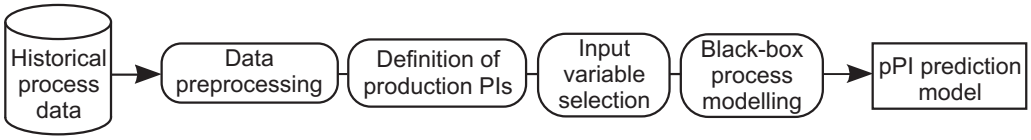


Fig. 3. Production control design steps.

In the literature, three major principles for variable selection are used [30]:

- feature construction,
- variable ranking,
- variable subset selection.

Feature construction represents methods where all the inputs are transformed to a subspace with reduced dimensionality. The transformation can be used to incorporate domain knowledge or to construct an orthogonal subspace in order to remove any collinearities among variables. So, while the dimensionality is reduced, the new variables should incorporate the majority of the information present in the original variables. A variable transformation can be performed with clustering, a linear transformation of the input variables (e.g., principal component analysis – PCA, linear discriminant analysis – LDA, factor analysis – FA)[31] or more complex nonlinear transformations (e.g., methods presented in [32]).

Variable ranking methods represent simple methods, successively evaluating all the potential inputs according to their importance for the output variable. All potential input variables are then appropriately ranked, and less influential inputs are neglected with a consideration of some threshold. Methods like analysis of correlations, mutual information, gamma test, etc. are usually employed.

On the other hand, *variable subset selection* methods also test combinations of the input variables, since completely useless variables can provide a significant prediction improvement, when used with some other variables [30]. Such methods consists of four main steps [33]: the generation of a candidate subset, subset evaluation and selection, stopping criteria and model validation (see Fig. 4).

An exhaustive selection procedure examines all the possible subsets, and consequently the optimality of the solution can be guaranteed. On the other hand, it is computationally very expensive, since all the possible combinations should be tested. Therefore, random methods (e.g., genetic algorithms) and stepwise algorithms (forward selection, backward elimination or their combination) are widely applied. At the end the selected inputs need to be validated to see if the solution is sufficient.

According to how the subsets are evaluated, the methods found in the literature can generally be divided into two main groups: *model-free methods* (filter methods) and *model-based methods* (wrapper methods) [34]. Model-free methods only need low computational capabilities, while the selection is based on an analysis of the available data. Therefore, the input analysis is based on statistical tests, properties of function, etc. (e.g., partial mutual information [35], Lipschitz coefficients [36], consistency and inconsistency measures [34], etc.). Model-based methods result in evaluation of models (e.g., fuzzy models, neural networks, etc.) with a different selection of input variables. Many models are created and evaluated in order to determine the optimal regression vector.

As we are dealing with dynamical systems, the current values of production performance indicators are not dependent only on the current input values, but also on their time-delayed values. The input–selection problem is therefore augmented by the selection of lagged inputs and outputs that are used as regressors.

3.4. Black-box process modelling

The HPC efficiency is closely related to the production model, which should describe the main features of the production process with an acceptable level of approximation. The production process is typically a highly complex process, with nonlinear relationships among the vast quantity of process variables. Since our model should be simple enough and the development time needs to be short, black-box modelling techniques are preferred. Furthermore, the production model should be extracted mainly from the historical process data, since extensive experimentation on the real process is often too expensive or restricted. If the process characteristics were to change during the use of the production model, new process data should be analysed and a better model extracted. The cyclical generation and validation of new models will enable a rather conservative adaptation of the model-in-use to long-term changes in the production.

The main idea of the parametric black-box modelling techniques is to trim some universal input–output functions, with a fixed number of parameters, to accurately represent the true process dynamics (2). The goal is to minimise the mismatch $e(t)$ between the true process response $y(t)$ and the model prediction $g(\cdot)$, where the trimming is performed solely on the basis of the process input–output data pairs $Z^N = \{u(t), y(t)\}_{t=1}^{N}$.

$$y(t) = g(\varphi(t), \theta) + e(t) \tag{2}$$

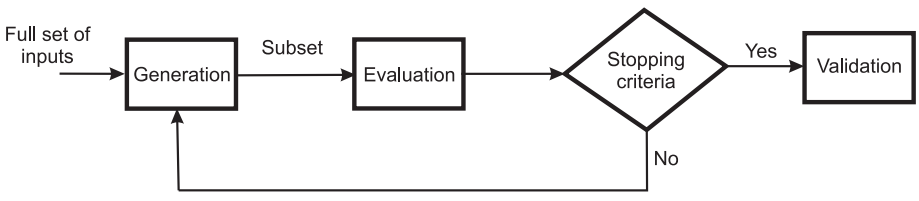


Fig. 4. Input-variable subset selection procedure [33].

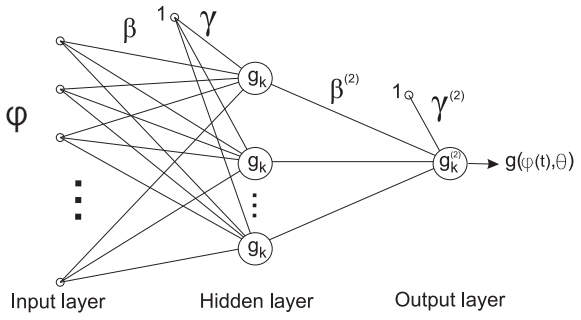


Fig. 5. Feed-forward network architecture with one hidden layer.

Black-box models can be seen as the connection of two mappings [37]. The first mapping constructs the regression vector $\varphi(t)$ from past inputs and outputs, which enables a representation of the dynamical model behaviour. Another mapping predicts the future behaviour of the system $\hat{y}(t)$, with the nonlinear mapping from the regressor space to the system output. This nonlinear mapping is found within a family of functions (3), parameterised by the parameter vector (θ), where g_k refers to the basis function, usually derived from a single, mother basis function.

$$\begin{aligned} &\hat{y}(t) = g(\varphi(t),\theta) = \sum_{k=1}^{n} \alpha_k g_k(\varphi(t),\beta_k,\gamma_k), \\ &\theta = [\alpha_1 \cdots \alpha_n, \beta_1 \cdots \beta_n, \gamma_1, \cdots, \gamma_n]^T \end{aligned} \tag{3}$$

From such a flexible structure, popular nonlinear mappings can be derived, like Neural Networks, Wavelets, Kernel Estimators, Nearest Neighbors, B-splines, Fuzzy models, etc. [38].

A neural network's ability to learn a complex nonlinear mapping only by example has received a lot of scientific attention over the past two decades. They have been found to be a useful tool for modelling and their practical use has been proved in different realisations in industry [39]. A multi-layer network comprising at least one hidden layer and a continuous nonlinear activation function has been recognised as a universal approximator [40]. Therefore, in theory, any continuous function can be approximated to an arbitrary degree of exactness.

The architecture of the multi-layer network is depicted in Fig. 5. The basis function expansion g_k contains adjustable parameters (β_k and γ_k):

$$g_k(\varphi(t)) = \kappa(\varphi(t),\beta_k,\gamma_k) = \kappa(\beta_k^T\varphi(t) + \gamma_k) \tag{4}$$

The network overall mapping is not constructed as a linear combination of basis functions (3), rather they are connected in a feedforward manner, where the outputs from the first layer of neurons represent the regression vector, for the basis functions in the next layer $g_k^{(2)}$ [38].

The model fitted on the training data set Z^N does not necessarily represent the true process dynamic, since it can be overtrained. Therefore, early stopping methods are widely used, where a cross-validation of the model on an additional data set is performed to test the model. Another approach is to directly influence the least-influential parameters of the model structure during training (e.g., regularisation methods [37]) or to completely remove the redundant parameters from the already trained model (e.g., network pruning [40]). With an adjusted model structure the model is simplified and the generalisation capabilities are enhanced.

4. Modelling assistant

To ease the modelling procedure for the HPC, a user-friendly tool is being developed. Using such a support tool, the system integrator and the production manager would have the possibility to identify a production model, based on the historical operational data of the process, and integrate it in a model-based HPC solution.

The main purpose of the developed modelling assistant is to automate the model-development procedure and to support the manipulation and maintenance of already existing models. As the potential users of such a tool are non-modelling experts (e.g., production managers), the program tends to simplify the model-identification procedure, where the user would not need to understand a detailed identification theory.

The assistant is comprised of an input–output analysis component with a cross-correlation input-variable selection algorithm and a neural network modelling procedure. The main characteristics of the tool are:

- incorporation of an input-variable selection step as a part of the model structure selection,
- automation of the modelling procedure,
- generation of many models with different training settings and alternative model structures,

- automatic validation of the models to extract the most robust model,
- validation of the models on the fresh production data.

Black-box modelling is based on the one-hidden-layer network model, with sigmoid transfer functions in the hidden layer and linear transfer functions in the output layer. This model structure was considered, as many difficult and diverse problems have been successfully modelled with such a type of neural network [40].

Simple and straightforward modelling solutions are implemented in order to limit the user-controlled settings. While the convergence of the model to an optimal solution strongly depends on the selection of the initial weights, different initial values are sequentially considered to avoid a convergence to local minima. For each initialisation of the inputs an additional model is constructed. Furthermore, the optimal network topology cannot be predicted in advance, since the number of neurons in the hidden layer influences the complexity of the neural network's nonlinearities. Therefore, a trial-and-error approach is usually applied, where the number of neurons in the hidden layer and the training parameters are systematically varied [39]. A similar approach is used in the modelling assistant, where many neural models with different parameters are identified in a sequence.

The selection of the model with best generalisation properties is made with a validation procedure. As extensive historical process data represent the basis for HPC design, there should not be a problem to obtain different validation datasets. A direct comparison of the generated models on the datasets not used in the training is therefore applied to distinguish the model with the best generalisation characteristics. Automatic validation is implemented, where many different validation datasets and alternative types of validations (simulation, k-step ahead predictions) can be evaluated. A direct comparison of the overall performance criteria (MSE, NPE) provides a simple way of finding the most robust neural-network model.

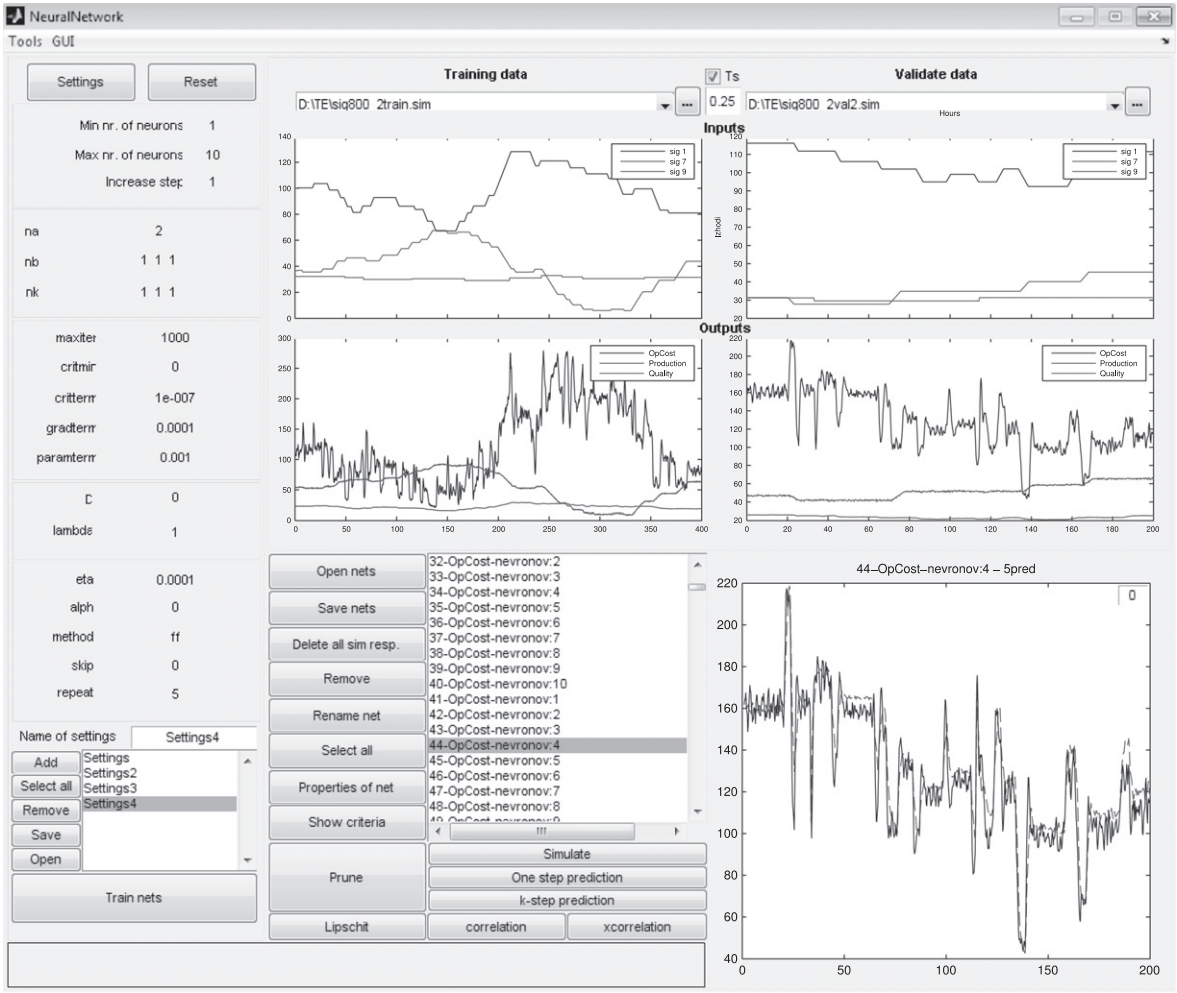


Fig. 6. Main window of the modelling assistant.

Furthermore, to also guarantee model quality for long-term operation, the performance of the HPC model needs to be monitored. The developed modelling-assistant tool enables the users to build and validate the model predictions with recent production data. Such a verification of the model alerts the user that a different model should be selected from the base of previously generated models or a new model should be identified from the newly acquired production data.

The main structure of the program can be viewed from Fig. 6, where the main window of the modelling assistant is shown.

The modelling assistant is based around the NNSYSID toolbox [41], where neural-network-related routines are implemented. This toolbox was applied, as it is specialised for system identification and as it tends to simplify neural-model derivation, with the reduction of many tuning parameters. The NNSYSID toolbox generally supports many model structures (NARX, NARMAX, NOE, NSSIF, NIOL etc.), different network-training algorithms and methods to prevent overfitting (regularisation and pruning algorithms). A complete overview of the implemented methods is available in [41].

5. A pPI model of the Tennessee Eastman benchmark process

To test the HPC concept an appropriate case study is needed. Our work is based on the well-known benchmark model of a real industrial plant, Tennessee Eastman (TE), which can be viewed as a complete and complex production process with many measured and manipulative variables.

5.1. Short introduction of TE process

The TE benchmark process was introduced by Downs and Vogel [42] as a model of a real chemical production process. The model represents a test problem for researchers to experiment with different control-related solutions. As depicted in Fig. 7, the process consists of five major units: a chemical reactor, a product condenser, a vapour–liquid separator, a product stripper and a recycle compressor. Four reactants (A, C, D, E) and an inert component (B) are entering the process, where four exothermic, irreversible reactions result in two products (G, H) and one byproduct (F). The process products leave the process through stream 11, where they are separated in a downstream refining section. The production process has 41 measured variables (y) and 12 different manipulative variables (u).

A specific combination of the production rate and/or the product mix are usually demanded by the market or some capacity limitations. Therefore, six typical operational modes (see Table 1) are defined in the article [42].

The model also provides 20 different process disturbances (for details see [42]), which imitate the disturbances typical of real TE production.

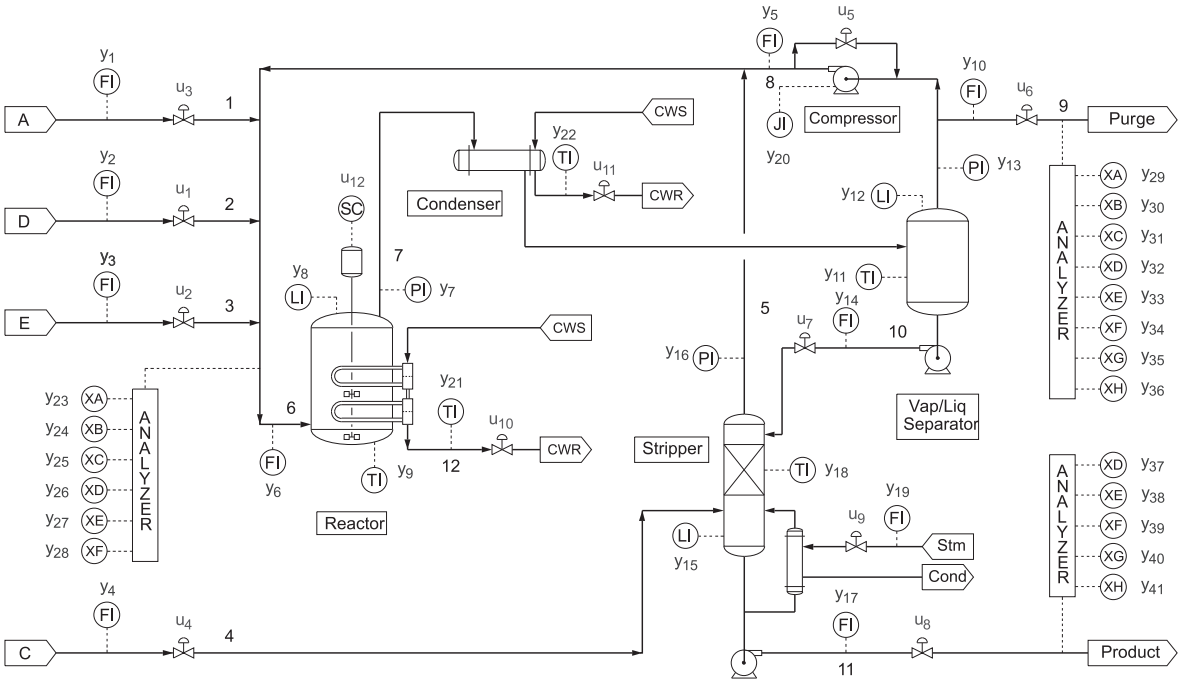


Fig. 7. Production scheme of the Tennessee Eastman process.

Table 1
List of the Tennessee Eastman production modes defined by Downs and Vogel [42].

Mode	G/H mass ration	Production rate (stream 11)
1	50/50	7038 kg h^{-1} G and 7038 kg h^{-1} H
2	10/90	1408 kg h^{-1} G and 12,669 kg h^{-1} H
3	90/10	10,000 kg h^{-1} G and 1111 kg h^{-1} H
4	50/50	Maximum
5	10/90	Maximum
6	90/10	Maximum

5.2. Data preprocessing

The data preprocessing step can be skipped for our case study, as we are working with the simulation model, where measurements have no outliers or other severe data anomalies. However, the process measurements do still have some intensionally added noise, typical for the specific measurement [42].

5.3. Definition of pPIs

The economic objectives can be translated to the production PIs, i.e., information related to the plant's efficiency that has to be extracted from a vast amount of available process measurements. With a holistic production control, where the pPIs are controlled, these economic objectives are optimised.

As the first pPI, an estimation of the production *Cost* is selected. The formulation of the cost function was already introduced by the authors of the TE model [42]. The costs are calculated from the process measurements in units of $\$/h$ and are formulated as shown in (5). The first row of the equation evaluates the costs of the compressor work and steam expenses, while other rows evaluate the loss of the components leaving the process with the product and purge.

$$\begin{aligned} Cost = {} & 0.0318\,\frac{\$}{\mathrm{kW\,h}}\cdot y_{19} + 0.0536\,\frac{\$}{\mathrm{kg}}\cdot y_{20} + 0.0921\frac{\mathrm{kg\,mol}}{\mathrm{m^3}} \\ & \cdot y_{17}\left[22.06\frac{\$}{\mathrm{kg\,mol}}\cdot y_{37} + 14.56\frac{\$}{\mathrm{kg\,mol}}\cdot y_{38} + 17.89\frac{\$}{\mathrm{kg\,mol}}\cdot y_{39}\right] \\ & + 0.04479\frac{\mathrm{kg\,mol\,h^{-1}}}{\mathrm{ks\,cm\,h}}\cdot y_{10}\left[2.209\frac{\$}{\mathrm{kg\,mol}}\cdot y_{29} + 6.177\frac{\$}{\mathrm{kg\,mol}}\cdot y_{31} + 22.06\frac{\$}{\mathrm{kg\,mol}}\right. \\ & \cdot \quad \left. y_{32} + 14.56\frac{\$}{\mathrm{kg\,mol}}\cdot y_{33} + 17.89\frac{\$}{\mathrm{kg\,mol}}\cdot y_{34} + 30.44\frac{\$}{\mathrm{kg\,mol}}\cdot y_{35} + 22.94\frac{\$}{\mathrm{kg\,mol}}\cdot y_{36}\right] \end{aligned} \tag{5}$$

Next, we want to express the productivity of the process. The definition of this pPI is quite straightforward, as the quantity of product leaving the process is directly measured ($Production = y_{17}$). Directly from the process objectives an indicator for the process quality is also derived, since the product quality can be viewed as a desired mass ratio between the two final products, G and H ($Quality = y_{40}$).

5.4. Selection of model inputs

The TE process is a highly unstable system, and without low-level process control it exceeds the process safety limits and automatically shuts downs within an hour. We used the system that was stabilized with the low-level control presented in the work of Larsson et al. [43], where nine outputs are controlled with cascade loops.

Table 2
Process manipulative variables.

Notation	Controlled variable setpoints
F_p	Production rate index
R_2	Striper level
R_3	Separator level
R_4	Reactor level
R_5	Reactor pressure
R_7	%C in purge
R_8	Recycle rate
R_9	Reactor temperature
r_2	D/E feed rates

In order to use this problem for a demonstration of the HPC concept, a few changes to the low-level control were needed. As *Production* and *Quality* are already controlled, the original control scheme had to be slightly modified. *Production rate* loop and *Product ratio* loop were removed and two new manipulative variables were defined (i.e. F_p and r_2). In this way the control of the *Quality* and *Production* was intentionally moved from the process level to the production level, where the HPC is being realised.

The TE process with a modified low-level control has nine manipulated variables (see Table 2), which should be employed to realise the objectives dictated by the systems from the business layer. To simplify the model structure and to reduce the optimisation problem, only the inputs with the greatest impact on the outputs have to be determined. For such an analysis

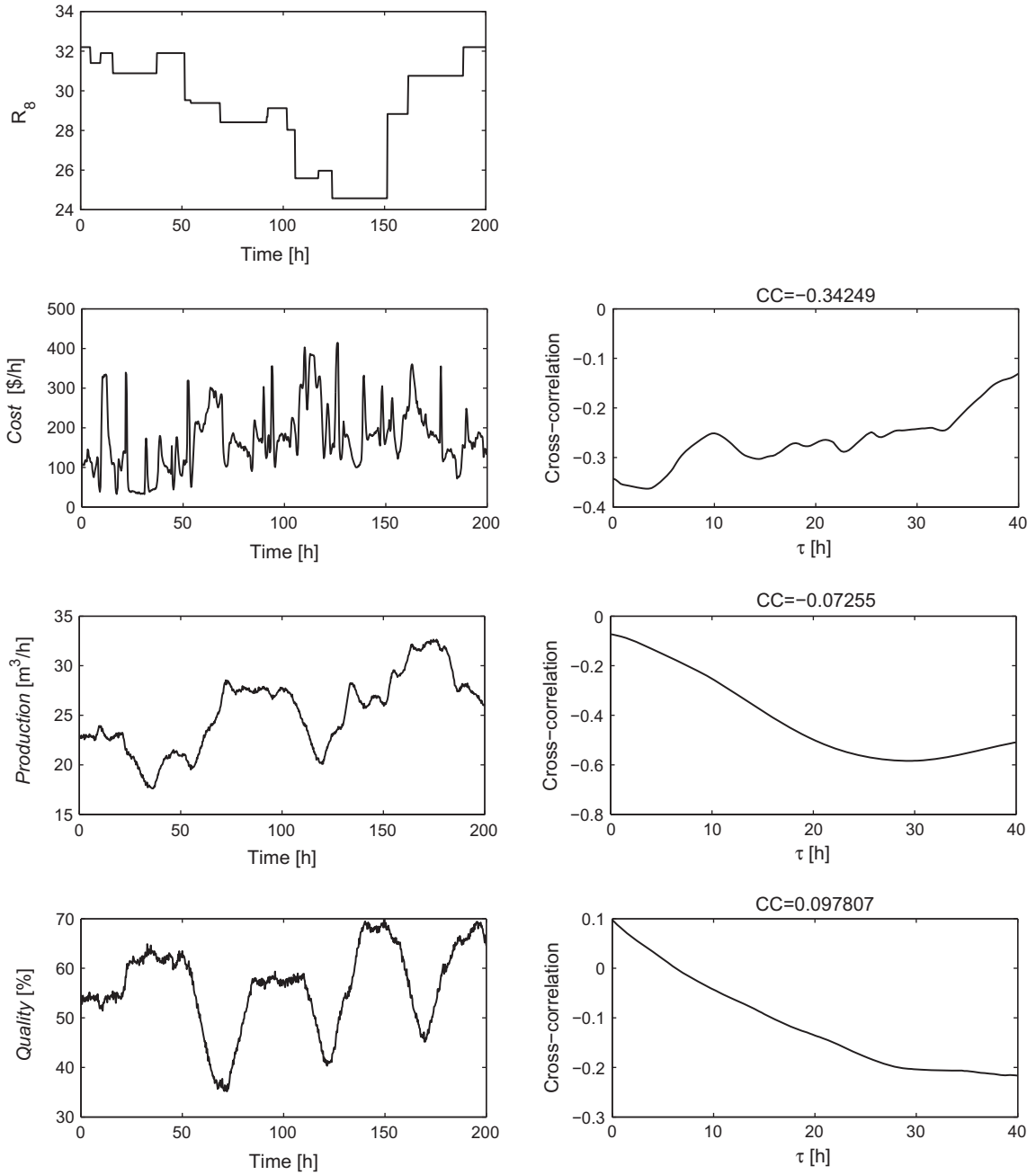


Fig. 8. Cross-correlation analyses between input R_8 and each of the outputs are presented in the last row. The figure is also depicts the cross-correlation coefficient (CC), indicating a match of the tested signals.

we do not need to perform an exhaustive subset selection, instead a simple variable-ranking method of cross-correlation is employed for each of the pPIs.

The experiment on the process is performed, where each input signal is changing over the vast operating range of the tested variable. The input–output signal of 200 h is obtained, where all the potential inputs are randomly changing. A cross-correlation analysis is then applied to the simulation data for every input and output. Fig. 8 illustrates part of the performed analysis, where the relations between the input R_8 and all three outputs are evaluated with the cross-correlation criterion. According to the scores of the cross-correlation coefficients, the following inputs were selected: F_p, R_8 and r_2. The model structure is therefore reduced to the three most influential inputs and three outputs (pPIs of the TE process).

5.5. Derivation of the neural network model

Based on the determination of the pPIs and the most influential inputs an input–output structure of the model to be derived was defined, as depicted in Fig. 9a Fig. 9b illustrates the inner structure of the model with respect to the TE process used.

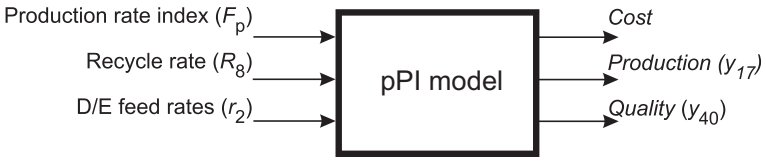
In the next step the derivation of the model structure and parameters using neural networks will be presented.

A dynamical model structure should be assumed, as the response of the production process does not depend only on the current input variables, but also on the current states of the process. A nonlinear autoregressive exogenous input (NARX) is applied, where previous records for the inputs and also the outputs are taken into consideration [37]. The regression vector, which represents the first mapping of the model (c.f. Section 3.4), is defined as:

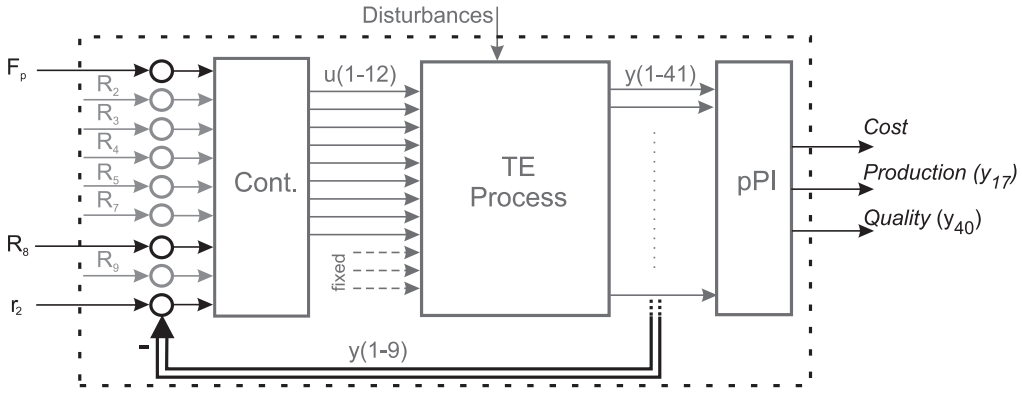
$$\varphi(t) = [y(t-1), \ldots c, y(t-n_a), u(t-1), \ldots c, u(t-n_b)]^T \tag{6}$$

In contrast to some other, more comprehensive, model structures, the NARX predictor is stable, despite possible modelling errors. The stability issue of a prediction is satisfied as there is a pure algebraic relationship between the prediction and past measurements of the outputs and inputs [44].

To estimate the proper model order (i.e., to specify n_a and n_b), a model-free, subset-selection method is applied. With the use of the Lipschitz method [36], the delayed inputs and outputs for the NARX model can be determined solely on the basis of the input–output data, as the optimal smoothness of the mapping, with respect to model order, is examined. As noise is usually present in the signal, it is hard to resolve the exact order of the process. However, the region of the exact solution can



(a) The block scheme of the input-output model to be derived.



(b) The inner structure of the model with the modified low-level control of the TE process and the corresponding pPI calculation (determination) block.

Fig. 9. Structure of the pPI model for the TE process.

Table 3
Modelling settings.

Option	Value
Number of delayed outputs as regressors	$n_a = 1$
Number of delayed inputs as regressors	$n_b = 2$ and 3
Input time delay	$n_k = 1$
Number of neurons in hidden layer	From 1 to 10
Regularisation parameter	$D = 0;\ 10^{-5};\ 10^{-4};\ 10^{-3}$

still be determined. The method is applied to the training dataset and a regression vector with two or three delayed inputs and one delayed output is selected.

A simulated historical run in the time range of 800 h is recorded, where the most influential inputs are changing, in order to reach different operating conditions. The simulated historical run is further divided into three individual signals, which will be later used to find an appropriate model and to test the quality of the model with the data not used in the training.

The neural network toolbox (NNSYSID [41]) used within the modelling assistant is limited to MISO systems. Therefore, three different models have to be identified, one for each output (*Cost*, *Production* and *Quality*).

In the search for the best model, various architecture-related parameters were tested (see Table 3). With the use of the modelling assistant, 80 models with alternative network structures are automatically trained for each of the outputs. The network training was performed with the Levenberg-Marquadt [44] learning algorithm because of its fast convergence for moderate-sized, feedforward neural networks. To enhance the convergence to the global minimum, prior to the training some small random weights and biases are assumed. Additionally, the modelling assistant scaled the process data to a mean value of 1 and a standard deviation of zero.

To enhance the generalisation capabilities of the acquired models, the stopping criteria and the weight-decay regularisation [44] were applied during the training. With the help of the modelling assistant the alternative weight decay settings D (see Table 3) are considered. Furthermore, the irrelevant connections in the network structure were removed, using OBS pruning [45]. The optimal network structure from the pruning procedure was selected with the validation on the second part of the dataset. According to the validation with the MSE criterion of the one-step-ahead prediction and simulation response from each network, two additional pruned networks were found, one for each validation response. A total of 240 potential network models were identified for each output in our case study.

5.6. Model validation

Many neural networks were generated, and among them those with the best generalisation capabilities need to be identified. As models will be used in the model-based control and optimisation, a few-steps-ahead predictions will be needed. Therefore, the models are validated with 10- and 20-steps-ahead prediction on the validation process signal. The difference between the TE process and the neural network model responses are evaluated with the MSE criterion. With respect to such a validation criterion for each output the best model is selected. It can be seen from the topologies of the selected networks (Fig. 10), that the least important weights were completely removed and also some of the inputs were pruned off, as their contribution to the prediction is inadequate.

To assess the performance of the model in comparison to the process, three validation experiments were carried out.

The first experiment (Fig. 11) represents the simulation responses for the third part of the dataset, which was not used in the training procedure. As such it represents a test for the generalisation capabilities of the obtained model.

The results of the second experiment are depicted in Fig. 12, where the production process is changing in accordance with the different pre-defined production modes (cf. Table 1).

The influence of the process disturbances on the model predictions is tested in the last validation. The model inputs are set constant, while some disturbances are introduced in the process (t = 0–15 h – *disturbance 1*: step change in A/C ratio in C feed; t = 50–75 h – *disturbance 4*: step change of reactor cooling water inlet temperature; t = 100–115 h – *disturbance 8*: random variation in A, B, C feed composition; t = 150–175 h – *disturbance 12* – random variation in the condenser cooling-water inlet temperature and *disturbance 15*: random variation in condenser cooling-water valve). Fig. 13 depicts two-steps-ahead predictions for this validation dataset.

Table 4 compares the prediction performance for the presented model responses, where the performance indexes of the Normalised Prediction Error (NPE – Eq. (7)), Integral of Absolute Error (IAE – Eq. (8)) and standard deviation of the prediction error are evaluated.

$$NPE = \left[\frac{\sum_{t=1}^{N}(\hat{y}(t) - y(t))^2}{\sum_{t=1}^{N} y^2(t)}\right]^{1/2} \cdot 100\% \tag{7}$$

$$IAE = \sum_{t=1}^{N} |\hat{y}(t) - y(t)| \tag{8}$$

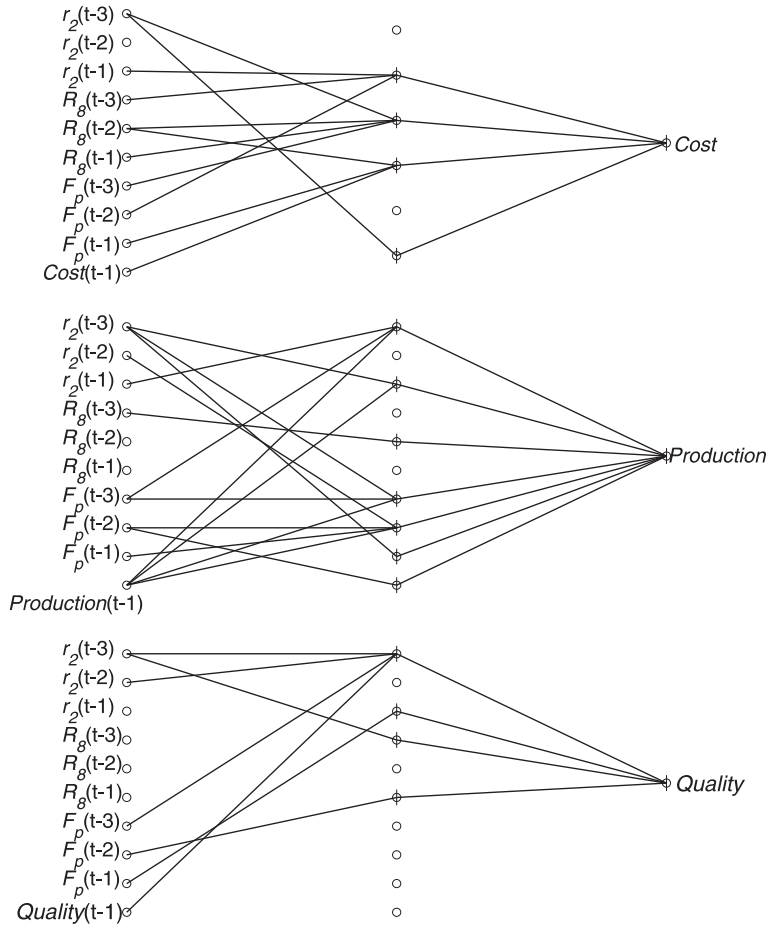


Fig. 10. Selected neural model architecture for *Cost*, *Production* and *Quality*.

From the model responses we can conclude that the *Production* and *Quality* PIs can be efficiently predicted. The long-term prediction of *Cost* was shown to be a more demanding task, since its definition is far more complex and it is connected with many process variables. Nevertheless, the resulting model describes satisfactorily the dominant dynamics and is prepared to be used for closed-loop control.

5.7. Model validation within production control

The practical usefulness of the model was also studied with more realistic validation tests. The model was used within the MPC controller to control a production process, which is represented by a complex non-linear model of the Tennessee Eastman process. In the following subsection a short introduction to the MPC strategy for HPC is presented. In subsection 5.7.2, the control results are presented and compared with the results obtained by other authors.

5.7.1. Production control

The purpose of the production model is to forecast the behaviour of a production process and to be used in production control. The model's knowledge about the system should be used to define the optimal values for manipulative variables, which would help the production manager to realise the business objectives more efficiently. This simplified process knowledge does not provide a detailed insight into the production process, but can still contribute to a better realisation of the business objectives and can help production managers with their decisions.

The most widely applied model-based control is the realisation of receding horizon control [46,47]. Such model predictive control (MPC) iteratively solves the finite-horizon optimisation of a plant model under process constraints. Only the control solution for the first time instance $u(k+1)$ is applied to the process, then the plant calculations are repeated, starting from the new model states, which are equivalent to the current process state measurements. In each iteration step the optimal solution of the following cost function should be calculated:

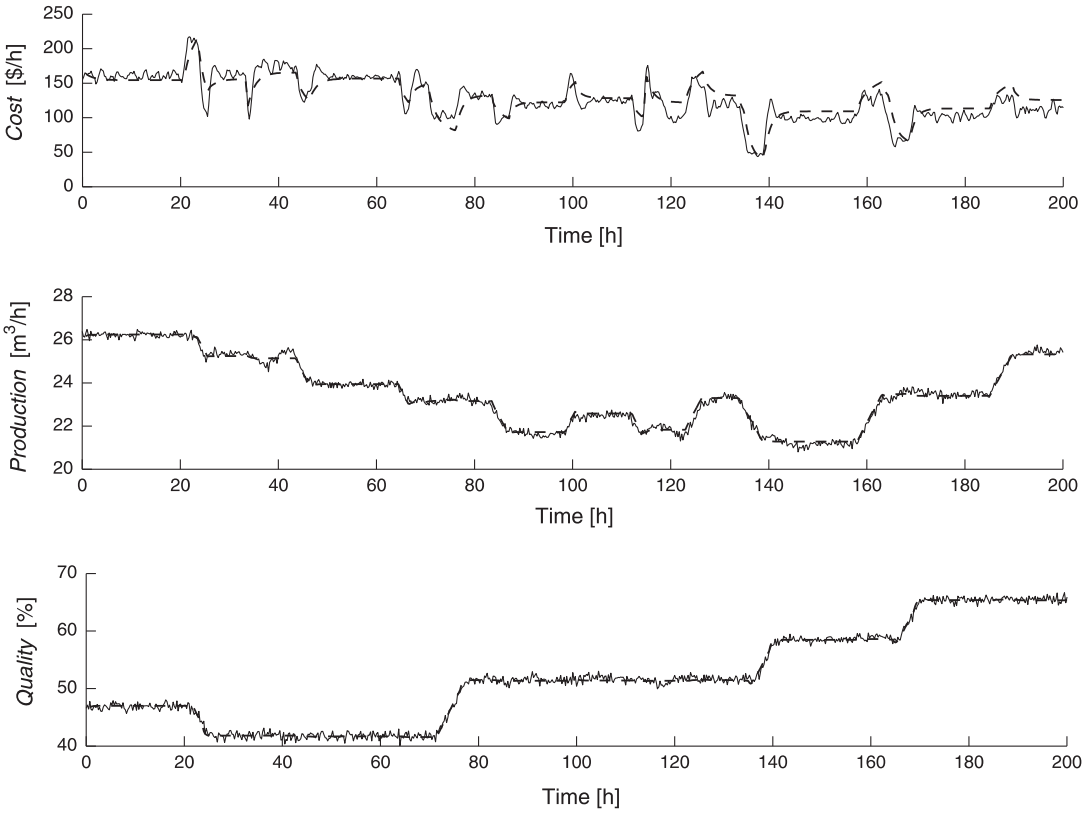


Fig. 11. Time responses of the selected model on a validation dataset. The solid line represents the actual response of the TE process and the dashed line the *simulation response* of the model.

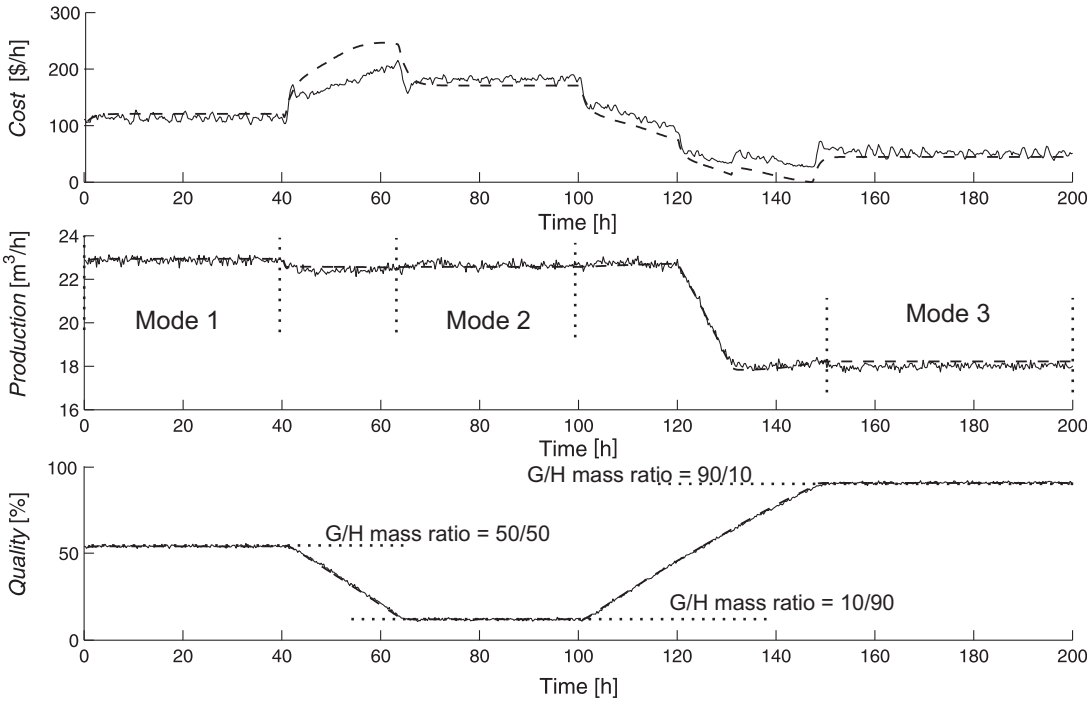


Fig. 12. Validation of the models on the dataset, where the process operates close to different production modes (see Table 1). The solid line represents the actual response of the TE process and the dashed line the *simulation response* of the model.

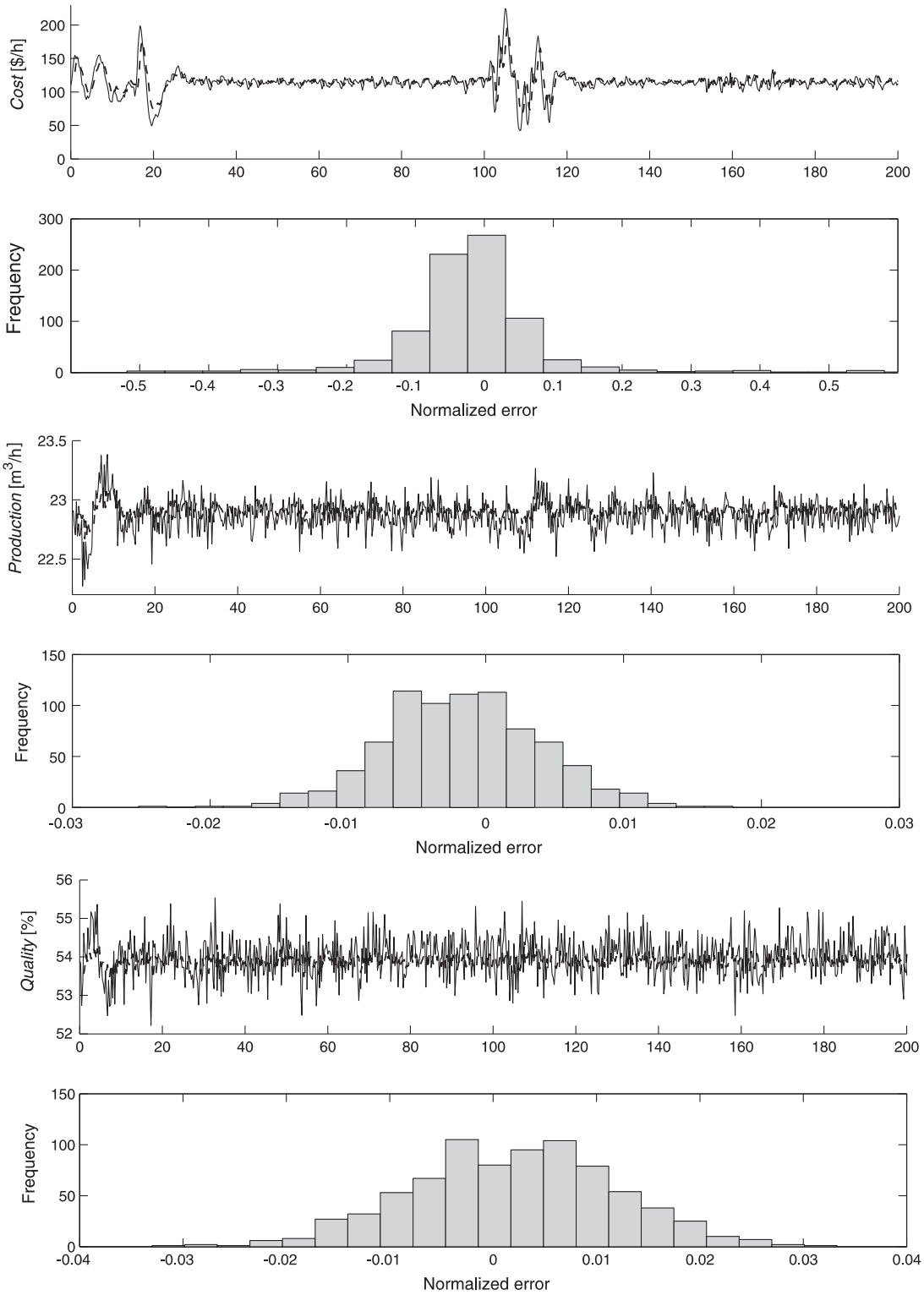


Fig. 13. Model validation on the dataset with constant model inputs and added process disturbances [42]. Solid line represents actual response of the process, dashed line represents *2-step ahead prediction* of the model. The histograms of normalised model errors ($NE = (\hat{y} - y)/\bar{y}$) depict the mismatch between model and process.

Table 4
Evaluation of NPE, IAE and standard deviation of the error for different validation datasets.

	Cost			*Production*			*Quality*		
	NPE	IAE	Std.	NPE	IAE	Std.	NPE	IAE	Std.
Fig. 11	10.6722	8748.93	13.8115	0.7010	104.89	0.1671	1.0534	353.69	0.5482
Fig. 12	16.7905	12348.62	20.2617	0.8579	116.59	0.1732	1.0294	389.35	0.6218
Fig. 13	10.8104	6116.54	12.3397	0.5906	86.36	0.1293	0.9985	346.70	0.5292

$$J = \sum_{j=1}^{H_p} \|\hat{y}_s(k+j) - r(k+j)\|^2_{Q_s} + \sum_{j=1}^{H_p} \|\hat{y}_m(k+j)\|^2_{Q_m} + \sum_{j=1}^{H_c} \|\Delta u(k+j-1)\|^2_{Q_u} \tag{9}$$

where $\hat{y}_s$ and $\hat{y}_m$ refer to the setpoint-controlled and minimisation-oriented pPIs (in Fig. 2 marked as K), respectively; r is a vector of the desired business objectives for setpoint-controlled pPIs (K^*); and Δu marks the change of the manipulating signal (ΔK_u), which is defined as $\Delta u(k+j) = u(k+j) - u(k+j-1)$ for time instance j. The operators $\|\cdot\|_{Q_s}$, $\|\cdot\|_{Q_m}$ and $\|\cdot\|_{Q_u}$ represent the weighted Euclidean distances, with weights Q_s, Q_m and Q_u. The weight Q_u penalises relatively big changes in control moves $u(k+j)$ and with the proper selection of weights Q_s and Q_m, the control objective priority for setpoint-controlled pPIs (y_s) and minimisation-oriented pPIs (y_m) can be adjusted.

Note, that within the HPC concept, the calculated control moves represent only the suggestion, given as a decision support to help the production manager and in practice an operator has the possibility to interact. In the frame of our experiment we have, however, designed a classical MPC controller.

An MPC controller with the following characteristics was implemented. The prediction horizon was chosen to be $H_p = 17$ time samples and the control horizon $H_c = 8$. To decrease the computational burden, the input blocking (IB) technique [48] was applied, where the control signal is set to be constant over a few time instances in the optimisation routine. The blocking setting was selected as [1 1 2 2 2 3 3 3], where each element of the vector indicates the number of consecutively constant time samples. To obtain the offset-free tracking in steady-state conditions, the difference in the model response and the process measurements was considered as a constant disturbance on the model output [49].

5.7.2. Simulation experiments and evaluation of results

To verify the usability of the model we carried out several simulation experiments. The considered pPI model consists of two setpoint-controlled pPIs (*Production*, *Quality*) and one minimisation-oriented pPI (*Cost*). In the first control experiment, *Production* and *Quality* are controlled in accordance with pre-defined production modes and the *Cost* minimisation objective is not considered. The pPI responses and reference values are shown in Fig. 14. It is clear that the *Production* and *Quality* pPIs follow the reference values.

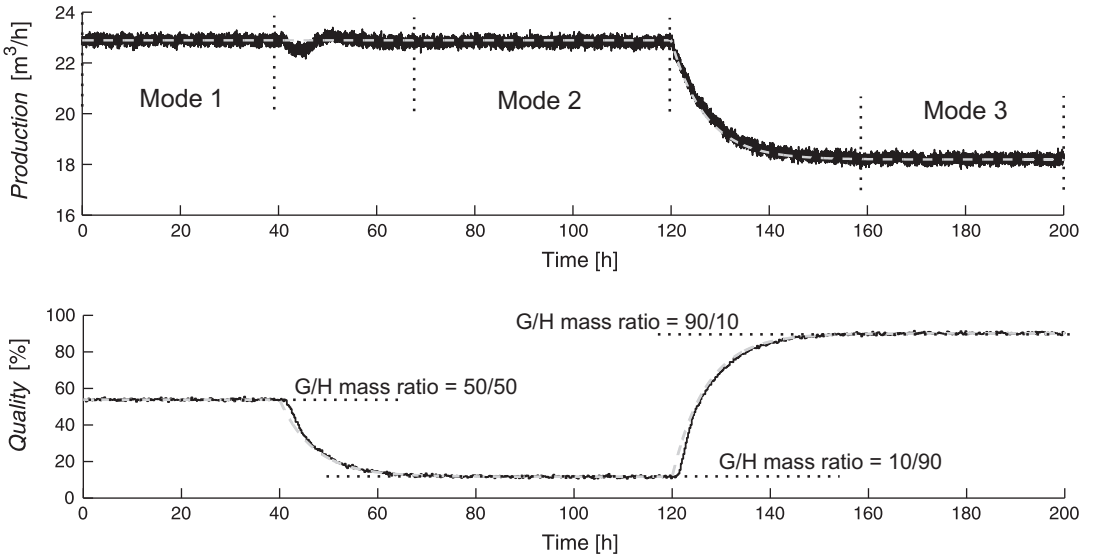


Fig. 14. First control experiment, where the process is controlled in accordance with different pre-defined production modes. The solid black line represents the response of the TE process and the dashed bright-grey line the control setpoints. The figure is plotted with the sample rate of the TE process ($T_{s_{fig}} = 0.01$).

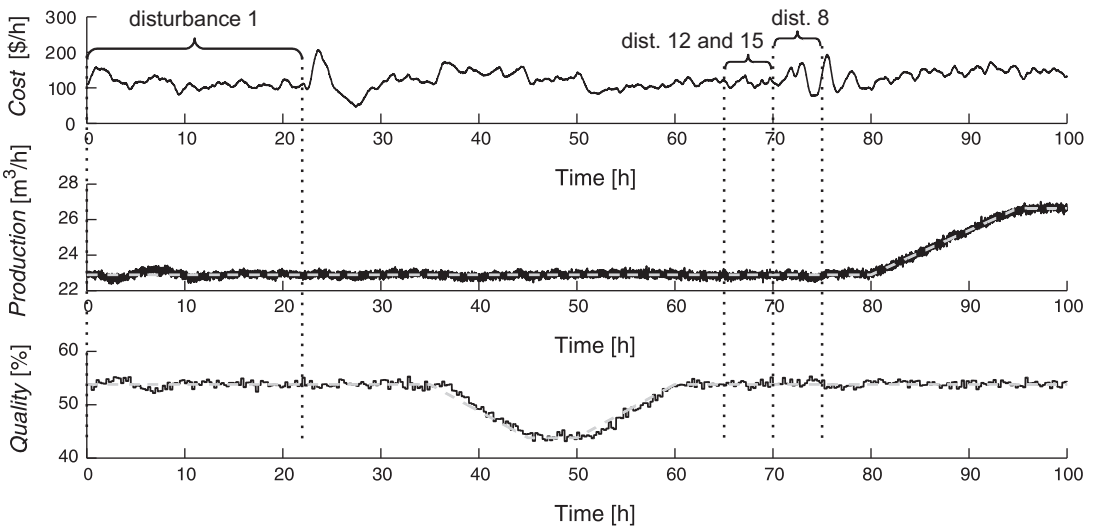


Fig. 15. Second control experiment from the work of Larsson et al. [43]. The process is controlled without a cost-minimisation objective ($Q_m = 0$) and the figure is plotted with $T_{s_{fig}} = 0.01$.

Table 5
Validation of second control experiment, where the *Cost* minimisation priority (Q_m) was gradually increased.

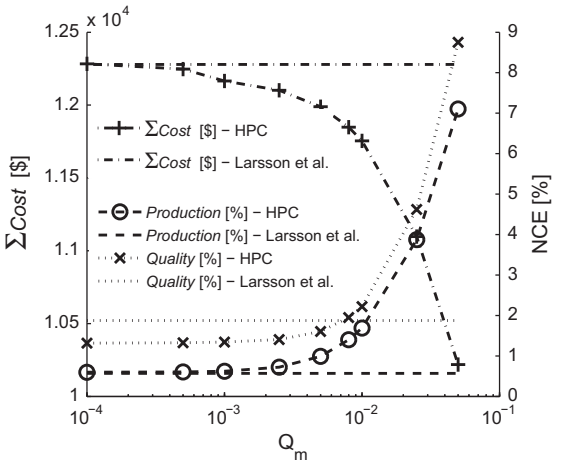
Method	$\sum(Cost)$ [$]	*Production*			*Quality*		
		NCE (%)	IAE	Std.	NCE (%)	IAE	Std.
Larsson et al. [43]	12278.54	0.5680	1055.54	0.1327	1.8766	7356.96	0.9797
HPC							
$Q_m = 0$	12287.81	0.5979	1107.46	0.1398	1.3123	5484.03	0.6832
$Q_m = 0.0005$	12247.15	0.6058	1123.09	0.1402	1.3255	5540.10	0.6849
$Q_m = 0.001$	12165.36	0.6227	1157.09	0.1403	1.3427	5612.19	0.6858
$Q_m = 0.005$	11988.89	0.9902	1956.40	0.1427	1.6063	6835.21	0.6940
$Q_m = 0.008$	11848.36	1.3992	2935.21	0.1497	1.9499	8536.91	0.7128

With the second control scenario we compared the effectiveness of our control with the results presented in the work of Larsson et al. [43]. The setpoints for *Production* and *Quality* are varied around the first operating mode. In addition, four different process disturbances are turned on during the experiment, as marked in Fig. 15.

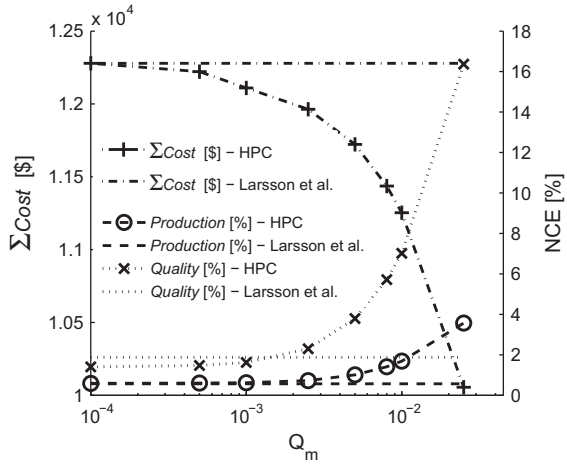
To evaluate the control performance of the setpoint-controlled pPIs, the Normalised Control Error (NCE), IAE and standard deviation of the error are observed. The criteria are defined as displayed in Eqs. (7) and (8), where NCE is normalised with the reference value (r) and the control error is defined as $e(t) = y(t) - r(t)$. In addition, overall costs are evaluated as the accumulation of the cost through the overall experiment.

The case when there is no direct minimisation of the *Cost* ($Q_m = 0$) and the objectives for *Prouction* and *Quality* are set as equally important, is shown in Fig. 15. This test run can be directly compared with the results published in [43]. According to the criteria presented in Table 5, the control of *Quality* achieved much better tracking characteristics, while achieving comparable *Production* control performance and similar overall *Costs*. In the next step of the validation, the *Cost* reduction objective (Q_m) is gradually increased. Different control runs are compared in Table 5. A graphical representation of the results is also shown in Fig. 16, where a comparison with the results published by Larsson et al. [43] is presented. In Fig. 16, the second control scenario is presented for the additional case, when the *Production* control objective is set as more important than the tracking control of *Quality* pPI. From the figures it is clear that HPC drives the process to the working conditions, which satisfy the objectives dictated from the business level.

The results show that it is possible to influence the overall costs of the process, with the introduction of a simple pPI model. The increased importance of the *Cost* minimisation objective results in lower accumulated *Costs* over the experiment. As expected, it is clear from Fig. 16, that lower *Costs* result in less accurate performance for *Production* and *Quality*, as the process has moved to the more *Cost* effective working condition. The presented experiments show that the prediction capabilities of the developed model are sufficient.



(a) $Q_s = [1\ 1]$



(b) $Q_s = [1\ 0.3]$

Fig. 16. Results of the second control experiment, where the *Cost* minimisation objective (Q_m) is gradually increased. Q_s weight refers to the importance of the tracking control objective for *Production* and *Quality*, respectively.

6. Conclusion

In this paper the issues regarding the modelling procedure for holistic production control are studied. The procedure consists of several steps: data preprocessing, determination of pPIs, input selection and black-box modelling. We have demonstrated the production modelling procedure in a case study of the Tennessee Eastman process. Here, the most influential inputs are adequately defined with the cross-correlation analysis and the neural networks are used to identify the model.

The comparison of the derived model responses and the original production process data showed that the model is appropriate. Additional validation of the model was performed, where model predictive control was realised in order to control *Cost*, *Production* and *Quality*. The results show that with the presented modelling methodology it is possible to obtain dynamical pPI models, which could be used within the HPC concept.

We have also noticed that the problem of inputs selection appears to be one of the most challenging problems in the pPI modelling procedure. Therefore, further research should be focused on an evaluation of the appropriate methods, where a determination of the input–output causal relations and a ranking of the inputs according to their sensitivity to the pPIs, should be performed.

Acknowledgements

The work has been supported by the Competence Centre for Advanced Control Technologies which is partly financed by the Republic of Slovenia, Ministry of Higher Education, Science and Technology and European Union (EU) – European Regional Development Fund.

References

[1] M.L. Luyben, B.D. Tyreus, W.L. Luyben, Plantwide control design procedure, AIChE Journal 43 (1997) 3161–3174.
[2] Sigurd Skogestad, Plantwide control: Towards a systematic procedure, in: J. Grievink, J. van Schijndel (Eds.), European Symposium on Computer Aided Process Engineering – 12, Computer Aided Chemical Engineering, vol. 10, Elsevier, 2002, pp. 57–69.
[3] G. Morel, P. Valckenaers, J.-M. Faure, C.E. Pereira, C. Diedrich, Manufacturing plant control challenges and issues, Control Engineering Practice 15 (2007) 1321–1331.
[4] J. Harding, M. Shahbaz, Srinivas, A. Kusiak, Data mining in manufacturing: a review, Journal of Manufacturing Science and Engineering 128 (2006) 969–976.
[5] A. Kusiak, Data mining in design of products and production systems, in: IFAC Annual Reviews in Control, pp. 147–156.
[6] A. Choudhary, J. Harding, M. Tiwari, Data mining in manufacturing: a review based on the kind of knowledge, Journal of Intelligent Manufacturing 20 (2009) 501–521.
[7] S. Engell, Feedback control for optimal process operation, Journal of Process Control 17 (2007) 203–219.
[8] J. Kadam, W. Marquardt, Integration of economical optimization and control for intentionally transient process operation, in: R. Findeisen, F. Allgöwer, L. Biegler (Eds.), Assessment and Future Directions of Nonlinear Model Predictive Control, Lecture Notes in Control and Information Sciences, vol. 358, Springer, Berlin, Heidelberg, 2007, pp. 419–434.
[9] J. Rawlings, R. Amrit, Optimizing process economic performance using model predictive control, in: L. Magni, D. Raimondo, F. Allgöwer (Eds.), Nonlinear Model Predictive Control, Lecture Notes in Control and Information Sciences, vol. 384, Springer, Berlin, Heidelberg, 2009, pp. 119–138.
[10] H.-P. Wiendahl, J.-W. Breithaupt, Automatic production control applying control theory, International Journal of Production Economics 63 (2000) 33–46.
[11] L. Biegler, I. Grossmann, Retrospective on optimization, Computers & Chemical Engineering 28 (8) (2004) 1169–1192.
[12] S. Zorzut, V. Jovan, D. Gradišar, G. Mušič, Closed-loop control of a polymerisation plant using production performance indicators (PIs), International Journal of Computer Integrated Manufacturing 22 (12) (2009) 1128–1143.
[13] ISO/WD 22400-1: Automation Systems and Integration – Key Performance Indicators for Manufacturing Operations Management – Part 1: Overview, Concepts and Terminology, 2010.
[14] IBM, Business Monitor, 2012 <http://www-01.ibm.com/software/integration/business-monitor/> (accessed 23.08.12).
[15] M. Abe, J.-J. Jeng, Y. Li, A tool framework for KPI application development, in: IEEE International Conference on e-Business Engineering, pp. 22–29.
[16] D. Sands, G. MacElroy, Asset management KPIs, validated, analysed and predicted utilising self learning hybrid modelling in real time, in: J. Mathew, J. Kennedy, L. Ma, A. Tan, D. Anderson (Eds.), Engineering Asset Management, Springer, London, 2006, pp. 757–766.
[17] R. Revetria, F. Tonelli, Neural networks and regressive KPI metamodels for business corporate management: methodology and case study, in: P. Taticchi (Ed.), Business Performance Measurement and Management, Springer, Berlin, Heidelberg, 2010, pp. 343–356.
[18] G. Fanuc, Proficiy Troubleshooter, 2012 <http://www.ge-ip.com/products/3391> (accessed 23.08.12).
[19] Siemens, Production Data Management/MES, 2012 <http://www.industry.siemens.com/services/global/en/cross_industry_solutions/automation_it/mes/Pages/Default.aspx> (accessed 23.08.12).
[20] R.K. Pearson, Nonlinear empirical modeling techniques, Computers & Chemical Engineering 30 (2006) 1514–1528.
[21] J. Kocijan, A. Girard, B. Banko, R. Murray-Smith, Dynamic systems identification with gaussian processes, mathematical and computer modelling of dynamical systems: methods, Tools and Applications in Engineering and Related Sciences 11 (2005) 411–424.
[22] D. Gradišar, M. Glavan, J. Kocijan, S. Strmčnik, Modelling case study for holistic production control, in: Management and Control of Production and Logistics, 2010.
[23] R. Martin, D. Thomson, Robust-resistant spectrum estimation, Proceedings of the IEEE 70 (1982) 1097–1115.
[24] H. Liu, S. Shah, W. Jiang, On-line outlier detection and data cleaning, Computers & Chemical Engineering 28 (2004) 1635–1647.
[25] P. Heinonen, Y. Neuvo, Fir-median hybrid filters, IEEE Transactions on Acoustics, Speech and Signal Processing 35 (1987) 832–838.
[26] R. Pearson, Outliers in process modeling and identification, IEEE Transactions on Control Systems Technology 10 (2002) 55–63.
[27] P. Folan, J. Browne, A review of performance measurement: towards performance management, Computers in Industry 56 (2005) 663–680.
[28] J. Gerry, G. Buckbee, The link between automation and enterprise KPIs, ExpertTune, Plant Performance Supervision & PID Tuning Software, White Paper, 2005.
[29] G. Smits, A. Kordon, K. Vladislavleva, E. Jordaan, M. Kotanchek, Variable selection in industrial datasets using pareto genetic programming, in: T. Yu, R. Riolo, B. Worzel (Eds.), Genetic Programming Theory and Practice III, The Kluwer International Series in Engineering and Computer Science, vol. 9, Springer, 2006, pp. 79–92.
[30] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, Journal of Machine Learning Research 3 (2003) 1157–1182.
[31] I.T. Jolliffe, Principal Component Analysis, second ed., Springer Verlag, New York, 2002.
[32] L. van der Maaten, E. Postma, J. van den Herik, Dimensionality Reduction: A Comparative Review, Tech. Report TiCC TR 2009-005, Tilburg University, Netherlands, 2009.
[33] C. Hu, F. Wan, Input selection in learning systems: a brief review of some important issues and recent developments, in: IEEE International Conference on Fuzzy Systems, pp. 530–535.
[34] R. Sindelár, R. Babuska, Input selection for nonlinear regression models, IEEE Transactions on Fuzzy Systems 12 (2004) 688–696.
[35] T. Fernando, H. Maier, G. Dandy, Selection of input variables for data driven models: an average shifted histogram partial mutual information estimator approach, Journal of Hydrology 367 (2009) 165–176.
[36] X. He, H. Asada, A new method for identifying orders of input–output models for nonlinear dynamic systems, in: American Control Conference, 1993, pp. 2520–2523.
[37] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Delyon, P.-Y. Glorennec, H. Hjalmarsson, A. Juditsky, Nonlinear black-box modeling in system identification: a unified overview, Automatica 31 (1995) 1691–1724.
[38] L. Ljung, System Identification: Theory for the User, second ed., Prentice Hall, Upper Saddle River, NJ, 1999.
[39] M. Meireles, P. Almeida, M. Simoes, A comprehensive review for industrial applicability of artificial neural networks, IEEE Transactions on Industrial Electronics 50 (2003) 585–601.
[40] S. Haykin, Neural Networks: A Comprehensive Foundation, second ed., Prentice Hall, Upper Saddle River, NJ, 1999.
[41] M. Nørgaard, Neural Network Based System Identification Toolbox, Tech. Report 00-E-891, Department of Automation, Technical University of Denmark, 2000 <http://www.iau.dtu.dk/research/control/nnsysid.html> (accessed 23.08.12).
[42] J.J. Downs, E.F. Vogel, A plant-wide industrial process control problem, Computers & Chemical Engineering 17 (1993) 245–255.
[43] T. Larsson, K. Hestetun, E. Hovland, S. Skogestad, Self-optimizing control of a large-scale plant: the Tennessee Eastman process, Industrial & Engineering Chemistry Research 40 (2001) 4889–4901.
[44] M. Nørgaard, O.E. Ravn, N.K. Poulsen, L.K. Hansen, Neural Networks for Modelling and Control of Dynamic Systems: A Practitioner's Handbook, Springer Verlag, London, 2000.
[45] L.K. Hansen, M. Pedersen, Controlled growth of cascade correlation nets, in: Proceedings of the International Conference on Artificial Neural Networks, 1994, pp. 797–800.
[46] S.J. Qin, T.A. Badgwell, A survey of industrial model predictive control technology, Control Engineering Practice 11 (2003) 733–764.
[47] E. Camacho, C. Bordons, Nonlinear model predictive control: an introductory review, in: R. Findeisen, F. Allgöwer, L. Biegler (Eds.), Assessment and Future Directions of Nonlinear Model Predictive Control, Lecture Notes in Control and Information Sciences, vol. 358, Springer, Berlin, Heidelberg, 2007, pp. 1–16.
[48] F. Manenti, Considerations on nonlinear model predictive control techniques, Computers & Chemical Engineering 35 (2011) 2491–2509.
[49] J.M. Maciejowski, Predictive control with constraints, Prentice Hall, Essex, England, 2002.