



Challenges in the Development of Soft Sensors for Bioprocesses: A Critical Review

Vincent Brunner, Manuel Siegl, Dominik Geier* and Thomas Becker

Chair of Brewing and Beverage Technology, Technical University of Munich, Freising, Germany

Among the greatest challenges in soft sensor development for bioprocesses are variable process lengths, multiple process phases, and erroneous model inputs due to sensor faults. This review article describes these three challenges and critically discusses the corresponding solution approaches from a data scientist's perspective. This main part of the article is preceded by an overview of the status quo in the development and application of soft sensors. The scope of this article is mainly the upstream part of bioprocesses, although the solution approaches are in most cases also applicable to the downstream part. Variable process lengths are accounted for by data synchronization techniques such as indicator variables, curve registration, and dynamic time warping. Multiple process phases are partitioned by trajectory or correlation-based phase detection, enabling phase-adaptive modeling. Sensor faults are detected by symptom signals, pattern recognition, or by changing contributions of the corresponding sensor to a process model. According to the current state of the literature, tolerance to sensor faults remains the greatest challenge in soft sensor development, especially in the presence of variable process lengths and multiple process phases.

Keywords: soft sensor, online prediction, bioprocess, multiphase process, data synchronization, sensor fault, fault tolerance

OPEN ACCESS

Edited by:

Johannes Felix Buyel,
Fraunhofer Society (FHG), Germany

Reviewed by:

Karl Bayer,
University of Natural Resources and
Life Sciences Vienna, Austria

Astrid Duerauer,
University of Natural Resources and
Life Sciences Vienna, Austria

*Correspondence:

Dominik Geier
dominik.geier@tum.de

Specialty section:

This article was submitted to
Bioprocess Engineering,
a section of the journal
Frontiers in Bioengineering and
Biotechnology

Received: 08 June 2021

Accepted: 03 August 2021

Published: 20 August 2021

Citation:

Brunner V, Siegl M, Geier D and
Becker T (2021) Challenges in the
Development of Soft Sensors for
Bioprocesses: A Critical Review.
Front. Bioeng. Biotechnol. 9:722202.
doi: 10.3389/fbioe.2021.722202

INTRODUCTION

The biologization of the manufacturing industry is leading to more and more processes that were previously based on chemical synthesis being replaced by biotechnological processes (Buyel et al., 2017). At the same time, the digitalization of these processes is leading to more transparent, lower-risk, and more efficient biological manufacturing (Scheper et al., 2021). At the intersection of these two trends—biologization and digitalization—a multitude of new technologies and approaches have emerged in recent decades. These include, in particular, advances in the fields of data science as well as monitoring and control technology for bioprocesses (Steinwandter et al., 2019). With the introduction of the quality by design (QbD) and process analytical technology (PAT) initiatives, this development has received institutional support (FDA, 2004; Rathore and Winkle, 2009).

Despite advances in bioprocess monitoring, many relevant process variables are still determined offline using laboratory analyses. On this basis, a prediction is made about the expected future behavior of the process. However, this procedure is often not sufficient to effectively react to process changes, for example, through closed-loop control. The development of soft sensors is a remedy to this situation.

A soft sensor (“software sensor”) is a combination of process data (input) and a model that uses these input data to predict a target quantity (output). It is therefore an indirect measurement. The

input data used for the prediction are typically composed of signals from hardware sensors and actuators. Dependent on the degree of process knowledge that is implemented, the prediction model can be classified as data-driven, knowledge-based, or hybrid.

The application fields of soft sensors can be distinguished by the nature of the target quantity (Kadlec et al., 2009). The largest application field of soft sensors is the online prediction of physical quantities such as, for example, concentrations of biomass, substrate, intermediate, or product. These types of soft sensors are used when online analyzers are not available or economically feasible for process variables of interest. Further, soft sensors can be used within supervisory control applications to monitor the state of the process on a higher level and detect process faults (Liu et al., 2017; Besenhard et al., 2018; Dumarey et al., 2019). Soft sensors for process monitoring and process fault detection use historical process data to derive higher-level, non-physical process quantities such as latent variables (Kourtis, 2005) that indicate deviations from the normal process conditions. Finally, soft sensors can be used to detect sensor faults. The soft sensor here is used to predict the reading of a hardware sensor. A deviation of the prediction and the hardware sensor reading indicates a sensor fault (Brunner et al., 2019). The falsified hardware sensor reading can be reconstructed using the soft sensor's prediction.

The development of soft sensors poses several challenges to the data scientist. These challenges can be assigned to either the data, information, or knowledge domain. **Table 1** lists the most important challenges together with corresponding solution approaches. Most of these solution approaches have been reviewed for the process industry, including phase division (Yao and Gao, 2009), adaption mechanisms for soft sensors (Kadlec et al., 2011), JIT learning (Kano and Fujiwara, 2012; Saptoro, 2014), data synchronization (Ündey et al., 2002), process fault detection (Venkatasubramanian et al., 2003a; Venkatasubramanian et al., 2003b; Venkatasubramanian et al., 2003c), dimension reduction (Pani and Mohanta, 2011), variable selection (Cawley and Talbot, 2010; Souza et al., 2016; Heinze et al., 2018), sensor fault detection and fault tolerance (Isermann, 2006; Isermann, 2011; Das et al., 2012), identification of overfitting (Hawkins, 2004), model maintenance (Wise and Roginski, 2015), digitalization of expert knowledge (Birle et al., 2013), and hybrid modeling (Stosch et al., 2014; Solle et al., 2017).

A small number of these reviews address bioprocesses, but in their majority, they play only a tangential role. Several of the above approaches are equally applicable to bioprocesses (e.g., variable selection, dimensional reduction). However, what needs an updated review or has not yet been reviewed at all in the context of bioprocesses are the following three challenges:

- variable process lengths,
- multiple process phases, and
- sensor faults.

Especially for bioprocesses, these challenges often occur in combination, so that solution approaches are becoming increasingly complex: Sensor faults, which impede the

reliability of soft sensors, are more difficult to detect or compensate for in processes with variable lengths and dynamic behavior (Brunner et al., 2019); data synchronization (for processes of variable lengths) is more complex for multiphase processes (Doan and Srinivasan, 2008). The focus of this review is thus on the synchronous consideration of these three challenges of soft sensor development. This review aims to critically evaluate the corresponding solution approaches regarding their practicality and applicability to bioprocesses. The following applies here: As simple as possible, as complex as necessary.

This review article is structured as follows. First, an overview of the status quo in the development and online application of soft sensors is provided. Here, the typical steps of soft sensor development and the state of the art in online implementation are described. The following chapter concerns the challenges in soft sensor development for bioprocesses from a data scientist's perspective, namely, variable process lengths, multiple process phases, and sensor faults. The corresponding solution approaches are critically discussed. This chapter is followed by a conclusion that reveals the greatest remaining research gaps in soft sensor development for bioprocesses.

SOFT SENSORS: THE STATUS QUO

Soft sensors have become an important tool within the QbD/PAT framework, as reviewed by Mandenius and Gustavsson (2015), Randek and Mandenius (2018), and Rathore et al. (2021). One reason is that they are often the only means of determining critical process parameters (CPP) or critical quality attributes (CQA) online at all (Capito et al., 2015; Melcher et al., 2015; Sauer et al., 2019; Spann et al., 2019; Walch et al., 2019; Pais et al., 2020; Wasalathanthri et al., 2020a). Making these quantities measurable by means of soft sensors, in turn, allows CPPs or CQAs to be closed-loop controlled (Birle et al., 2015; Matthews et al., 2016; Voss et al., 2017; Brunner et al., 2020; Gomis-Fons et al., 2020). This type of control, also called inferential control, plays an important role in the automation of bioprocesses, since by far not all process quantities to be closed-loop controlled can be measured directly (Rathore et al., 2021).

As mentioned at the beginning, soft sensors are used to indirectly measure a target variable by combining a predictive model with corresponding input data. Process data used as input to soft sensors can compose differently depending on the organism (bacteria, yeast, filamentous fungi, mammalian or insect cells, etc.) used in upstream processing (USP) and the techniques used in downstream processing (DSP). Instrumentation of bioprocesses and thus possible input data for soft sensors have recently been reviewed by several authors (with varying emphases): Simon et al., 2015 (industrial application); Biechele et al., 2015 (USP, disposable technology); Mandenius and Gustavsson, 2015 (price, utility, and relevance of online analyzers for soft sensor development); Claßen et al., 2017 (spectroscopic sensors); Wasalathanthri et al., 2020b (spectroscopic sensors, chromatography, and mass spectrometry); Gargalo et al., 2020 (spectroscopic sensors, biosensors, and free-floating wireless sensors). Therefore, only

TABLE 1 | Overview of the most important challenges and corresponding solution approaches in the development of soft sensors. The challenges are herein broadly assigned to either the data, information, or knowledge domain.

Domain	Challenges	Solution approaches	Details and most important methods
Data	Multiple process phases	Phase detection and division	Algorithms for phase detection can be based on the shape of process trajectories (e.g., sharp peak in specific process variable) or the correlation structure of process variables (e.g., change in loading matrices of latent variable submodels) (Yao and Gao, 2009; Luo et al., 2016)
		Adaption mechanisms	The adaption of the prediction model to multiple process phases can be realized by moving window, recursive adaption, or ensemble-based methods (Kadlec et al., 2011). Just-in-time (JIT) learning is a special case of adaptive modeling, because the local JIT models are built during the online application (Kano and Fujiwara, 2012; Saptoro, 2014)
	Variable process lengths	Data synchronization	Datasets with variable process lengths can be aligned based on: indicator variable techniques, where a measured or computed variable (e.g., maturity index) indicates the progress of the process instead of time; curve registration techniques, where batch trajectories are aligned with respect to process landmarks (Ünçey et al., 2002); and dynamic time warping (DTW), where the data patterns are compressed and expanded so that similar features are aligned
		Adaption mechanisms	A prediction model for time-variant data with nonlinear behavior needs to be adaptive rather than static. Adaptive modeling approaches include moving window, recursive adaption, and ensemble-based methods (Kadlec et al., 2011) as well as JIT learning (Kano and Fujiwara, 2012; Saptoro, 2014)
	(Multi)collinearity	Dimension reduction	Latent variable methods (principal component analysis (PCA) or partial least squares (PLS) variants) intrinsically lead to a dimension reduction and thus eliminate (multi)collinearity (Pani and Mohanta, 2011)
		Variable selection	A sound variable selection can reduce (multi)collinearity. Approaches for variable selection include stepwise regression (e.g., backward elimination, forward selection), penalization of model complexity (e.g., based on least absolute shrinkage and selection operator), and through expert knowledge (e.g., a variables' variance is known to be just due to noise or control error and is thus excluded from model inputs) (Cawley and Talbot, 2010; Souza et al., 2016; Heinze et al., 2018)
Information	Process deviations or faults	Enlarge training data pool	The training data pool can be enlarged by the inclusion of datasets of various fault scenarios and the whole design space instead of only the operating space. Cases that are not covered in the training data pool will lead to unreliable extrapolation of the prediction model
		Process fault detection	Methods of process fault detection can be classified as based on quantitative models, qualitative models and search strategies, and on process history (Venkatasubramanian et al., 2003a; Venkatasubramanian et al., 2003b; Venkatasubramanian et al., 2003c)
	Sensor faults	Sensor fault detection	Sensor faults can be detected via various approaches (Das et al., 2012): symptom signal estimation, where the residual between the original and calculated (predicted) sensor reading indicates a sensor fault (Isermann, 2006; Isermann, 2011); multivariate statistical process control (MSPC), where faults are detected by the contribution of each input variable to underlying statistics of an empirical process model (e.g., PCA or PLS variants); and pattern recognition, where supervised or unsupervised learning algorithms are used to differentiate between faulty and non-faulty sensor data
		Fault tolerance	Fault tolerant soft sensors compensate for faults of inputs to the prediction model by a reconstruction of those inputs (Isermann, 2006; Isermann, 2011). Ensemble-based methods can potentially be used to discard or underweight sub-models with faulty model inputs
	Overfitting	Identification of overfitting	Overfitting can be determined during model evaluation via internal cross-validation (e.g., leave-one-out, <i>k</i> -fold, stratified, or time-series cross validation) and external (holdout) validation (Hawkins, 2004)
		Controlling model complexity	Model complexity can be controlled and thus overfitting can be reduced by a sound variable selection (see above)
	Deterioration of model performance	Model maintenance	In cases where the performance of the prediction model deteriorates due to unseen events (not yet included in the training data pool, e.g., changes in the production strain or seasonal changes in media components), the training data pool and sometimes also the model structure need to be updated (Wise and Roginski, 2015). In all other cases (similar events already included in the training data pool), adaptive modeling approaches such as recursive adaption and ensemble-based methods (Kadlec et al., 2011) as well as JIT learning (Kano and Fujiwara, 2012; Saptoro, 2014) can be used to maintain the prediction model
Knowledge	Implementation of expert knowledge	Digitalization of expert knowledge	Expert knowledge can be digitalized via fuzzy-logic-based approaches in the form of a rule base (Birle et al., 2013) or via first-principle models (Ohadi et al., 2015; Tahir et al., 2019)
		Hybrid modeling	Data-driven modeling can be combined with knowledge-based approaches to make use of available expert knowledge (Stosch et al., 2014; Solle et al., 2017). Hybrid modeling often results in a combination of the advantages and compensation of disadvantages of the two approaches

a compact selection of the most important process variables and analyzers, respectively, is given in this article. Typical online process data are composed of at least the following readings: flow rates, (differential) pressure (Krippel et al., 2021), temperature, pH, stirrer speed, pO₂, off-gas CO₂/O₂, and conductivity. Often, this standard instrumentation is supplemented by advanced measurement principles, such as turbidity (transmission, transflexion, reflection), impedance, pCO₂, high performance liquid chromatography (Dumarey et al., 2019), flow cytometry, *in-situ* microscopy, ultrasound, biosensors, proton-transfer-reaction mass spectrometry (Berbegal et al., 2020), and, last but not least, various spectroscopic techniques, such as ultraviolet–visible, near- or mid-infrared (Capito et al., 2015; Sauer et al., 2019; Walch et al., 2019; Wasalathanthri et al., 2020a; Cabaneros Lopez et al., 2021), 2D fluorescence (Melcher et al., 2015; Bayer et al., 2020), Raman (Matthews et al., 2016; Voss et al., 2017), and nuclear magnetic resonance (Kern et al., 2019).

As mentioned, the choice of analyzers used for monitoring and control depends on the used production organism. In mammalian bioprocesses (e.g., Chinese hamster ovary cells), for example, the cell concentration is in most cases significantly lower than in microbial bioprocesses (e.g., *Pichia pastoris*, *Saccharomyces cerevisiae*, *Escherichia coli*). Further, metabolite concentrations, which are particularly relevant in mammalian bioprocesses such as ammonium and lactate (Matthews et al., 2016), are relatively low. Due to higher growth rates, the cultivation time is typically shorter for microbial than for mammalian bioprocesses. For the development of soft sensors, special challenges may therefore arise for the respective expression system: First, the accuracy of the reference and online measurements limits the accuracy of the resulting soft sensors, which can take effect when analyte concentrations are low. Second, faster processes require higher measurement frequency according to the Nyquist–Shannon sampling theorem (microbial: ca. 20–120 h⁻¹ (Voss et al., 2017; Cabaneros Lopez et al., 2021); mammalian: ca. 0.5–12 h⁻¹ (Ohadi et al., 2015; Matthews et al., 2016)). This must be considered when specifying the prediction frequency of the soft sensor. Especially with the complex preprocessing necessary for spectroscopic data (see next section), the computational power can limit the prediction frequency of the soft sensor (Afseth et al., 2006).

Following this description of possible input data to a soft sensor, the subsequent section shows step by step how to develop a soft sensor. Afterwards, the state of the art in online implementation of soft sensors is shown, i.e., how the soft sensor is concretely used for online prediction.

Workflow of Soft Sensor Development

The development of soft sensors has been reviewed by several authors. Systematic approaches to soft sensor development have been presented by Fortuna et al. (2007), Kadlec et al. (2009), and Souza et al. (2016) for the process industry and by Haimi et al. (2013) for wastewater treatment plants. They all show a similar workflow. However, the focus of these review articles is on data-driven modeling approaches, and knowledge-based modeling approaches are for the most part neglected. Khatibisepehr

et al. (2013) present a systematic workflow for soft sensor development based on Bayesian methods, which inherently combine knowledge-based and data-driven modeling.

The basic workflow used as a framework in this review article generally assumes a hybrid use of knowledge-based and data-driven approaches (Figure 1). The core of soft sensor development is setting up and evaluating the prediction model. Besides these mandatory steps, the workflow is nonrigid: It depends on the individual case (degree of process knowledge, noisiness of inputs, need for model maintenance, etc.) whether all steps are conducted to the full extent.

The first step in soft sensor development is to evaluate the available raw data in terms of outliers and patterns in the datasets. Outlier analysis is important to identify samples or measurements that distinctly stand out from the rest of the data. An initial correlation analysis between model input and output can provide a matrix of correlation coefficients (e.g., Pearson's), which helps to assess relationships among the data. When interpreting the results of correlation analysis, however, one must keep in mind that correlation is not equivalent to causality. The correlation analysis can, in combination with available process knowledge, already be employed to preselect information-bearing model inputs (Melcher et al., 2015; Bidar et al., 2018). These analyses provide the basis for the selection of suitable data preprocessing and modeling methods.

The purpose of data preprocessing is to transform the raw input data into a form that minimizes the effect of noise and outliers while preserving the information content. Methods of data preprocessing include formatting, centering, scaling (e.g., to variance), and—specifically for spectroscopic data—baseline correction and peak alignment (Afseth et al., 2006; Matthews et al., 2016; Voss et al., 2017). Signal processing by smoothing and filtering (e.g., Hampel filter (Pearson et al., 2016)) can help to reduce noise and eliminate outliers. However, it is important to note that all preprocessing measures applied during model establishment must also be executable online.

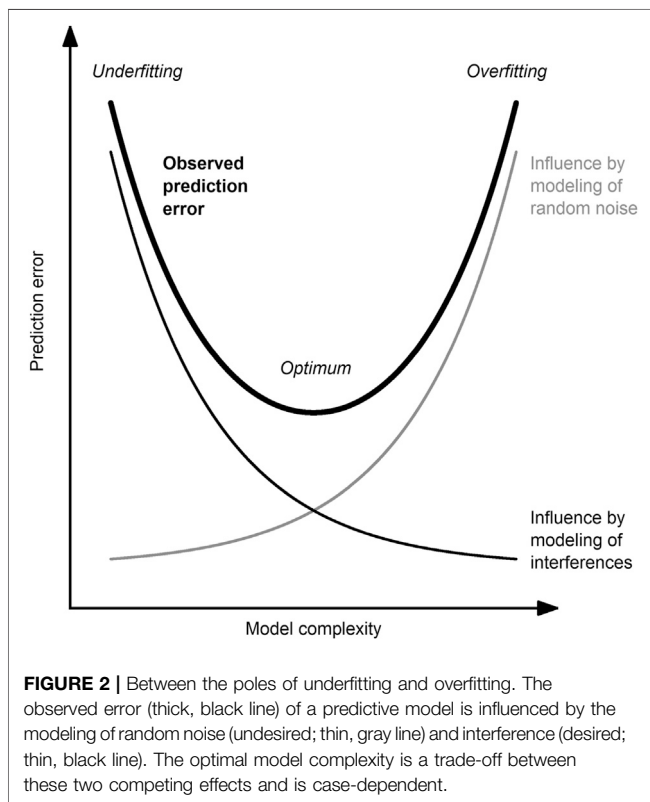
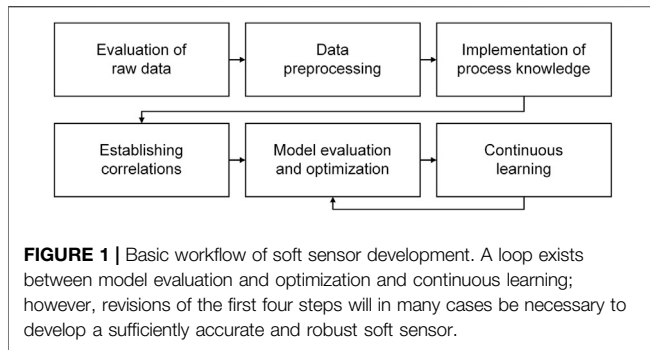
Process knowledge can be implemented into the soft sensor model. Knowledge-based model parts such as first-principle models (Ohadi et al., 2015; Steinwandter et al., 2017; Pappenreiter et al., 2019; Tahir et al., 2019; Krippel et al., 2021) can be employed to develop a more accurate and robust model. Process knowledge in the form of linguistic expressions can be digitalized using approaches based on fuzzy logic, as reviewed by Birle et al. (2013).

After these preceding steps, the actual correlation—the core of the soft sensor algorithm—is established. This correlation model maps the process data X (input) to the target quantity y (output) using model coefficients b . In its simplest, linear form, this model can be formulated as:

$$y = bX. \quad (1)$$

If more than one target quantity is predicted with the same model, the vector y in Eq. (1) is replaced by the matrix Y .

Taking into account the application fields of soft sensors described above, y can be a physical quantity that can be measured only offline (online prediction), a higher level, non-physical quantity (process monitoring and process fault



detection), or the reading of a hardware sensor (sensor fault detection). Various modeling techniques have so far been applied for soft sensor development, including variants of multiple linear regression (MLR; Jenzsch et al., 2006), partial least squares regression (PLSR; Sokolov et al., 2015; Voss et al., 2017; Zheng and Song, 2018; Walch et al., 2019; Cabaneros Lopez et al., 2021), principal component regression (PCR; Zhu et al., 2018), artificial neural networks (ANN; Paquet-Durand et al., 2017; Zhang et al., 2020), and support vector regression (SVR; Voss et al., 2017; Meng et al., 2019). The choice of the right modeling method depends on the degree of (multi)collinearity, nonlinearity, and the availability of process knowledge.

The model is typically trained, i.e., \mathbf{b} is determined, using historical data \mathbf{X}_{hist} and \mathbf{y}_{hist} (exception: just-in-time (JIT)

learning), so that $\mathbf{y}_{hist} = \mathbf{bX}_{hist}$. Subsequently, the resulting model needs to be evaluated in terms of goodness-of-fit, predictivity, and robustness (OECD, 2014). Here, the received model is used to predict the target quantity $\hat{\mathbf{y}}_{hist}$, so that $\hat{\mathbf{y}}_{hist} = \mathbf{bX}_{hist}$. After training and model evaluation, \mathbf{b} can be used together with online process data \mathbf{X}_{on} to predict the target quantity $\hat{\mathbf{y}}_{on}$, so that $\hat{\mathbf{y}}_{on} = \mathbf{bX}_{on}$.

For the robustness of the developed model, it is crucial that the model has neither too many nor too few model inputs nor too high nor too low model complexity, respectively (Figure 2). Methods to determine the optimal model complexity have been reviewed by several authors (Cawley and Talbot, 2010; Souza et al., 2016; Heinze et al., 2018).

Even with a robust and sufficiently accurate soft sensor, model quality or prediction performance, respectively, usually deteriorates if the process characteristics change (Kano and Fujiwara, 2012). Therefore, the maintenance or recalibration of soft sensors—just as for hardware sensors—is necessary in practice to preserve the quality of their prediction performance. In this context, model maintenance refers to the (automatic) adaptation of models in the event of changing system conditions. For the prediction models of a soft sensor, this means that the model parameters and, if necessary, the entire model structure (e.g., number and type of input variables) must be adapted over time.

Which programming environment or software solution is used to develop soft sensors in practice? Soft sensor development in the academic environment typically takes place in a programming language of choice such as Matlab (The MathWorks Inc.), Python, or R. The corresponding programming environments provide steadily growing libraries of functions or toolboxes for signal processing, data preprocessing, and model calibration and validation. Especially in the industrial environment, software specially developed for chemometrics is often used for soft sensor development (e.g., SIMCA by Sartorius AG; Unscrambler by Aspen Technology Inc.). Here, the full flexibility of development via program code is exchanged for a relatively straightforward and guided development process. Also, many vendors of online analyzers offer software modules for soft sensor development. In particular, vendors of spectroscopic sensors should be mentioned here (e.g., OPUS suite by Bruker Corp., iC suite by Mettler Toledo Inc., GRAMS suite by Thermo Fisher Scientific Inc.), but also vendors of other multivariate sensors (e.g., BlueVis by BlueSens gas sensors GmbH) offer corresponding software modules. Some software tools (chemometric and analyzer software) also offer the option to embed scripts generated via the above-mentioned programming languages into the soft sensor algorithm. This allows adding customized functions for signal processing and data preprocessing as well as developing prediction models that might not be included in the commercial software tool. Finally, soft sensors can also be developed on cloud-based platforms (e.g., MindSphere by Siemens AG, Predix by General Electric Co.) to have access to a wide variety of data processing and modeling tools and to be able to share the developed soft sensors across plant or company boundaries (Chen et al., 2020; Kabugo et al., 2020).

Online Implementation of Soft Sensors

How is a soft sensor used in practice for online prediction? In theory, the soft sensor is merely a combination of input data and a prediction model (see definition above). In practice, however, several additional aspects must be considered if a soft sensor is to be used for online prediction, i.e., implemented online.

First of all, online implementation of soft sensors requires at least communication between field (sensors and actuators) and control level (programmable logic controller and/or process control system) and in most cases also supervisory level (supervisory control and data acquisition, SCADA, and/or other data management system). The data used as inputs to the soft sensor can originate from various sources (Steinwandter et al., 2019). Therefore, a standardized communication between these sources and the software instance in which the soft sensor is implemented is essential. While a variety of standard communication protocols exist for communication between field and control level (4–20 mA, Modbus, Profibus, etc.), it is communication via OPC UA (open platform communications unified architecture) that seems to become the predominant standard for communication in the control and supervisory level (Chen et al., 2020; Biermann et al., 2021). Recent efforts even aim at field-level communication using OPA UA (Veichtlbauer et al., 2017). OPC UA, unlike its predecessors of OPC classic (data access, alarms and events, historical data access), allows hardware- and platform-independent communication.

Once the communication and thus the data flow between field, control, and supervisory level has been established, the question arises on which level of the automation pyramid the soft sensor is implemented. Technically, it is possible to implement soft sensors directly in the control level. However, the implementation of scripts directly in the control system is intended for end users only in exceptional cases and the proprietary language must be used (Nair et al., 2020). Systems above the control level, on the other hand, commonly offer the possibility to implement soft sensors directly or indirectly. In the direct variant, soft sensors are implemented in the SCADA (e.g., MFCS by Sartorius AG, Eve by Infors AG, BioXpert by Applikon Biotechnology BV) or other data management system (e.g., SIMATIC SIPAT by Siemens AG, synTQ by Optimal Industrial Technologies Ltd., xPAT by ABB Ltd., Lucullus PIMS by Securecell AG, LabVIEW by National Instruments Corp.). Here, preprocessing steps and model calculations can be implemented directly to a certain extent. More importantly, these software tools often offer the possibility to communicate with external chemometric or analyzer software (Matthews et al., 2016; Voss et al., 2017; Dumarey et al., 2019) or to integrate customized scripts that are executed online (Besenhard et al., 2018). In this indirect variant, soft sensors are implemented in real-time capable chemometric (e.g., SIMCA-online by Sartorius AG (Voss et al., 2017), Process Pulse by Aspen Technology Inc.) or analyzer software (e.g., CMET by Bruker Corp. (Wasalathanthri et al., 2020a), iC Quant by Mettler Toledo Inc. (Wu et al., 2015)) that communicates with the SCADA or data management system. Here, communication often already takes place via OPC UA (Kern et al., 2019). In this indirect implementation, the chemometric or analyzer software

preferentially communicates information (e.g., the predicted value) rather than data back to the SCADA or data management system (Luttmann et al., 2012).

The PAT software products mentioned in this section are only a selection and should not be seen as a recommendation. For a more comprehensive overview of PAT software, the reader is referred to Chew and Sharratt (2010). The authors also list whether the respective software is compliant with regulatory requirements for electronic records and signatures according to 21 CFR Part 11 (FDA, 2003).

When soft sensors are implemented in an industrial environment, they must first undergo an intensive functional and risk assessment (qualification). A step-by-step guidance for structured development and implementation has been proposed by Randek and Mandenius (2018). This guidance considers the regulatory validation requirements for software including recommended protocols for installation, operational, and performance qualification. The validation of software, especially in the pharmaceutical environment, commonly follows guidelines such as GAMP 5 (ISPE, 2008), 21 CFR Part 11 (FDA, 2003), or EU GMP Annex 11 (EC, 2010).

CHALLENGES IN SOFT SENSOR DEVELOPMENT FOR BIOPROCESSES

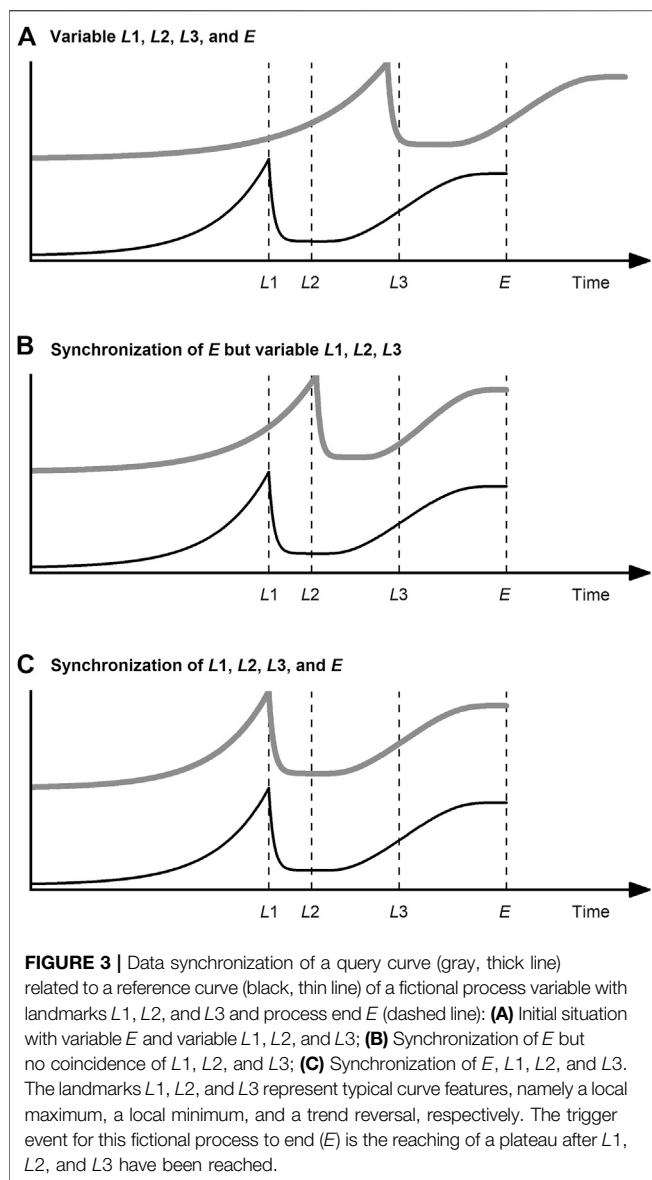
This chapter concerns the challenges in soft sensor development for bioprocesses from a data scientist's perspective, namely, variable process lengths, multiple process phases, and sensor faults. For each of these three challenges, the problem statement is initially outlined. Subsequently, the solution approaches are critically discussed, linking them to the other two challenges, wherever possible. Each solution approach is summarized at the end particularly regarding its practicality and applicability to bioprocesses.

Variable Process Lengths Problem Statement

From a process engineering perspective, the end of a bioprocess is defined either by the expiration of a certain process time or by the occurrence of a certain process event. Such termination events can be, for example, the reaching of a target value for the biomass or product concentration or a specific pattern in the process data (e.g., a CO₂ peak indicating the consumption of a carbon source).

In the case of an event-driven process end, the process length can vary from batch to batch due to multiple sources of variance. Besides the typical variance of biological reactions, variance can be introduced by raw materials (e.g., media or feed), by preceding processing units (e.g., preculture), or by deviations in the current process itself.

The variable length of process runs can lead to the following problems. First of all, it can distort the equal weighting of the individual datasets during model development and evaluation if the reference data y_{hist} are generated at a constant frequency: More reference data points for longer processes lead to an overweighting of longer processes compared to shorter ones. Secondly, in case a dynamic soft sensor model incorporates



time as an input variable to compensate for time-variant behavior, variable process lengths lead to another problem: Model performance may deteriorate to the extent that the rate of process progress deviates (i.e., the process is too fast or too slow) from the historical data that were used to train the model. Thirdly, in multiphase processes, the sources of process variance described above can lead to deviations in the time of occurrence of process events (Ündey et al., 2002). If this issue is not accounted for, the adaptation of a soft sensor to process phases could be impeded.

Various methods of data synchronization have been developed to address the challenge of variable process lengths. Data synchronization has two goals, as illustrated in **Figure 3** for a fictional process variable: on the one hand, to bring all process datasets to the same lengths (**Figure 3B**); on the other hand, to ensure that the relevant process events (landmarks) coincide (**Figure 3C**).

The three techniques used most commonly for data synchronization are discussed in the following: indicator variable, curve registration, and dynamic time warping (DTW). The goal of all these methods is to find a warping function h that replaces the time t on the abscissa and thus to obtain synchronized process data X_{sync} (Ramsay and Silverman, 2005):

$$X_{sync} = X[h(t)]. \quad (2)$$

As part of soft sensors that are adaptable to variable process lengths, the synchronization algorithm needs to be executable both offline during model development (for X_{hist} and y_{hist}) as well as during the online application (for X_{on}).

As with all, the choice of the data synchronization method is highly dependent on the process being monitored (Rato et al., 2016; Rato et al., 2018). It should also be noted that, regardless of the method used for data synchronization, all subsequent levels of the monitoring algorithm (soft sensor prediction, fault detection, etc.) depend for better or worse on the robustness and accuracy of the synchronization method used.

Indicator Variable Techniques

In this method, the time scale is replaced by an alternative scale, the indicator variable. The indicator variable can be either a real (physical) process variable or an estimated process progress, often referred to as maturity index or percent completion. Process variables that are used as termination criteria for the process or as trigger variables for an automation system are particularly suitable as indicator variables (Ündey et al., 2003; García-Muñoz et al., 2011). Examples of process variables suitable as indicator variable are decrease of substrate concentration (Ündey et al., 2002), cumulative feed volume (Ündey et al., 2003), bioreactor volume, and biomass concentration (Rato et al., 2016). Regardless whether a real process variable or a maturity index is used, the indicator variable should ideally progress strictly monotonically, continuously, and smoothly and have the same start and end value (e.g., 0 and 100 % maturity) for all process runs (Nomikos and MacGregor, 1995; Ündey et al., 2002; Ündey et al., 2003).

When developing a prediction model for the maturity index, the percentage of process progress is calculated for the training data, e.g., by a simple linear transformation. The model requires monotonically progressing variables that correlate with process progress. Examples of the use of a maturity index for data synchronization in bioprocesses can be found in Krause et al. (2015) and Brunner et al. (2019). Both studies demonstrate how a maturity index based on a PLS model can be used to determine process progress online and thus enable adaption to the time-variant behavior of biological batch processes. Only through information about the process maturity was it possible to detect sensor faults in the respective bioprocesses.

Ündey et al. (2003) addressed the challenge of variable process length for a multiphase process, namely a simulated fed-batch penicillin fermentation with two phases (batch and fed-batch phase). They proposed using separate indicator variables for each process phase to compensate for the variable lengths of the

phases. As a result, the authors were able to construct tighter control limits for an MSPC model, which in turn enabled faster fault detection. A similar approach was presented by García-Muñoz et al. (2003) for an industrial drying process with three process stages. It was shown that incorporating warping information—i.e., “the information that comes out of an alignment” (García-Muñoz et al., 2003)—resulting from the stage-by-stage alignment can improve a quality prediction model.

In summary, indicator variables are suited for data synchronization of bioprocess data under the condition that there is a minimum understanding of the temporal behavior of the process variables. If this knowledge is available and especially if the process variable used as the indicator variable is used as termination criterion for the process, there is no more robust and simple method than this. Problems with the prediction of the maturity index can occur if the input variables of the model change fast in certain process phases and slowly in others. This is not uncommon, especially in USP (lag vs. exponential phase). Even if this is considered by using a non-linear model, the resolution of the input variables restricts the relative accuracy in “slow” process phases. This resolution is determined by the sensors and actuators used. In cases where it is difficult or impossible to find or calculate an indicator variable that comes close to the above-mentioned requirements (strictly monotonically progressing, etc.), curve registration techniques or DTW should be considered. Finally, it must be stressed that indicator variable techniques are per se designed to be independent of any landmarks. These structural features, which are especially helpful for multiphase processes, are ignored during data synchronization and thus cannot be exploited. Data synchronization with indicator variable techniques is therefore limited to the scenario shown in **Figure 3B**.

Curve Registration Techniques

Within functional data analysis, curve registration is referred to as the process of aligning one function curve to another (Ramsay and Silverman, 2005). In this sense, the term curve registration does not differ from the term data synchronization, only that it refers specifically to functional data. The process data are seen as observations of an underlying continuous function (Ündey et al., 2002). The curves are aligned with respect to their structural features, referred to as landmarks. These landmarks can be certain levels, extrema (minima, maxima), or trend reversals (see *L1*, *L2*, *L3*, and *E* in **Figure 3**). The relevant landmarks are identified using process knowledge and/or numerical computations, such as first and second derivative, respectively, and zero crossing (Ündey et al., 2002; Ramsay and Silverman, 2005). After matching the landmarks between reference and query, the sections between the landmarks are warped, which in the simplest case means that they are resampled linearly.

Williams et al. (2001) and Ündey et al. (2002) used curve registration to align the process data of a simulated fed-batch penicillin fermentation. For the alignment of multivariate data, the authors suggest first aligning all process data with respect to the landmarks of the most important variable (determined, e.g., via process knowledge). In the second step, a principal

component analysis (PCA) is carried out and the process variables are aligned with respect to the landmarks of the first principal component. The second step is repeated until the landmarks converge. In these studies, it was shown that curve registration provides relatively smooth variable trajectories after the alignment compared to DTW; in this way, fewer false alarms occurred with MSPC-based fault detection. In one other of the few examples from the bioprocess field, Andersen and Runger (2012) used landmarks of a pharmaceutical batch fermentation process for data synchronization. The significant landmarks were automatically identified as the zero crossings of a continuous Gaussian wavelet transformation (Bigot, 2006). Afterwards, the resulting curve segments were warped linearly and piecewise for each segment.

In summary, curve registration techniques allow not only the alignment of variable lengths—as with an indicator variable—but also the alignment of curve features. Scenario C in **Figure 3** can therefore be achieved. Since the features of many process variables occur simultaneously at phase transitions, curve registration techniques are particularly suitable for multiphase processes (Ündey and Çınar, 2002). However, applications of curve registration for bioprocesses are rare. The existence of this niche in the field of bioprocesses can at most be explained by the circumstance that the indicator variable technique is more intuitive and comparatively easy to implement and DTW can be used with less fine tuning.

Dynamic Time Warping

DTW, initially developed for speech recognition (Sakoe and Chiba, 1978), was proposed for the synchronization of process data by Kassidas et al. (1998). Since then, it has become one of the most widely used methods for this purpose. Reasons for this are that not only variable process lengths but also landmarks can be aligned using DTW. Scenario C in **Figure 3** can therefore be achieved, just as with curve registration. DTW expands, contracts, or translates the time axis of the datasets in such a way that the shape of the variable trajectory is largely preserved, landmarks coincide in time and all datasets have a uniform number of measuring points. The basic sequence of DTW algorithms is as follows: First, the distance matrix (e.g., Euclidean) between the instants of the reference and the query time series is calculated. Then the warping path is searched for that minimizes the sum of distances and at the same time considers several boundary conditions (local, global, endpoint). Using this warping path, the query time series is aligned to the reference time series by expanding, contracting, and translating.

Since its introduction for data synchronization, the original DTW algorithm has been varied in several ways to address issues such as singularities. Singularity in this context refers to the mapping of a single point of the reference time series to multiple points of the query time series or vice versa. Derivative DTW (DDTW) uses local derivatives of the time series instead of raw data and was proposed for overcoming singularities (Keogh and Pazzani, 2001). DDTW compared to DTW tends to align more based on shape rather than magnitude (Spooner et al., 2018). Since numerical derivation often leads to an amplification of noise, a Savitzky-Golay filtering step can be implemented in the

DDTW algorithm to make the alignment more robust (Zhang et al., 2013).

For process data that show sections with many successive landmarks (feature-rich) and then again sections with few landmarks (feature-poor), a fixed warping resolution is often not sufficient. Therefore, a dynamic warping resolution was proposed by Gins et al. (2012). This is achieved by a combination of correlation optimized warping (COW; Nielsen et al., 1998; Fransson and Folestad, 2006) for feature-poor and DDTW for feature-rich sections (hybrid DDTW).

The difficulty with the online application of DTW is that a *partially complete* dataset (query) needs to be aligned with a *complete* dataset (reference). This issue was first addressed by Kassidas et al. (1998) and later further elaborated by González-Martínez et al. (2011). In both studies, the endpoint constraint, i.e., that the endpoint of the query must equal the endpoint of the reference, was omitted. This means, however, that the alignment has to be calculated at each sampling point and each time the recent history of the trajectory has to be considered. For this reason, a computationally efficient way of finding the optimal warping path within a moving window was proposed by González-Martínez et al. (2011), referred to as relaxed-greedy time warping (RGTW). Another online application of DTW was presented by Srinivasan and Qian (2007). They used dynamic locus analysis (Srinivasan and Qian, 2006) to identify the best matching signal segment from a reference library by making use of singular points (landmarks) and thus to determine the state of the process. For the actual online warping, a greedy version of the DTW algorithm, referred to as extrapolative time warping (XTW; Srinivasan and Qian, 2005), was used.

González-Martínez et al. (2014) extended the concept of DTW (offline application) and RGTW (online application) to the problem of multiple asynchronisms for a simulated *S. cerevisiae* fermentation. Multiple asynchronism in this context refers to a combination of at least two of the following asynchronism scenarios: variable process length; no coincidence or overlapping of key process events; initial delay or premature termination of a process. The authors proposed a two-step approach in which the asynchronism pattern is firstly detected based on the warping information and secondly batch synchronization is performed based on the detected pattern.

In the standard DTW procedure (univariate DTW), a single representative process variable is used as a reference to align all other process variables. In certain cases, however, univariate DTW can lead to misleading results; this includes, for example, a delayed measurement in a bypass (on-line) or the bioreactor periphery (at-line) compared to the remaining measurements in the bioreactor (in-line). In these cases, multivariate DTW (MDTW) should be considered. Two fundamental variants are distinguished in MDTW (Shokoohi-Yekta et al., 2015): Either DTW is performed separately for each of the process variables j , resulting in j potentially different alignments (“independent” MDTW); or the warping path is determined via a multidimensional p -norm as cost function, whereby multiple process variables are included in the calculation of the distance (“dependent” MDTW). For a

review of MDTW, the reader is referred to Moser and Schramm (2019).

In summary, DTW and its variants have—at least for simulated data—proven to be well suited for synchronizing bioprocess data, both offline and online. No process knowledge is necessary to develop this preprocessing method. When dealing with multiple process phases, DTW can be used in two different ways: first, it can be used to detect process phases (Gollmer and Posten, 1996); second, it can be used to align data within a process phase (Doan and Srinivasan, 2008; Spooner et al., 2018). The use of DTW for these purposes is further described in the following section. Finally, it should be noted that the warping information can be used for the classification of deviations from normal operating conditions, such as sensor faults (González-Martínez et al., 2013). However, in order to identify the deviating sensor, each fault scenario of interest must explicitly be included in the training data pool.

Multiple Process Phases

Problem Statement

From a monitoring perspective, industrial processes can take place either in multiple processing units (multistage) or in a single one. A process with a single processing unit (e.g., USP in a bioreactor) can have multiple operational regimes, such as a batch and fed-batch phase, and is referred to as multiphase process. Multiphase processes are often treated analogously to multistage processes (Yao and Gao, 2009), i.e., different process phases are treated as if they took place in separate processing units.

The necessity of considering multiple process phases when developing a soft sensor is obvious: The relationships within the input data X (multicollinearity) and between the input data and the target variable y (correlation) can vary substantially in the individual phases. The challenges discussed in this section refer to changes in the relationships from X to y that are related to the process strategy. These include, for example, an induced change in media composition due to feeding, the start or end of a starvation phase, long-term changes between oxygen-limited and non-limited process conditions, or changes in the temperature setpoint. Changes in the relationships of X to y that are associated with time-variant and nonlinear behavior are not within the scope of this review article, although the respective adaption mechanisms partly overlap. These adaption mechanisms have been excellently reviewed by Kadlec et al. (2011).

Only with much greater development effort or available process knowledge will a global process model attain the same accuracy and robustness as several submodels for each process phase. Graphically expressed, the required model complexity (cf. **Figure 2**) of a global model is allocated to several less complex local models. This in turn can make it easier to optimize the model (Jin et al., 2015), for example, in terms of avoidance of overfitting.

The main difference between datasets of multistage and multiphase processes is that in multiphase processes the individual phase segments must first be identified and often cannot be precisely separated. The actual modeling step is therefore often preceded by a phase detection and division

step. The detection and division is based either on trajectories of phase-sensitive process variables or on the changing correlation structure among the process variables (Luo et al., 2016).

Trajectory-Based Phase Detection and Division

The sequence of the most biotechnological processes is not given by nature, but by process experts. Therefore, if knowledge about the process sequence is available, it is reasonable to use it for phase detection and division. The definition of landmarks by process experts leads to a solution that is both robust and comprehensible. An example of this can be found in Spooner et al. (2018), who, in contrast to their previous study (Spooner et al., 2017), first divided a bacterial fermentation process into two phases and then aligned the process data of these phases separately via DTW and DDTW, respectively. The pH and pH correction agent (flow and cumulative amount) signals were used to distinguish the phases. Brunner et al. (2020) used the off-gas CO₂ signal to detect the consumption of the carbon source and thus the end of the batch phase in a *P. pastoris* fed-batch bioprocess. To make the detection of this landmark (CO₂ peak) more robust, a threshold for the cumulative amount of pH correction agent was additionally implemented in the phase detection algorithm.

The role of DTW for data synchronization has already been described in the previous section. As a means of detecting process phases, it was first proposed by Gollmer and Posten (1996) for time-varying fed-batch bioprocesses (*E. coli* and *S. cerevisiae*). With the use of historical time trajectories of CO₂ and O₂ together with available process knowledge, six different process phases were classified. This reference (prototype) was used in the online application by the DTW algorithm to assign unknown process data to this pattern and thus detect the previously defined process phases. Doan and Srinivasan (2008) proposed a variant of DTW augmented by singular points (landmarks) for the combined detection and synchronization of process phases. They used substrate feed and pH as key variables for the detection of process phases of a simulated fed-batch penicillin fermentation. Phase changes were considered equivalent to the occurrence of singular points and were identified using the methods described in the previous section (Srinivasan and Qian, 2005; Srinivasan and Qian, 2007). DTW (offline) and XTW (online), respectively, were then used for data synchronization within the phase segments.

Especially in USP, phase transitions must be considered, as biological systems involve living cells, which do not react instantaneously to environmental changes. Luo et al. (2016) proposed a framework for adapting process models to a sequence of multiple process phases while explicitly considering phase transitions in a simulated fed-batch penicillin fermentation. They used fuzzy *c*-means (FCM) clustering for phase detection and division to account for the gradual transition from one steady phase to another. The FCM clustering algorithm was constrained by the temporal sequence of the dataset. Phase-based multiway PLS models were used for prediction in the steady phases, and JIT-PLS models were used during the transition phases.

In summary, trajectory-based algorithms are suitable for phase detection and division in cases where a minimum of

process knowledge is available. This knowledge is necessary to select the phase-sensitive process variables. Provided that suitable phase-sensitive process variables can be identified, this approach is more comprehensible than the correlation-based approach. This is especially due to the fact that the identified phases usually correspond to operational phases (Yao and Gao, 2009). Finally, it must be emphasized that DTW is suitable not only for data synchronization in the case of variable process lengths, as described above, but also for the detection of multiple process phases.

Correlation-Based Phase Detection and Division

A difficulty with the methods mentioned so far is to find variables that are measurable and sensitive to the individual phases (Luo et al., 2016) and whose trajectories are reproducible and as noise-free as possible. In the following, methods are presented that accomplish phase detection and division without the need for process knowledge. These methods are based on changes in the correlation structure among process variables.

Camacho et al. (2008) proposed an algorithm based on latent variable models (PCA or PLS) for the detection and division of process phases for a *S. cerevisiae* and a wastewater treatment process. The whole process dataset is iteratively divided into incrementally smaller phases. At each iteration step, the separation point that leads to the maximum improvement of the explained variance of the PCA or PLS submodel, respectively, compared to the undivided dataset is identified (Camacho and Picó, 2006).

Another method to make use of changes in the correlation structure is to first determine the loading matrices of PCA or PLS submodels following a moving window approach and then to find groups in which the underlying variable correlation remains similar (Lu et al., 2004; Lu and Gao, 2005). These groups can, for example, be determined by *k*-means clustering (Lu et al., 2004).

However, if different operation modes with variable phase and process lengths are to be considered, the classical moving window approach leads to misleading results. The reason for this is that in the online application it is not clear whether the current moving window coincides with that of the reference (historical data). Therefore, the described method for phase detection and division (Lu et al., 2004) was extended by the ability to identify the current mode of operation (resulting in variable lengths) online. Zhang et al. (2018) generated a series of moving windows within a constrained searching range around the current sample. They then used the *k*-nearest neighbor rule to identify the most similar time slices. The time slices found in this way are then used as described above (loading matrices of submodels, *k*-means clustering) to enable online phase detection despite variable phase and process lengths.

Finally, Gaussian mixture models (GMM) have proven to be suitable for phase detection and division for a simulated fed-batch penicillin fermentation (Yu and Qin, 2009; Yu et al., 2013). Here, each phase is represented by a Gaussian component with distinct mean and covariance. The posterior probability is used to group the process data into separate process phases. This concept was later adopted for a real industrial bioprocess (Mei et al., 2017).

In summary, for correlation-based phase division, the reduced effort in implementing process knowledge is compensated with an increased effort in modeling. Depending on the modeling method, however, an entirely different division of the process phases may result and fine-tuning of the latent variable models is necessary (Luo et al., 2016). Because the correlation structure is of multivariate nature, the interpretability of the results of the phase division is limited in contrast to most trajectory-based methods.

Sensor Faults

Problem Statement

Sensor faults are defined as deviations of the observed sensor reading from the true value (Balaban et al., 2009; Sharma et al., 2010). They are distinguished according to the type of occurrence as abrupt (stepwise) or incipient (driftwise) faults (Isermann, 2006) and according to the shape of the deviation as bias, precision degradation, and complete failure.

During the training phase of soft sensor development, sensor faults can severely affect the resulting goodness-of-fit and predictivity. If sensor faults are present in the training data X_{hist} and y_{hist} , these deviations may be reflected in the model coefficient b and the prediction \hat{y}_{hist} . In this case, evaluation criteria for goodness-of-fit and predictivity (e.g., R^2 , root mean squared error) are affected. During the online application of the soft sensor, i.e., the prediction phase, sensor faults in X_{on} may directly affect the prediction of the target quantity \hat{y}_{on} .

The validation of sensor readings prior to their use for quality control, e.g., via soft sensors, is therefore of crucial importance, as outlined by Feital and Pinto (2015). A sensor reading is valid if there are no sensor faults or unconsidered influences on the measurement, which can occur due to cross-sensitivity to matrix compounds (matrix effects). Deviations between the observed sensor reading and the true value thus need to be detected, and a decision logic needs to classify the observed sensor reading as reliable (valid) or faulty (invalid). Valid sensor readings can be used for quality control by means of soft sensors, while invalid ones can lead to misleading results.

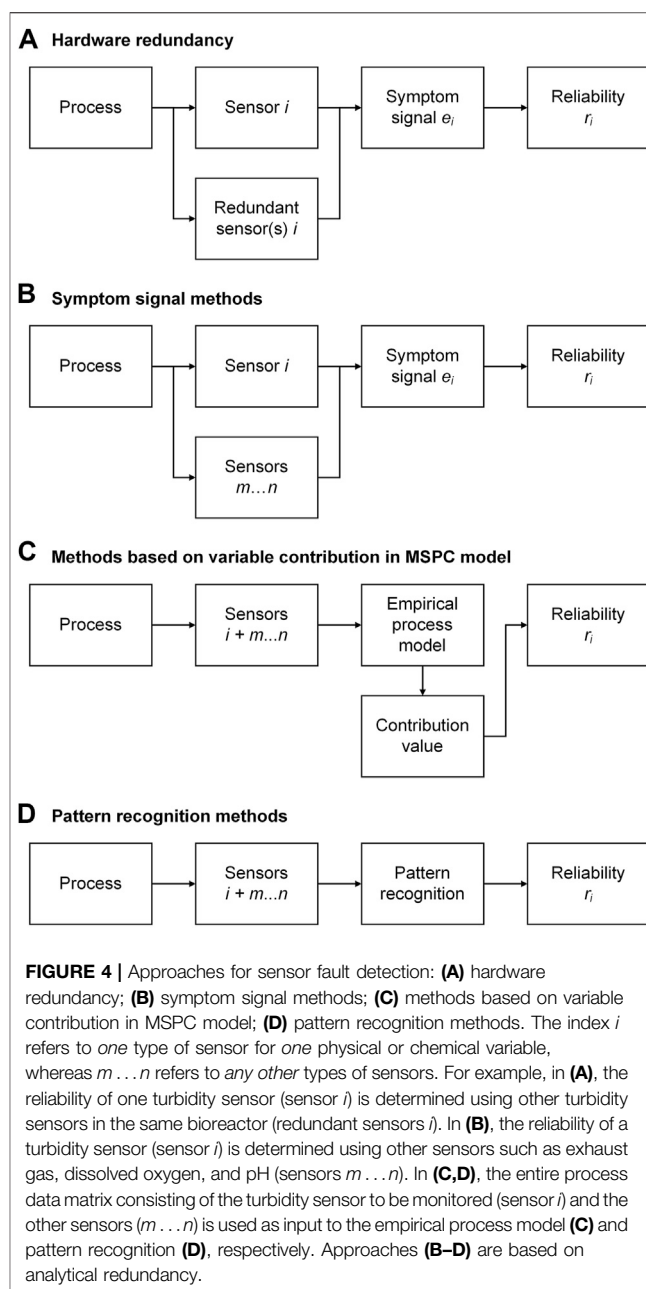
The fault tolerance of soft sensors, or, in other words, a reliable soft sensor prediction in the presence of sensor faults represents one of the remaining core challenges in the development of soft sensors. The reason for this is that the detection and subsequent compensation of sensor faults alone are difficult to realize, but they become even more complex when the conditions described above (variable process lengths and multiple process phases) occur simultaneously.

This section first discusses various methods of detecting sensor faults. Afterwards, the approaches for tolerance of soft sensors towards sensor faults are discussed.

Sensor Fault Detection

When a sensor i that is used to monitor a bioprocess gives faulty readings, its reliability r_i decreases. The aim of sensor fault detection is to detect these faulty readings and thus indirectly determine r_i . **Figure 4** shows four fundamental approaches to sensor fault detection.

Hardware redundancy uses multiple identical sensors to derive the occurrence and size of sensor faults in case of a



significant discrepancy among these sensors (as, for example, in airplanes). Voter structures can be implemented into the fault detection algorithm to allow a “democratic” decision on which of the individual sensor values is faulty. If, for example, two of three sensors give a similar reading and the third reading deviates significantly, the third sensor is considered to be faulty. For hardware redundancy, the spatial distribution of the sensors must be considered, and the costs of the sensors limit this approach (Stork and Kowalski, 1999). Hardware redundancy can also be used to determine the type of fault as bias, gain, precision degradation, complete failure, and noise (Kullaa, 2013).

The other three approaches are based on analytical redundancy and are described in the following.

Symptom Signal Methods

As mentioned in the introduction, soft sensors themselves can be used to assist in sensor fault detection. Here, the target quantity y of the soft sensor is the reading of the hardware sensor to be monitored. A deviation of the prediction \hat{y}_{on} from the original reading y_{on} beyond a defined threshold value indicates a sensor fault. The residual between \hat{y}_{on} and y_{on} is referred to as symptom signal e (Zarei and Shokri, 2014) and can in its simplest form be formulated as:

$$e = y_{on} - \hat{y}_{on}. \quad (3)$$

Several authors make use of state-space models for the generation of the symptom signal, as described in the following. Zarei and Shokri (2014) used a nonlinear unknown input observer to generate symptom signals and thereby to detect sensor faults in a simulated continuous stirred-tank reactor (CSTR) process. Alag et al. (2001) proposed a framework for sensor fault detection based on the symptom signal method exemplarily for a gas turbine power plant. They proposed a multi-step algorithm for determining \hat{y}_{on} followed by the generation and evaluation of the symptom signal. First, a redundant prediction for each sensor in the network is generated based on regression methods such as neural networks (redundancy creation). Subsequently, these predictions are fused with original sensor readings into a state-space model based on a Kalman filter approach (state prediction and fusion). The statistical properties of the symptom signal are in combination with probabilistic reasoning finally used to identify both abrupt and incipient sensor faults. Since a symptom signal is created for each sensor, the proposed methodology is capable of detecting multiple sensor faults simultaneously.

Autoassociative neural networks (AANN) were first introduced by Kramer (1991) for sensor fault detection and reconstruction in a simulated chemical batch process. They have proven to be effective in detecting sensor faults in a fermentation process (*Streptomyces virginiae*) with variable process length and multiple process phases (Huang et al., 2002). AANN are feed-forward neural networks consisting of an input, an output, and three hidden layers (mapping, bottleneck, and demapping layer). The outputs of the bottleneck layer are considered equivalent to the principal components of a nonlinear PCA (Kramer, 1991). The key concept of AANN is that the model is trained with fault-free process data X_{hist} both as input and output, so that $X_{hist} = Y_{hist}$. The resulting nonlinear model is used for determining online predictions of the process data, \hat{Y}_{on} , based on online measured process data, $X_{on} = Y_{on}$; then, analogous to Eq. (3), the residual is calculated for each variable and used for the detection of sensor faults. This concept was extended to a complex nonlinear system with time-delays, namely a multicomponent distillation column (Perla et al., 2004).

The symptom signal method was used by Brunner et al. (2019) to detect sensor faults in a *P. pastoris* batch process. Due to the time-variant behavior and variable lengths of the batch processes, an indicator variable (maturity index) is introduced to predict the

process progress online. For each process section, a set of prediction models for \hat{y}_{on} is generated. A regularization approach based on binary particle swarm optimization (PSO) is used to select the 25 best prediction models. The distribution of the predictions \hat{y}_{on} is compared to a moving window distribution of y_{on} using the Kullback–Leibler divergence (Kullback and Leibler, 1951). The divergence between \hat{y}_{on} and y_{on} indicates a sensor fault and is used to quantify the sensor reliability r_i .

Most studies use a fixed threshold for fault detection based on symptom signals. This can lead to false alarms when unforeseen events or noise occur in the sensor network data. In these cases, a time-varying as opposed to a fixed threshold can increase robustness and minimize the fault detection time, as shown by Armaou and Demetriou (2008) for simulated chemical processes. However, if process lengths vary, the threshold needs to adapt dynamically to process progress and not just to process time. For this reason, Brunner et al. (2019) proposed a dynamic threshold, which is calculated by means of the confidence width of \hat{y}_{on} , which in turn is dependent on the process progress.

In addition to the mere detection of a sensor fault, information about the type of fault may also be necessary for the potential subsequent compensation (fault tolerance). To determine the type of fault as either bias, complete failure, drifting, or precision degradation, Dunia et al. (1996) developed a concept in which the symptom signal is generated using a PCA prediction model.

In summary, symptom signal methods are well suited for the detection of sensor faults and they are relatively intuitive due to their similarity to hardware redundancy. The main bottleneck of this approach is the model for the prediction of \hat{y}_{on} , which is used for generating the symptom signal. For most bioprocesses, the model needs to consider time-variant behavior and variable process lengths. With the exception of AANNs, this model must be developed separately for each sensor to be monitored. The main advantage of the symptom signal method is that there is a direct reconstruction for the faulty sensor value available. Soft sensors or control systems, which depend on a reliable sensor input, can fall back on the reconstructed value and thus be designed fault tolerant.

Methods Based on Variable Contribution in Multivariate Statistical Process Control Model

Multivariate statistical process control (MSPC) and its corresponding empirical process models and control charts are another method to detect sensor faults. The original idea of MSPC is to map a wealth of process data X to one or a few higher-level, non-physical process quantities y or Y , respectively, such as latent variables (Kourti, 2005). Deviations between historical, X_{hist} , and online process data X_{on} are detected using control charts based on the complementary *SPE* (squared prediction error; sometimes denoted as Q) and Hotelling's T^2 statistics (Nomikos and MacGregor, 1995; Liu et al., 2017; Sánchez-Fernández et al., 2018). Once these test statistics indicate a significant deviation, the contribution of each input variable in X_{on} to the test statistic(s) is calculated. Sensors or variables, respectively, with a significantly high contribution to the test statistic(s) are associated with a sensor fault. A general analysis of the

variable contribution approach is given by Qin (2003). Various methods for decomposing the test statistics to contributions, such as complete, partial, or reconstruction-based decomposition, were analyzed by Alcalá and Qin (2011).

Sánchez-Fernández et al. (2018) combined the symptom signal and the contribution-based method for the detection of process and sensor faults. Residuals between predictions and observations for each variable in X_{on} are used as inputs to a PCA-based MSPC model. Residuals that are calculated with faultless training data are used for calculating the thresholds (control chart limits) for fault detection based on T^2 and SPE statistics. Multivariate and univariate exponentially weighted moving average control charts are used for the detection of process and sensor faults, respectively. Two simulated benchmark processes (Tennessee Eastman process and wastewater treatment plant) were used for validating the concept.

Another combination of the symptom signal and the contribution-based method was proposed by Yoo and Lee (2006) for sensor fault detection. Here, contribution plots assist in identifying faulty variables. In case the contribution plots indicate a fault, the original measurement is compared with a prediction based on a fuzzy PLS model of the corresponding variable. However, no algorithm was presented on how to derive the sensor reliability or the fault magnitude and type, respectively. The concept was evaluated on a real and a simulated wastewater treatment plant.

Reconstruction-based contributions (RBC) were proposed for sensor fault detection by Yue and Qin (2001). Here, T^2 and SPE are combined in a fault detection index φ . This combined index proved to have better detectability both for single and multiple sensor faults than if the contributions to T^2 and SPE are considered separately (Yue and Qin, 2001; Alcalá and Qin, 2009). This concept was adopted by Tôrres et al. (2018) for pharmaceutical tablet manufacturing. The RBC approach was extended by Mnassri and Ouladsine (2015) to handle multiple and more complex sensor faults.

A contribution-based approach to sensor fault detection and tolerance was developed by Krause et al. (2015) for the monitoring of a yeast fermentation process. This approach does not consider the contribution to the test statistics as described above, but the direct contribution to the model b and the prediction \hat{y}_{on} , respectively, for fault detection. They developed a PLS-based MSPC model using an indicator variable to compensate for variable process lengths. For each process section, a set of MSPC models is generated and PSO is used for finding the best models with respect to historical process data (with normal process behavior). Variable importance in the projection (VIP) scores (Chong and Jun, 2005) were used to evaluate the input variables for their contribution (information content) to the MSPC model. A reduction of the VIP score of a variable is assigned to a fault of the corresponding sensor.

A problem not to be underestimated in contribution-based fault detection is the smearing effect (Alcalá and Qin, 2011; van den Kerkhof et al., 2013). Smearing here refers to the “influence of faulty variables on the contributions of non-faulty variables” (van den Kerkhof et al., 2013). Faulty variables (i.e., soft sensor inputs) can thus be concealed, and non-faulty variables can be incorrectly

associated with faults. In contribution-based fault detection, groups of correlating variables are often displayed as faulty due to the smearing effect (van den Kerkhof et al., 2013); this is an obstacle especially for the often multicollinear data of bioprocesses.

To account for the nonlinearity of CSTR processes, several authors introduced the kernel PCA (Schölkopf et al., 1998) as a nonlinear extension of the PCA and adapted the calculation of the contributions to the T^2 and SPE statistics accordingly (Cho et al., 2005; Choi et al., 2005; Alcalá and Qin, 2010).

The functional principle of AANN has already been described above for fault detection using symptom signals. Ren et al. (2018) proposed a reconstruction-based AANN to detect faults in nonlinear processes (simulated gas turbine). Both single and multiple faults could be detected despite the occurrence of smearing effects. It was further shown that in this case reconstruction-based AANN is superior to the other investigated methods (contribution plots-based PCA, contribution plots-based AANN, and reconstruction-based PCA) in terms of detection rate.

In summary, methods based on variable contribution currently represent the largest share among studies on sensor fault detection in the process industry. The main advantage of these methods is that the MSPC model can be used both for process and sensor fault detection. With only one MSPC model it is theoretically possible to monitor all input variables or sensors, respectively. To the best of our knowledge, however, there is only one study (Krause et al., 2015) that shows that, for highly multicollinear bioprocess data, smearing effects do not prevent successful sensor fault detection. For multiphase processes with variable process lengths, the MSPC models used for defect detection can be developed separately for each phase and a phase-specific indicator variable can be used for time synchronization (Ündey et al., 2003).

Pattern Recognition Methods

Unsupervised (clustering) and supervised (classification and regression) pattern recognition has been applied extensively for bioprocess monitoring (Lourenço et al., 2012; Rodríguez-Méndez et al., 2016). Also, in the detection of sensor faults by pattern recognition, a distinction is made between unsupervised and supervised methods.

In the case of unsupervised pattern recognition, the training data consist of fault-free process data. The relationships within the process variables are learned as patterns. A specific deviation from the fault-free pattern can then be assigned to a specific sensor fault (Barbariol et al., 2020). In this context, unsupervised pattern recognition is comparable to the aforementioned methods based on variable contribution in MSPC models: First, a deviation from the fault-free standard process is detected and then it is examined to determine to which variable the fault can be traced. These two approaches (MSPC vs. unsupervised pattern recognition) differ less by this underlying principle than by the modeling methods used (empirical process model vs. clustering algorithm).

Barbariol et al. (2020) used unsupervised anomaly detection algorithms to detect faults of a multiphase flow meter. Artificial faults were added to data of normal operating conditions. The

type of fault was identified as either complete failure, bias, precision degradation, or drift by a root cause analysis algorithm.

In the case of supervised pattern recognition, the training data contain sensor faults. These sensor faults can be artificial or real, but in any case, they must be labeled according to their reliability r_i or—conversely—their degree of faultiness ($1 - r_i$). Faultiness is indicated either binarily (fault = true/false) or on a discrete (Mehranbod et al., 2003) or continuous scale (Guo and Nurre, 1991). Detecting sensor faults becomes a classification problem in case of binary or discrete faultiness and a regression problem in case of continuous faultiness. In both cases, labeled faulty data X_{hist} represent the inputs and the degree of faultiness (or the converse: r_i) represents the output. These input and output data are used for training the classification or regression model.

Guo and Nurre (1991) used supervised pattern recognition to detect and reconstruct sensor faults in a space shuttle main engine. Artificial random Gaussian noise was added to parts of fault-free data from normal operation. If the resulting artificial sensor readings are within the valid range, they are assigned a reliability of 0.9; if they are outside the valid range, they are assigned a reliability of 0.1. A feedforward ANN is trained with the manipulated sensor readings as inputs and the corresponding labeled reliability as outputs using a backpropagation algorithm to adjust the weights. In this way, even with a very small amount of original data, sensors whose readings do not match the rest of the sensor network can be identified. It was further shown that supervised pattern recognition is also suitable for the detection of multiple simultaneous sensor faults (Palmé et al., 2011) even in the presence of system failures (Romesis and Mathioudakis, 2003; Mathioudakis and Romessis, 2004). In this case, the training data must cover each of these cases (deviation from normal operation and multiple sensor faults), which causes the number of training patterns to increase rapidly.

In addition to the mere detection of faults, Mehranbod et al. (2003) distinguished between three different fault types (bias, drift, or noise) by identifying fault patterns in a moving window. They trained a Bayesian belief network to detect both single and multiple sensor faults in a polymerization reactor. This concept was later extended for the time-variant behavior of transient processes (Mehranbod et al., 2005).

In summary, pattern recognition methods are particularly attractive because ready-to-use—and in many toolboxes also auto-tuned—algorithms of machine learning can be applied to the problem of sensor fault detection without extensive statistical knowledge. At least in mechanical or chemical processes, efficient sensor fault detection can be realized with only little original data from normal operation together with artificial faults (Guo and Nurre, 1991). Despite this high potential, there are, to our knowledge, no studies that have explicitly used the previously trained pattern of sensor faults for their subsequent detection in bioprocesses. This lack of studies is all the more remarkable as pattern recognition methods are particularly efficient with such a high degree of multicollinearity as in bioprocesses.

Sensor Fault Tolerance

In the last sections, three different approaches to sensor fault detection were described. In the absence of sensor faults, the

model input to soft sensors is considered reliable (online validation). But what is the use of online validation if the fault detected in the input data makes the soft sensor prediction unreliable? We know that something is going wrong, but we cannot change anything (upper branch in Figure 5). This is where the fault tolerance of soft sensors comes into play.

In general, modules for fault tolerance can be implemented at two layers of a soft sensor: at the inputs or in the actual soft sensor model.

The first variant of fault-tolerant soft sensors is shown in Figure 5. Here, sensor faults are first detected and, after a decision logic, they are compensated for by a reconstruction of the faulty sensor reading. This reconstruction is equivalent to missing data imputation (Dunia et al., 1996). The outputs of the fault tolerance module in Figure 5 are the inputs to the soft sensor. The inputs and outputs of the described fault tolerant soft sensor are hereinafter referred to as $X_{on,FT}$ and $\hat{y}_{on,FT}$. For bioprocesses, there are, to our knowledge, no studies available that explicitly address the development a fault-tolerant soft sensor based on fault detection and reconstruction. However, some authors have separately described the fault tolerance module shown in Figure 5, as described in the following.

In the already mentioned study by Huang et al. (2002), an AANN was used for sensor fault detection by means of a symptom signal and fault reconstruction in a fermentation process with variable process length and time-variant behavior. Examples of sensor fault reconstruction using AANN in applications other than biotechnology are given in Kramer (1991), Kramer (1992), and Hamidreza et al. (2014). Variable contribution statistics (T^2 and SPE) in a MSPC model were used by Lawal and Zhang (2017) to

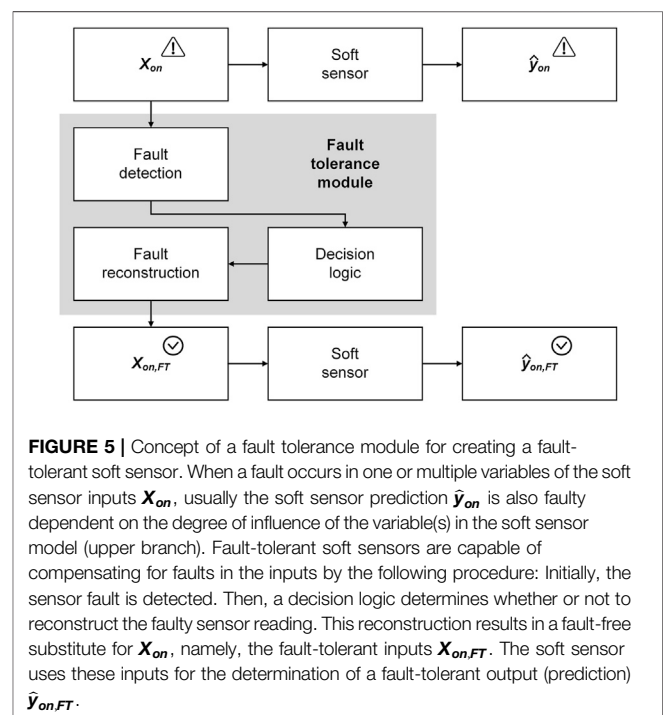


FIGURE 5 | Concept of a fault tolerance module for creating a fault-tolerant soft sensor. When a fault occurs in one or multiple variables of the soft sensor inputs X_{on} , usually the soft sensor prediction \hat{y}_{on} is also faulty dependent on the degree of influence of the variable(s) in the soft sensor model (upper branch). Fault-tolerant soft sensors are capable of compensating for faults in the inputs by the following procedure: Initially, the sensor fault is detected. Then, a decision logic determines whether or not to reconstruct the faulty sensor reading. This reconstruction results in a fault-free substitute for X_{on} , namely, the fault-tolerant inputs $X_{on,FT}$. The soft sensor uses these inputs for the determination of a fault-tolerant output (prediction) $\hat{y}_{on,FT}$.

detect and subsequently reconstruct faulty sensor reading within a crude distillation unit. In the above-mentioned study by Guo and Nurre (1991) an ANN was used to learn the patterns of both fault-free and faulty sensor readings to detect sensor faults. A separate ANN was trained to reconstruct the faulty readings.

In the second variant of fault-tolerant soft sensors, the faulty inputs are not reconstructed but the soft sensor algorithm itself is responsible for the fault management. For bioprocesses, there is, to our knowledge, only one study available that explicitly addresses fault tolerance by adapting the soft sensor models (Krause et al., 2015). The already described MSPC model developed by Krause et al. (2015) is capable of giving reliable predictions $\hat{y}_{on,FT}$ (here: higher-level process quantity) in the presence of sensor faults. As mentioned, PSO is used to find the best models with respect to historical process data. When sensor readings in X_{on} differ significantly from historical data X_{hist} , they are penalized by the PSO cost function. This in turn results in a drastically decreased contribution of the faulty sensor reading and thus a fault-tolerant prediction of \hat{y}_{on} . With this approach to sensor fault tolerance the same has to be considered as with the entire MSPC concept: Both are ultimately based purely on a statistically significant deviation from X_{on} to X_{hist} and are thus strongly dependent on the size and quality of the process data pool for X_{hist} .

In summary, it must be noted that, with very few exceptions, there are no studies on fault-tolerant soft sensors for the process industry. With regard to fault detection before the subsequent reconstruction, all three methods described above are applicable. However, for the methods of variable contribution in a MSPC model and pattern recognition methods, a separate model must be developed to reconstruct the faulty sensor reading. Symptom signal methods offer the advantage that the reconstructed sensor reading is directly available.

CONCLUSION

Based on an overview of the status quo of soft sensor development and online implementation, this review article describes the challenges of variable process lengths, multiple phases, and sensor faults, and critically discusses the corresponding solution approaches. The challenges are considered both individually and synchronously, and the solution approaches are evaluated in terms of their practicality and applicability to bioprocesses.

Variable process lengths: Data synchronization techniques are employed to ensure that soft sensors provide correct predictions despite variable process lengths. For data synchronization, indicator variable techniques and particularly DTW dominate the bioprocess literature compared to curve registration techniques. Indicator variables alone can only be used for the alignment of the entire process lengths. In contrast, DTW and curve registration techniques can additionally be used for the alignment of landmarks. Indicator variable techniques require a higher degree of process knowledge (selection of appropriate process variables etc.) compared to DTW and curve registration techniques. DTW is the technique of choice

when a solution is sought that does not require much process knowledge (compared to indicator variable techniques) and fine-tuning (compared to curve registration techniques).

Multiple process phases: The basic strategy for coping with multiple process phases is to divide the process datasets into individual phase segments and develop separate models for these segments. For the detection and division of process phases, trajectory-based and correlation-based methods have been proposed in the literature. Methods based on the progression of process trajectories, most notably via DTW, have to date been proposed more frequently in the bioprocess literature compared to correlation-based methods. Reasons for this include better comprehensibility of algorithms, easier interpretability of results, and coincidence with actual operational process phases in trajectory-based methods (Luo et al., 2016). On the other hand, correlation-based methods offer the advantage that they can be developed almost entirely without process knowledge. The consideration of phase transitions has so far been described only for trajectory-based methods (via FCM; Luo et al., 2016); for correlation-based methods, the consideration of phase transitions is still lacking.

Sensor faults: If the input to a soft sensor is faulty, there is a high probability that the output is faulty as well. Despite this obvious relation, studies on the detection of or even tolerance to sensor faults in bioprocesses are rare. Methods based on variable contributions in MSPC models are well established in the process industry for the identification of sensor faults. Further research is required to evaluate the applicability of these methods to highly collinear bioprocesses, as groups of correlating variables are often displayed as faulty due to smearing effects (van den Kerkhof et al., 2013). Symptom signal methods have been used to detect sensor faults and to reconstruct faults in bioprocesses. These methods, especially AANN, seem to be promising tools for the fault tolerance of soft sensors. The recognition of previously trained fault patterns has been used in mechanical engineering for fault detection, but to our knowledge has not yet been addressed in the bioprocess field. However, it can be assumed that this branch of machine learning will also increase in popularity in the field of bioprocesses due to steadily growing libraries of ready-to-use algorithms. For all three approaches presented for the detection of sensor faults (symptom signal, MSPC, pattern recognition) it could be shown that they are also capable of detecting simultaneously occurring sensor faults.

Synchronous consideration of the three challenges: The development of soft sensors for bioprocesses with multiple phases and variable process lengths has been investigated in several studies (e.g., Ündey et al., 2003; Luo et al., 2016). As described above, landmark-based data synchronization is particularly suitable for multiphase processes. For sensor fault detection for bioprocesses with variable lengths but without multiple phases individual studies exist (Krause et al., 2015; Brunner et al., 2019). Regarding sensor fault detection for multiphase bioprocesses with variable lengths, the question remains open as to which of the three methods presented is most suitable. This is because there is to the best of our knowledge only one study that provides a solution for the synchronous occurrence of all three challenges for bioprocesses (Huang et al., 2002).

The core conclusions of this review article are as follows:

- The choice of methods to handle variable process lengths and multiple process phases is dependent on the level of implementable process knowledge.
- The dilemma with sensor fault detection via soft sensors is that the input to the soft sensor can itself be erroneous.
- There is a clear research gap regarding the validation of the input data to soft sensors.
- Specifically, approaches to the tolerance of soft sensors to sensor faults need to be found.

Closing these gaps not only will allow existing sensor networks to be used more efficiently to monitor bioprocesses but will also strengthen confidence in soft sensors and PAT.

REFERENCES

- Afseth, N. K., Segtnan, V. H., and Wold, J. P. (2006). Raman Spectra of Biological Samples: A Study of Preprocessing Methods. *Appl. Spectrosc.* 60, 1358–1367. doi:10.1366/000370206779321454
- Alag, S., Agogino, A. M., and Morjaria, M. (2001). A Methodology for Intelligent Sensor Measurement, Validation, Fusion, and Fault Detection for Equipment Monitoring and Diagnostics. *Artif. Intell. Eng. Des. Anal. Manuf.* 15, 307–320. doi:10.1017/s0890060401154053
- Alcala, C. F., and Joe Qin, S. (2011). Analysis and Generalization of Fault Diagnosis Methods for Process Monitoring. *J. Process Control* 21, 322–330. doi:10.1016/j.jprocont.2010.10.005
- Alcala, C. F., and Qin, S. J. (2009). Reconstruction-Based Contribution for Process Monitoring. *Automatica* 45, 1593–1600. doi:10.1016/j.automatica.2009.02.027
- Alcala, C. F., and Qin, S. J. (2010). Reconstruction-based Contribution for Process Monitoring with Kernel Principal Component Analysis. *Ind. Eng. Chem. Res.* 49, 7849–7857. doi:10.1021/ie9018947
- Andersen, S. W., and Runger, G. C. (2012). Automated Feature Extraction from Profiles with Application to a Batch Fermentation Process. *J. R. Stat. Soc. Ser. C* 61, 327–344. doi:10.1111/j.1467-9876.2011.01032.x
- Armaou, A., and Demetriou, M. A. (2008). Robust Detection and Accommodation of Incipient Component and Actuator Faults in Nonlinear Distributed Processes. *AIChE J.* 54, 2651–2662. doi:10.1002/aic.11539
- Balaban, E., Saxena, A., Bansal, P., Goebel, K. F., and Curran, S. (2009). Modeling, Detection, and Disambiguation of Sensor Faults for Aerospace Applications. *IEEE Sens. J.* 9, 1907–1917. doi:10.1109/jksen.2009.2030284
- Barbariol, T., Feltresi, E., and Susto, G. A. (2020). Self-Diagnosis of Multiphase Flow Meters Through Machine Learning-Based Anomaly Detection. *Energies* 13, 3136. doi:10.3390/en13123136
- Bayer, B., Stosch, M., Melcher, M., Duerkop, M., and Striedner, G. (2020). Soft Sensor Based on 2D-Fluorescence and Process Data Enabling Real-time Estimation of Biomass in *Escherichia coli* Cultivations. *Eng. Life Sci.* 20, 26–35. doi:10.1002/elsc.201900076
- Berbegal, C., Khomenko, I., Russo, P., Spano, G., Fragasso, M., Biasioli, F., et al. (2020). PTR-ToF-MS for the Online Monitoring of Alcoholic Fermentation in Wine: Assessment of VOCs Variability Associated with Different Combinations of *Saccharomyces/Non-Saccharomyces* as a Case-Study. *Fermentation* 6, 55. doi:10.3390/fermentation6020055
- Besenhard, M. O., Scheibelhofer, O., François, K., Joksch, M., and Kavsek, B. (2018). A Multivariate Process Monitoring Strategy and Control Concept for a Small-Scale Fermenter in a PAT Environment. *J. Intell. Manuf.* 29, 1501–1514. doi:10.1007/s10845-015-1192-8
- Bidar, B., Shahraki, F., Sadeghi, J., and Khalilipour, M. M. (2018). Soft Sensor Modeling Based on Multi-State-Dependent Parameter Models and Application for Quality Monitoring in Industrial Sulfur Recovery Process. *IEEE Sens. J.* 18, 4583–4591. doi:10.1109/jksen.2018.2818886

AUTHOR CONTRIBUTIONS

VB reviewed the literature and drafted the manuscript. MS, DG, and TB edited the manuscript. All authors have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

FUNDING

We appreciate the support from the German Federal Ministry of Education and Research (project 031B0475E), the German Research Foundation (project BE 2245/17-1), and the Open Access Publishing Fund of the Technical University of Munich (TUM).

- Biechele, P., Busse, C., Solle, D., Scheper, T., and Reardon, K. (2015). Sensor Systems for Bioprocess Monitoring. *Eng. Life Sci.* 15, 469–488. doi:10.1002/elsc.201500014
- Biermann, F., Mathews, J., Niefing, B., König, N., and Schmitt, R. H. (2021). Automating Laboratory Processes by Connecting Biotech and Robotic Devices—An Overview of the Current Challenges, Existing Solutions and Ongoing Developments. *Processes* 9, 966. doi:10.3390/pr9060966
- Bigot, J. (2006). Landmark-Based Registration of Curves via the Continuous Wavelet Transform. *J. Comput. Graph. Stat.* 15, 542–564. doi:10.1198/106186006x133023
- Birle, S., Hussein, M. A., and Becker, T. (2013). Fuzzy Logic Control and Soft Sensing Applications in Food and Beverage Processes. *Food Control* 29, 254–269. doi:10.1016/j.foodcont.2012.06.011
- Birle, S., Hussein, M. A., and Becker, T. (2015). On-Line Yeast Propagation Process Monitoring and Control Using an Intelligent Automatic Control System. *Eng. Life Sci.* 15, 83–95. doi:10.1002/elsc.201400058
- Brunner, V., Klöckner, L., Kerpes, R., Geier, D. U., and Becker, T. (2019). Online Sensor Validation in Sensor Networks for Bioprocess Monitoring Using Swarm Intelligence. *Anal. Bioanal. Chem.* 412, 2165–2175. doi:10.1007/s00216-019-01927-7
- Brunner, V., Siegl, M., Geier, D., and Becker, T. (2020). Biomass Soft Sensor for a *Pichia* Pastorised-batch Process Based on Phase Detection and Hybrid Modeling. *Biotechnol. Bioeng.* 117, 2749–2759. doi:10.1002/bit.27454
- Buyel, J. F., Twyman, R. M., and Fischer, R. (2017). Very-Large-Scale Production of Antibodies in Plants: The Biologization of Manufacturing. *Biotechnol. Adv.* 35, 458–465. doi:10.1016/j.biotechadv.2017.03.011
- Cabaneros Lopez, P., Udugama, I. A., Thomsen, S. T., Roslander, C., Junicke, H., Iglesias, M. M., et al. (2021). Transforming Data to Information: A Parallel Hybrid Model for Real-time State Estimation in Lignocellulosic Ethanol Fermentation. *Biotechnol. Bioeng.* 118, 579–591. doi:10.1002/bit.27586
- Camacho, J., Picó, J., and Ferrer, A. (2008). Multi-phase Analysis Framework for Handling Batch Process Data. *J. Chemom.* 22, 632–643. doi:10.1002/cem.1151
- Camacho, J., and Picó, J. (2006). Multi-Phase Principal Component Analysis for Batch Processes Modelling. *Chemom. Intell. Lab. Syst.* 81, 127–136. doi:10.1016/j.chemolab.2005.11.003
- Capito, F., Skudas, R., Kolmar, H., and Hunzinger, C. (2015). At-Line Mid Infrared Spectroscopy for Monitoring Downstream Processing Unit Operations. *Process Biochem.* 50, 997–1005. doi:10.1016/j.procbio.2015.03.005
- Cawley, G. C., and Talbot, N. L. C. (2010). On Over-fitting in Model Selection and Subsequent Selection Bias in Performance Evaluation. *J. Machine Learn. Res.* 11, 2079–2107.
- Chen, Y., Yang, O., Sampat, C., Bhalode, P., Ramachandran, R., and Ierapetritou, M. (2020). Digital Twins in Pharmaceutical and Biopharmaceutical Manufacturing: A Literature Review. *Processes* 8, 1088. doi:10.3390/pr8091088
- Chew, W., and Sharratt, P. (2010). Trends in Process Analytical Technology. *Anal. Methods* 2, 1412. doi:10.1039/c0ay00257g
- Cho, J.-H., Lee, J.-M., Wook Choi, S., Lee, D., and Lee, I.-B. (2005). Fault Identification for Process Monitoring Using Kernel Principal Component Analysis. *Chem. Eng. Sci.* 60, 279–288. doi:10.1016/j.ces.2004.08.007

- Choi, S. W., Lee, C., Lee, J.-M., Park, J. H., and Lee, I.-B. (2005). Fault Detection and Identification of Nonlinear Processes Based on Kernel PCA. *Chemom. Intell. Lab. Syst.* 75, 55–67. doi:10.1016/j.chemolab.2004.05.001
- Chong, I.-G., and Jun, C.-H. (2005). Performance of Some Variable Selection Methods when Multicollinearity Is Present. *Chemom. Intell. Lab. Syst.* 78, 103–112. doi:10.1016/j.chemolab.2004.12.011
- Claßen, J., Aupert, F., Reardon, K. F., Solle, D., and Scheper, T. (2017). Spectroscopic Sensors for In-Line Bioprocess Monitoring in Research and Pharmaceutical Industrial Application. *Anal. Bioanal. Chem.* 409, 651–666. doi:10.1007/s00216-016-0068-x
- Das, A., Maiti, J., and Banerjee, R. N. (2012). Process Monitoring and Fault Detection Strategies: A Review. *Int. J. Qual. Reliab. Manage.* 29, 720–752. doi:10.1108/02656711211258508
- Doan, X.-T., and Srinivasan, R. (2008). Online Monitoring of Multi-phase Batch Processes Using Phase-Based Multivariate Statistical Process Control. *Comput. Chem. Eng.* 32, 230–243. doi:10.1016/j.compchemeng.2007.05.010
- Dumarey, M., Hermanto, M., Airiau, C., Shapland, P., Robinson, H., Hamilton, P., et al. (2019). Advances in Continuous Active Pharmaceutical Ingredient (API) Manufacturing: Real-Time Monitoring Using Multivariate Tools. *J. Pharm. Innov.* 14, 359–372. doi:10.1007/s12247-018-9348-7
- Dunia, R., Qin, S. J., Edgar, T. F., and McAvoy, T. J. (1996). Identification of Faulty Sensors Using Principal Component Analysis. *AIChE J.* 42, 2797–2812. doi:10.1002/aic.690421011
- EC (2010). Good Manufacturing Practice Medicinal Products for Human and Veterinary Use – Annex 11: Computerised Systems. Available at: https://ec.europa.eu/health/sites/default/files/files/eudralex/vol-4/annex11_01-2011_en.pdf.
- FDA (2004). Guidance for Industry, PAT-A Framework for Innovative Pharmaceutical Development, Manufacturing and Quality Assurance. Available at: <http://www.fda.gov/cder/guidance/published.html>.
- FDA (2003). Guidance for Industry: Part 11, Electronic Records; Electronic Signatures—Scope and Application. Available at: <https://www.fda.gov/regulatory-information/search-fda-guidance-documents/part-11-electronic-records-electronic-signatures-scope-and-application>.
- Feital, T., and Pinto, J. C. (2015). Use of Variance Spectra for In-Line Validation of Process Measurements in Continuous Processes. *Can. J. Chem. Eng.* 93, 1426–1437. doi:10.1002/cjce.22219
- Fortuna, L., Graziani, S., Rizzo, A., and Xibilia, M. G. (2007). *Soft Sensors for Monitoring and Control of Industrial Processes*. Berlin/Heidelberg: Springer Science & Business Media.
- Fransson, M., and Folestad, S. (2006). Real-time Alignment of Batch Process Data Using COW for On-Line Process Monitoring. *Chemom. Intell. Lab. Syst.* 84, 56–61. doi:10.1016/j.chemolab.2006.04.020
- García-Muñoz, S., Kourti, T., MacGregor, J. F., Mateos, A. G., and Murphy, G. (2003). Troubleshooting of an Industrial Batch Process Using Multivariate Methods. *Ind. Eng. Chem. Res.* 42, 3592–3601. doi:10.1021/ie0300023
- García-Muñoz, S., Polizzi, M., Prpich, A., Strain, C., Lalonde, A., and Negron, V. (2011). Experiences in Batch Trajectory Alignment for Pharmaceutical Process Improvement Through Multivariate Latent Variable Modelling. *J. Process Control.* 21, 1370–1377. doi:10.1016/j.jprocont.2011.07.013
- Gargalo, C. L., Udugama, I., Pontius, K., Lopez, P. C., Nielsen, R. F., Hasanzadeh, A., et al. (2020). Towards Smart Biomanufacturing: a Perspective on Recent Developments in Industrial Measurement and Monitoring Technologies for Bio-Based Production Processes. *J. Ind. Microbiol. Biotechnol.* 47, 947–964. doi:10.1007/s10295-020-02308-1
- Gins, G., van den Kerkhof, P., and van Impe, J. F. M. (2012). Hybrid Derivative Dynamic Time Warping for Online Industrial Batch-End Quality Estimation. *Ind. Eng. Chem. Res.* 51, 6071–6084. doi:10.1021/ie2019068
- Gollmer, K., and Posten, C. (1996). Supervision of Bioprocesses Using a Dynamic Time Warping Algorithm. *Control Eng. Pract.* 4, 1287–1295. doi:10.1016/0967-0661(96)00136-0
- Gomis-Fons, J., Schwarz, H., Zhang, L., Andersson, N., Nilsson, B., Castan, A., et al. (2020). Model-based Design and Control of a Small-Scale Integrated Continuous End-to-End mAb Platform. *Biotechnol. Prog.* 36, e2995. doi:10.1002/btpr.2995
- González-Martínez, J. M., Noord, O. E. de., and Ferrer, A. (2014). Multisynchro: a Novel Approach for Batch Synchronization in Scenarios of Multiple Asynchronisms. *J. Chemometrics* 28, 462–475.
- González-Martínez, J. M., Ferrer, A., and Westerhuis, J. A. (2011). Real-time Synchronization of Batch Trajectories for On-Line Multivariate Statistical Process Control Using Dynamic Time Warping. *Chemom. Intell. Lab. Syst.* 105, 195–206. doi:10.1016/j.chemolab.2011.01.003
- González-Martínez, J. M., Westerhuis, J. A., and Ferrer, A. (2013). Using Warping Information for Batch Process Monitoring and Fault Classification. *Chemom. Intell. Lab. Syst.* 127, 210–217. doi:10.1016/j.chemolab.2013.07.003
- Guo, T.-H., and Nurre, J. (1991). “Sensor Failure Detection and Recovery by Neural Networks,” in IJCNN-91-Seattle International Joint Conference on Neural Networks, Seattle, WA, July 8–12, 1991.
- Haimi, H., Mulas, M., Corona, F., and Vahala, R. (2013). Data-Derived Soft-Sensors for Biological Wastewater Treatment Plants: An Overview. *Environ. Model. Softw.* 47, 88–107. doi:10.1016/j.envsoft.2013.05.009
- Hamidreza, M., Mehdi, S., Hooshang, J.-R., and Aliakbar, N. (2014). Reconstruction Based Approach to Sensor Fault Diagnosis Using Auto-Associative Neural Networks. *J. Cent. South. Univ.* 21, 2273–2281. doi:10.1007/s11771-014-2178-y
- Hawkins, D. M. (2004). The Problem of Overfitting. *J. Chem. Inf. Comput. Sci.* 44, 1–12. doi:10.1021/ci0342472
- Heinze, G., Wallisch, C., and Dunkler, D. (2018). Variable Selection - A Review and Recommendations for the Practicing Statistician. *Biom. J.* 60, 431–449. doi:10.1002/bimj.201700067
- Huang, J., Shimizu, H., and Shioya, S. (2002). Data Preprocessing and Output Evaluation of an Autoassociative Neural Network Model for Online Fault Detection in Virginiamycin Production. *J. Biosci. Bioeng.* 94, 70–77. doi:10.1016/s1389-1723(02)80119-0
- Isermann, R. (2011). *Fault-Diagnosis Applications: Model-Based Condition Monitoring: Actuators, Drives, Machinery, Plants, Sensors, and Fault-Tolerant Systems*. Berlin/Heidelberg: Springer Science & Business Media.
- Isermann, R. (2006). *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*. Berlin/Heidelberg: Springer Science & Business Media.
- ISPE (2008). GAMP 5 Guide: Compliant GxP Computerized Systems. Available at: <https://ispe.org/publications/guidance-documents/gamp-5>.
- Jenzsch, M., Simutis, R., Eisbrenner, G., Stückrath, I., and Lübbert, A. (2006). Estimation of Biomass Concentrations in Fermentation Processes for Recombinant Protein Production. *Bioproc. Biosyst. Eng.* 29, 19–27. doi:10.1007/s00449-006-0051-6
- Jin, H., Chen, X., Yang, J., Zhang, H., Wang, L., and Wu, L. (2015). Multi-Model Adaptive Soft Sensor Modeling Method Using Local Learning and Online Support Vector Regression for Nonlinear Time-Variant Batch Processes. *Chem. Eng. Sci.* 131, 282–303. doi:10.1016/j.ces.2015.03.038
- Kabugo, J. C., Jämsä-Jounela, S.-L., Schiemann, R., and Binder, C. (2020). Industry 4.0 Based Process Data Analytics Platform: A Waste-To-Energy Plant Case Study. *Int. J. Electr. Power Energy Syst.* 115, 105508. doi:10.1016/j.ijepes.2019.105508
- Kadlec, P., Gabrys, B., and Strandt, S. (2009). Data-driven Soft Sensors in the Process Industry. *Comput. Chem. Eng.* 33, 795–814. doi:10.1016/j.compchemeng.2008.12.012
- Kadlec, P., Grbić, R., and Gabrys, B. (2011). Review of Adaptation Mechanisms for Data-Driven Soft Sensors. *Comput. Chem. Eng.* 35, 1–24. doi:10.1016/j.compchemeng.2010.07.034
- Kano, M., and Fujiwara, K. (2012). Virtual Sensing Technology in Process Industries: Trends and Challenges Revealed by Recent Industrial Applications. *J. Chem. Eng. Jpn.* 46, 1–17. doi:10.1252/jcej.12we167
- Kassidas, A., MacGregor, J. F., and Taylor, P. A. (1998). Synchronization of Batch Trajectories Using Dynamic Time Warping. *AIChE J.* 44, 864–875. doi:10.1002/aic.690440412
- Keogh, E. J., and Pazzani, M. J. (2001). “Derivative Dynamic Time Warping,” in Proceedings of the First SIAM International Conference on Data Mining, Chicago, IL, April 5–7, 2001. Editor R. Grossman (Philadelphia, PA: SIAM), 1–11.
- Kern, S., Wander, L., Meyer, K., Guhl, S., Mikkula, A. R. G., Holtkamp, M., et al. (2019). Flexible Automation with Compact NMR Spectroscopy for Continuous Production of Pharmaceuticals. *Anal. Bioanal. Chem.* 411, 3037–3046. doi:10.1007/s00216-019-01752-y
- Khatibisepehr, S., Huang, B., and Khare, S. (2013). Design of Inferential Sensors in the Process Industry: A Review of Bayesian Methods. *J. Process Control.* 23, 1575–1596. doi:10.1016/j.jprocont.2013.05.007

- Kourti, T. (2005). Application of Latent Variable Methods to Process Control and Multivariate Statistical Process Control in Industry. *Int. J. Adapt. Control. Signal. Process.* 19, 213–246. doi:10.1002/acs.859
- Kramer, M. A. (1992). Autoassociative Neural Networks. *Comput. Chem. Eng.* 16, 313–328. doi:10.1016/0098-1354(92)80051-a
- Kramer, M. A. (1991). Nonlinear Principal Component Analysis Using Autoassociative Neural Networks. *AIChE J.* 37, 233–243. doi:10.1002/aic.690370209
- Krause, D., Hussein, M. A., and Becker, T. (2015). Online Monitoring of Bioprocesses via Multivariate Sensor Prediction within Swarm Intelligence Decision Making. *Chemom. Intell. Lab. Syst.* 145, 48–59. doi:10.1016/j.chemolab.2015.04.012
- Krippel, M., Kargl, T., Duerkop, M., and Dürauer, A. (2021). Hybrid Modeling Reduces Experimental Effort to Predict Performance of Serial and Parallel Single-Pass Tangential Flow Filtration. *Separat. Purif. Technol.* 276, 119277. doi:10.1016/j.seppur.2021.119277
- Kullaa, J. (2013). Detection, Identification, and Quantification of Sensor Fault in a Sensor Network. *Mech. Syst. Signal Process.* 40, 208–221. doi:10.1016/j.ymsp.2013.05.007
- Kullback, S., and Leibler, R. A. (1951). On Information and Sufficiency. *Ann. Math. Statist.* 22, 79–86. doi:10.1214/aoms/1177729694
- Lawal, S. A., and Zhang, J. (2017). “Actuator and Sensor Fault Tolerant Control of a Crude Distillation Unit,” in *27th European Symposium on Computer Aided Process Engineering*. Editors A. Espuña, M. Graells, and L. Puigjaner (Amsterdam, Boston, Heidelberg: Elsevier), 1705–1710. doi:10.1016/b978-0-444-63965-3.50286-5
- Liu, Y.-J., André, S., Saint Cristau, L., Lagresle, S., Hannas, Z., Calvosa, É., et al. (2017). Multivariate Statistical Process Control (MSPC) Using Raman Spectroscopy for In-Line Culture Cell Monitoring Considering Time-Varying Batches Synchronized with Correlation Optimized Warping (COW). *Analytica Chim. Acta* 952, 9–17. doi:10.1016/j.aca.2016.11.064
- Lourenço, N. D., Lopes, J. A., Almeida, C. F., Sarragaça, M. C., and Pinheiro, H. M. (2012). Bioreactor Monitoring with Spectroscopy and Chemometrics: a Review. *Anal. Bioanal. Chem.* 404, 1211–1237. doi:10.1007/s00216-012-6073-9
- Lu, N., and Gao, F. (2005). Stage-Based Process Analysis and Quality Prediction for Batch Processes. *Ind. Eng. Chem. Res.* 44, 3547–3555. doi:10.1021/ie048852l
- Lu, N., Gao, F., and Wang, F. (2004). Sub-PCA Modeling and On-Line Monitoring Strategy for Batch Processes. *AIChE J.* 50, 255–259. doi:10.1002/aic.10024
- Luo, L., Bao, S., Mao, J., Tang, D., and Gao, Z. (2016). Fuzzy Phase Partition and Hybrid Modeling Based Quality Prediction and Process Monitoring Methods for Multiphase Batch Processes. *Ind. Eng. Chem. Res.* 55, 4045–4058. doi:10.1021/acs.iecr.5b04252
- Luttmann, R., Bracewell, D. G., Cornelissen, G., Gernaey, K. V., Glassey, J., Hass, V. C., et al. (2012). Soft Sensors in Bioprocessing: A Status Report and Recommendations. *Biotechnol. J.* 7, 1040–1048. doi:10.1002/biot.201100506
- Mandenius, C.-F., and Gustavsson, R. (2015). Mini-Review: Soft Sensors as Means for PAT in the Manufacture of Bio-Therapeutics. *J. Chem. Technol. Biotechnol.* 90, 215–227. doi:10.1002/jctb.4477
- Mathioudakis, K., and Romesis, C. (2004). Probabilistic Neural Networks for Validation of On-Board Jet Engine Data. *Proc. Inst. Mech. Eng. G: J. Aerospace Eng.* 218, 59–72. doi:10.1177/095441000421800105
- Matthews, T. E., Berry, B. N., Smelko, J., Moretto, J., Moore, B., and Wiltberger, K. (2016). Closed Loop Control of Lactate Concentration in Mammalian Cell Culture by Raman Spectroscopy Leads to Improved Cell Density, Viability, and Biopharmaceutical Protein Production. *Biotechnol. Bioeng.* 113, 2416–2424. doi:10.1002/bit.26018
- Mehranbod, N., Soroush, M., and Panjapornpon, C. (2005). A Method of Sensor Fault Detection and Identification. *J. Process Control.* 15, 321–339. doi:10.1016/j.jprocont.2004.06.009
- Mehranbod, N., Soroush, M., Piovoso, M., and Ogunnaike, B. A. (2003). Probabilistic Model for Sensor Fault Detection and Identification. *AIChE J.* 49, 1787–1802. doi:10.1002/aic.690490716
- Mei, C., Su, Y., Liu, G., Ding, Y., and Liao, Z. (2017). Dynamic Soft Sensor Development Based on Gaussian Mixture Regression for Fermentation Processes. *Chin. J. Chem. Eng.* 25, 116–122. doi:10.1016/j.cjche.2016.07.005
- Melcher, M., Scharl, T., Spangl, B., Luchner, M., Cserjan, M., Bayer, K., et al. (2015). The Potential of Random forest and Neural Networks for Biomass and Recombinant Protein Modeling in Escherichia Colifed-Batch Fermentations. *Biotechnol. J.* 10, 1770–1782. doi:10.1002/biot.201400790
- Meng, Y., Lan, Q., Qin, J., Yu, S., Pang, H., and Zheng, K. (2019). Data-Driven Soft Sensor Modeling Based on Twin Support Vector Regression for Cane Sugar Crystallization. *J. Food Eng.* 241, 159–165. doi:10.1016/j.jfoodeng.2018.07.035
- Mnassri, B., El Adel, E. M., and Ouladsine, M. (2015). Reconstruction-based Contribution Approaches for Improved Fault Diagnosis Using Principal Component Analysis. *J. Process Control.* 33, 60–76. doi:10.1016/j.jprocont.2015.06.004
- Moser, U., and Schramm, D. (2019). Multivariate Dynamic Time Warping in Automotive Applications: A Review. *Intell. Data Anal.* 23, 535–553. doi:10.3233/ida-184130
- Nair, A. M., Hykkerud, A., and Ratnaweera, H. (2020). A Cost-Effective IoT Strategy for Remote Deployment of Soft Sensors - A Case Study on Implementing a Soft Sensor in a Multistage MBBR Plant. *Water Sci. Technol.* 81, 1733–1739. doi:10.2166/wst.2020.067
- Nielsen, N.-P. V., Carstensen, J. M., and Smedsgaard, J. (1998). Aligning of Single and Multiple Wavelength Chromatographic Profiles for Chemometric Data Analysis Using Correlation Optimised Warping. *J. Chromatogr. A* 805, 17–35. doi:10.1016/s0021-9673(98)00021-1
- Nomikos, P., and MacGregor, J. F. (1995). Multivariate SPC Charts for Monitoring Batch Processes. *Technometrics* 37, 41–59. doi:10.1080/00401706.1995.10485888
- OECD (2014). *Guidance Document on the Validation of (Quantitative) Structure-Activity Relationship [(Q)SAR] Models*. Paris.
- Ohadi, K., Legge, R. L., and Budman, H. M. (2015). Development of a Soft-Sensor Based on Multi-Wavelength Fluorescence Spectroscopy and a Dynamic Metabolic Model for Monitoring Mammalian Cell Cultures. *Biotechnol. Bioeng.* 112, 197–208. doi:10.1002/bit.25339
- Pais, D. A. M., Galvão, P. R. S., Kryzhanska, A., Barbau, J., Isidoro, I. A., and Alves, P. M. (2020). Holographic Imaging of Insect Cell Cultures: Online Non-invasive Monitoring of Adeno-Associated Virus Production and Cell Concentration. *Processes* 8, 487. doi:10.3390/pr8040487
- Palmé, T., Fast, M., and Thern, M. (2011). Gas Turbine Sensor Validation Through Classification with Artificial Neural Networks. *Appl. Energ.* 88, 3898–3904. doi:10.1016/j.apenergy.2011.03.047
- Pani, A. K., and Mohanta, H. K. (2011). A Survey of Data Treatment Techniques for Soft Sensor Design. *Chem. Product. Process Model.* 6. doi:10.2202/1934-2659.1536
- Pappenreiter, M., Sissolak, B., Sommeregger, W., and Striedner, G. (2019). Oxygen Uptake Rate Soft-Sensing via Dynamic kLa Computation: Cell Volume and Metabolic Transition Prediction in Mammalian Bioprocesses. *Front. Bioeng. Biotechnol.* 7, 195. doi:10.3389/fbioe.2019.00195
- Paquet-Durand, O., Assawarajuwan, S., and Hitzmann, B. (2017). Artificial Neural Network for Bioprocess Monitoring Based on Fluorescence Measurements: Training without Offline Measurements. *Eng. Life Sci.* 17, 874–880.
- Pearson, R. K., Neuvo, Y., Astola, J., and Gabbouj, M. (2016). Generalized Hampel Filters. *EURASIP J. Adv. Signal Process.* 2016, 1–18.
- Perla, R., Mukhopadhyay, S., and Samanta, A. N. (2004). “Sensor Fault Detection and Isolation Using Artificial Neural Networks,” in 2004 IEEE Region 10 Conference TENCON, Chiang Mai, Thailand, November 24, 2004.
- Qin, S. J. (2003). Statistical Process Monitoring: Basics and Beyond. *J. Chemom.* 17, 480–502. doi:10.1002/cem.800
- Ramsay, J. O., and Silverman, B. W. (2005). *Functional Data Analysis*. New York, NY: Springer Science+Business Media, Inc.
- Randek, J., and Mandenius, C.-F. (2018). On-line Soft Sensing in Upstream Bioprocessing. *Crit. Rev. Biotechnol.* 38, 106–121. doi:10.1080/07388551.2017.1312271
- Rathore, A. S., Mishra, S., Nikita, S., and Priyanka, P. (2021). Bioprocess Control: Current Progress and Future Perspectives. *Life* 11, 557. doi:10.3390/life11060557
- Rathore, A. S., and Winkle, H. (2009). Quality by Design for Biopharmaceuticals. *Nat. Biotechnol.* 27, 26–34. doi:10.1038/nbt0109-26
- Rato, T. J., Rendall, R., Gomes, V., Chin, S.-T., Chiang, L. H., Saraiva, P. M., et al. (2016). A Systematic Methodology for Comparing Batch Process Monitoring Methods: Part I-Assessing Detection Strength. *Ind. Eng. Chem. Res.* 55, 5342–5358. doi:10.1021/acs.iecr.5b04851
- Rato, T. J., Rendall, R., Gomes, V., Saraiva, P. M., and Reis, M. S. (2018). A Systematic Methodology for Comparing Batch Process Monitoring Methods:

- Part II-Assessing Detection Speed. *Ind. Eng. Chem. Res.* 57, 5338–5350. doi:10.1021/acs.iecr.7b04911
- Ren, S., Si, F., Zhou, J., Qiao, Z., and Cheng, Y. (2018). A New Reconstruction-Based Auto-Associative Neural Network for Fault Diagnosis in Nonlinear Systems. *Chemom. Intell. Lab. Syst.* 172, 118–128. doi:10.1016/j.chemolab.2017.12.005
- Rodríguez-Méndez, M. L., Saja, J. A. de., González-Antón, R., García-Hernández, C., Medina-Plaza, C., García-Cabezón, C., et al. (2016). Electronic Noses and Tongues in Wine Industry. *Front. Bioeng. Biotechnol.* 4, 81. doi:10.3389/fbioe.2016.00081
- Romesis, C., and Mathioudakis, K. (2003). Setting up of a Probabilistic Neural Network for Sensor Fault Detection Including Operation with Component Faults. *J. Eng. Gas Turbines Power* 125, 634–641. doi:10.1115/1.1582493
- Sakoe, H., and Chiba, S. (1978). Dynamic Programming Algorithm Optimization for Spoken Word Recognition. *IEEE Trans. Acoust. Speech, Signal. Process.* 26, 43–49. doi:10.1109/tassp.1978.1163055
- Sánchez-Fernández, A., Baldán, F. J., Sainz-Palmero, G. I., Benítez, J. M., and Fuente, M. J. (2018). Fault Detection Based on Time Series Modeling and Multivariate Statistical Process Control. *Chemometrics Intell. Lab. Syst.* 182, 57–69. doi:10.1016/j.chemolab.2018.08.003
- Sapto, A. (2014). State of the Art in the Development of Adaptive Soft Sensors Based on Just-In-Time Models. *Proced. Chem.* 9, 226–234. doi:10.1016/j.proche.2014.05.027
- Sauer, D. G., Melcher, M., Mosor, M., Walch, N., Berkemeyer, M., Scharl-Hirsch, T., et al. (2019). Real-time Monitoring and Model-based Prediction of Purity and Quantity during a Chromatographic Capture of Fibroblast Growth Factor 2. *Biotechnol. Bioeng.* 116, 1999–2009. doi:10.1002/bit.26984
- Scheper, T., Beutel, S., McGuinness, N., Heiden, S., Oldiges, M., Lammers, F., et al. (2021). Digitalization and Bioprocessing: Promises and Challenges. *Adv. Biochem. Eng. Biotechnol.* 176, 57–69. doi:10.1007/10_2020_139
- Schölkopf, B., Smola, A., and Müller, K.-R. (1998). Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Comput.* 10, 1299–1319. doi:10.1162/089976698300017467
- Sharma, A. B., Golubchik, L., and Govindan, R. (2010). Sensor Faults. *ACM Trans. Sen. Netw.* 6, 1–39. doi:10.1145/1754414.1754419
- Shokoohi-Yekta, M., Wang, J., and Keogh, E. (2015). “On the Non-trivial Generalization of Dynamic Time Warping to the Multi-Dimensional Case,” in *Proceedings of the 2015 SIAM International Conference on Data Mining*. Editors S. Venkatasubramanian and J. Ye (Philadelphia, PA: Society for Industrial and Applied Mathematics).
- Simon, L. L., Pataki, H., Marosi, G., Meemken, F., Hungerbühler, K., Baiker, A., et al. (2015). Assessment of Recent Process Analytical Technology (PAT) Trends: A Multi-author Review. *Org. Process. Res. Dev.* 19, 3–62. doi:10.1021/op500261y
- Sokolov, M., Soos, M., Neunstoecklin, B., Morbidelli, M., Butté, A., Leardi, R., et al. (2015). Fingerprint Detection and Process Prediction by Multivariate Analysis of Fed-Batch Monoclonal Antibody Cell Culture Data. *Biotechnol. Prog.* 31, 1633–1644. doi:10.1002/btpr.2174
- Solle, D., Hitzmann, B., Herwig, C., Pereira Remelhe, M., Ulonska, S., Wuerth, L., et al. (2017). Between the Poles of Data-Driven and Mechanistic Modeling for Process Operation. *Chem. Ingenieur Technik* 89, 542–561. doi:10.1002/cite.201600175
- Souza, F. A. A., Araújo, R., and Mendes, J. (2016). Review of Soft Sensor Methods for Regression Applications. *Chemom. Intell. Lab. Syst.* 152, 69–79. doi:10.1016/j.chemolab.2015.12.011
- Spann, R., Gernaey, K. V., and Sin, G. (2019). A Compartment Model for Risk-Based Monitoring of Lactic Acid Bacteria Cultivations. *Biochem. Eng. J.* 151, 107293. doi:10.1016/j.bej.2019.107293
- Spooer, M., Kold, D., and Kulahci, M. (2018). Harvest Time Prediction for Batch Processes. *Comput. Chem. Eng.* 117, 32–41. doi:10.1016/j.compchemeng.2018.05.019
- Spooer, M., Kold, D., and Kulahci, M. (2017). Selecting Local Constraint for Alignment of Batch Process Data with Dynamic Time Warping. *Chemom. Intell. Lab. Syst.* 167, 161–170. doi:10.1016/j.chemolab.2017.05.019
- Srinivasan, R., and Qian, M. (2007). Online Temporal Signal Comparison Using Singular Points Augmented Time Warping. *Ind. Eng. Chem. Res.* 46, 4531–4548. doi:10.1021/ie060111s
- Srinivasan, R., and Qian, M. S. (2005). Off-Line Temporal Signal Comparison Using Singular Points Augmented Time Warping. *Ind. Eng. Chem. Res.* 44, 4697–4716. doi:10.1021/ie049528t
- Srinivasan, R., and Sheng Qian, M. (2006). Online Fault Diagnosis and State Identification during Process Transitions Using Dynamic Locus Analysis. *Chem. Eng. Sci.* 61, 6109–6132. doi:10.1016/j.ces.2006.05.037
- Steinwandter, V., Borchert, D., and Herwig, C. (2019). Data Science Tools and Applications on the Way to Pharma 4.0. *Drug Discov. Today* 24, 1795–1805. doi:10.1016/j.drudis.2019.06.005
- Steinwandter, V., Zabel, T., Sagmeister, P., and Herwig, C. (2017). Propagation of Measurement Accuracy to Biomass Soft-Sensor Estimation and Control Quality. *Anal. Bioanal. Chem.* 409, 693–706. doi:10.1007/s00216-016-9711-9
- Stork, C. L., and Kowalski, B. R. (1999). Distinguishing Between Process Upsets and Sensor Malfunctions Using Sensor Redundancy. *Chemom. Intell. Lab. Syst.* 46, 117–131. doi:10.1016/s0169-7439(98)00180-4
- Tahir, F., Islam, M. T., Mack, J., Robertson, J., and Lovett, D. (2019). Process Monitoring and Fault Detection on a Hot-Melt Extrusion Process Using In-Line Raman Spectroscopy and a Hybrid Soft Sensor. *Comput. Chem. Eng.* 125, 400–414. doi:10.1016/j.compchemeng.2019.03.019
- Tórres, A. R., de Oliveira, A. D. P., Grangeiro, S., and Fragoso, W. D. (2018). Multivariate Statistical Process Control in Annual Pharmaceutical Product Review. *J. Process Control.* 69, 97–102. doi:10.1016/j.jprocont.2018.06.001
- Ündey, C., and Çınar, A. (2002). Statistical Monitoring of Multistage, Multiphase Batch Processes. *IEEE Control. Syst.* 22, 40–52. doi:10.1109/MCS.2002.1035216
- Ündey, C., Ertunç, S., and Çınar, A. (2003). Online Batch/Fed-Batch Process Performance Monitoring, Quality Prediction, and Variable-Contribution Analysis for Diagnosis. *Ind. Eng. Chem. Res.* 42, 4645–4658. doi:10.1021/ie0208218
- Ündey, C., Williams, B. A., and Çınar, A. (2002). Monitoring of Batch Pharmaceutical Fermentations: Data Synchronization, Landmark Alignment, and Real-Time Monitoring. *IFAC Proc. Volumes* 35, 271–276. doi:10.3182/20020721-6-es-1901.01354
- van den Kerkhof, P., Vanlaer, J., Gins, G., and van Impe, J. F. M. (2013). Analysis of Smearing-Out in Contribution Plot Based Fault Isolation for Statistical Process Control. *Chem. Eng. Sci.* 104, 285–293. doi:10.1016/j.ces.2013.08.007
- Veichtlbauer, A., Ortmyer, M., and Heistracher, T. (2017). “OPC UA Integration for Field Devices,” in 2017 IEEE 15th International Conference on Industrial Informatics (INDIN): University of Applied Science Emden/Leer, Emden, Germany, 24–26 July 2017 (Piscataway, NJ: IEEE), 419–424.
- Venkatasubramanian, V., Rengaswamy, R., and Kavuri, S. N. (2003a). A Review of Process Fault Detection and Diagnosis. *Comput. Chem. Eng.* 27, 313–326. doi:10.1016/s0098-1354(02)00161-8
- Venkatasubramanian, V., Rengaswamy, R., Kavuri, S. N., and Yin, K. (2003b). A Review of Process Fault Detection and Diagnosis. *Comput. Chem. Eng.* 27, 327–346. doi:10.1016/s0098-1354(02)00162-x
- Venkatasubramanian, V., Rengaswamy, R., Yin, K., and Kavuri, S. N. (2003c). A Review of Process Fault Detection and Diagnosis. *Comput. Chem. Eng.* 27, 293–311. doi:10.1016/s0098-1354(02)00160-6
- von Stosch, M., Davy, S., Francois, K., Galvanuskas, V., Hamelink, J. M., Luebbert, A., et al. (2014). Hybrid Modeling for Quality by Design and PAT-Benefits and Challenges of Applications in Biopharmaceutical Industry. *Biotechnol. J.* 9, 719–726. doi:10.1002/biot.201300385
- Voss, J.-P., Mittelheuser, N. E., Lemke, R., and Luttmann, R. (2017). Advanced Monitoring and Control of Pharmaceutical Production Processes with *Pichia pastoris* by Using Raman Spectroscopy and Multivariate Calibration Methods. *Eng. Life Sci.* 17, 1281–1294. doi:10.1002/elsc.201600229
- Walch, N., Scharl, T., Felföldi, E., Sauer, D. G., Melcher, M., Leisch, F., et al. (2019). Prediction of the Quantity and Purity of an Antibody Capture Process in Real Time. *Biotechnol. J.* 14, e1800521. doi:10.1002/biot.201800521
- Wasalathanthri, D. P., Feroz, H., Puri, N., Hung, J., Lane, G., Holstein, M., et al. (2020a). Real-time Monitoring of Quality Attributes by In-line Fourier Transform Infrared Spectroscopic Sensors at Ultrafiltration and Diafiltration of Bioprocess. *Biotechnol. Bioeng.* 117, 3766–3774. doi:10.1002/bit.27532
- Wasalathanthri, D. P., Rehmann, M. S., Song, Y., Gu, Y., Mi, L., Shao, C., et al. (2020b). Technology Outlook for Real-time Quality Attribute and Process Parameter Monitoring in Biopharmaceutical Development-A Review. *Biotechnol. Bioeng.* 117, 3182–3198. doi:10.1002/bit.27461

- Williams, B. A., Ündey, C., and Cinar, A. (2001). Detection of Process Landmarks Using Registration for On-Line Monitoring. *IFAC Proc. Volumes* 34, 221–226. doi:10.1016/s1474-6670(17)33827-2
- Wise, B. M., and Roginski, R. T. (2015). A Calibration Model Maintenance Roadmap. *IFAC-PapersOnLine* 48, 260–265. doi:10.1016/j.ifacol.2015.08.191
- Wu, H., Read, E., White, M., Chavez, B., Brorson, K., Agarabi, C., et al. (2015). Real Time Monitoring of Bioreactor mAb IgG3 Cell Culture Process Dynamics via Fourier Transform Infrared Spectroscopy: Implications for Enabling Cell Culture Process Analytical Technology. *Front. Chem. Sci. Eng.* 9, 386–406. doi:10.1007/s11705-015-1533-3
- Yao, Y., and Gao, F. (2009). A Survey on Multistage/Multiphase Statistical Modeling Methods for Batch Processes. *Annu. Rev. Control.* 33, 172–183. doi:10.1016/j.arcontrol.2009.08.001
- Yoo, C. K., and Lee, I.-B. (2006). Integrated Framework of Nonlinear Prediction and Process Monitoring for Complex Biological Processes. *Bioproc. Biosyst. Eng.* 29, 213–228. doi:10.1007/s00449-006-0063-2
- Yu, J., Chen, J., and Rashid, M. M. (2013). Multiway Independent Component Analysis Mixture Model and Mutual Information Based Fault Detection and Diagnosis Approach of Multiphase Batch Processes. *AIChE J.* 59, 2761–2779. doi:10.1002/aic.14051
- Yu, J., and Qin, S. J. (2009). Multiway Gaussian Mixture Model Based Multiphase Batch Process Monitoring. *Ind. Eng. Chem. Res.* 48, 8585–8594. doi:10.1021/ie900479g
- Yue, H. H., and Qin, S. J. (2001). Reconstruction-based Fault Identification Using a Combined index. *Ind. Eng. Chem. Res.* 40, 4403–4414. doi:10.1021/ie000141+
- Zarei, J., and Shokri, E. (2014). Robust Sensor Fault Detection Based on Nonlinear Unknown Input Observer. *Measurement* 48, 355–367. doi:10.1016/j.measurement.2013.11.015
- Zhang, A. H., Zhu, K. Y., Zhuang, X. Y., Liao, L. X., Huang, S. Y., Yao, C. Y., et al. (2020). A Robust Soft Sensor to Monitor 1,3-Propanediol Fermentation Process by *Clostridium Butyricum* Based on Artificial Neural Network. *Biotechnol. Bioeng.* 117, 3345–3355. doi:10.1002/bit.27507
- Zhang, S., Zhao, C., and Gao, F. (2018). Two-directional Concurrent Strategy of Mode Identification and Sequential Phase Division for Multimode and Multiphase Batch Process Monitoring with Uneven Lengths. *Chem. Eng. Sci.* 178, 104–117. doi:10.1016/j.ces.2017.12.025
- Zhang, Y., Lu, B., and Edgar, T. F. (2013). Batch Trajectory Synchronization with Robust Derivative Dynamic Time Warping. *Ind. Eng. Chem. Res.* 52, 12319–12328. doi:10.1021/ie303310c
- Zheng, J., and Song, Z. (2018). Semisupervised Learning for Probabilistic Partial Least Squares Regression Model and Soft Sensor Application. *J. Process Control.* 64, 123–131. doi:10.1016/j.jprocont.2018.01.008
- Zhu, P., Liu, X., Wang, Y., and Yang, X. (2018). Mixture Semisupervised Bayesian Principal Component Regression for Soft Sensor Modeling. *IEEE Access* 6, 40909–40919. doi:10.1109/access.2018.2859366

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2021 Brunner, Siegl, Geier and Becker. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

GLOSSARY

- b*** model coefficients
- E*** process end
- e*** residual between prediction and original sensor reading
- h*** warping function
- i*** index for sensor of interest
- j*** number of process variables
- L*** landmark
- m . . . n*** indices for sensors other than sensor *i*
- r_i*** reliability of sensor *i*
- SPE*** squared prediction error (sometimes denoted as *Q*)
- t*** time
- X*** process data = input to soft sensor
- X_{hist}*** historical process data
- X_{on}*** online process data
- X_{on,FT}*** fault-tolerant online process data
- X_{sync}*** synchronized process data
- y* or *Y*** target quantity (vector or matrix) = output of soft sensor
- y_{on}*** online data of target quantity
- y_{hist}*** historical data of target quantity
- \hat{y}_{hist}** prediction for historical data of target quantity
- \hat{y}_{on} or \hat{Y}_{on}** prediction for target quantity (vector or matrix)
- $\hat{y}_{on,FT}$** fault-tolerant prediction for target quantity
- φ** combined fault detection index
- AANN** autoassociative neural network
- ANN** artificial neural network
- COW** correlation optimized warping
- CPP** critical process parameter
- CQA** critical quality attribute
- CSTR** continuous stirred-tank reactor
- (D/M)DTW** (derivative/multivariate) dynamic time warping
- DSP** downstream processing
- EC** European Commission
- FCM** fuzzy *c*-means
- FDA** Food and Drug Administration
- GMM** Gaussian mixture model
- ISPE** International Society for Pharmaceutical Engineering
- JIT** just-in-time
- MLR** multiple linear regression
- MSPC** multivariate statistical process control
- OECD** Organisation for Economic Co-operation and Development
- OPC (UA)** open platform communications (unified architecture)
- PSO** particle swarm optimization
- RBC** reconstruction-based contributions
- RGTW** relaxed-greedy time warping
- PAT** process analytical technology
- PCA** principal component analysis
- PCR** principal component regression
- PLS(R)** partial least squares (regression)
- QbD** quality by design
- SCADA** supervisory control and data acquisition
- SVR** support vector regression
- USP** upstream processing
- VIP** variable importance in the projection
- XTW** extrapolative time warping