Computers and Chemical Engineering 128 (2019) 117–127

Contents lists available at ScienceDirect

Computers and Chemical Engineering

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

Campaign-based modeling for degradation evolution in batch processes using a multiway partial least squares approach

Ouyang Wu[a,b], Ala Bouaswaig[a], Lars Imsland[b,*], Stefan Marco Schneider[a], Matthias Roth[a], Fernando Moreno Leira[a]

[a] Automation Technology, BASF SE, 67056 Ludwigshafen, Germany
[b] Department of Engineering Cybernetics, NTNU, 7491, Trondheim, Norway

ARTICLE INFO

Article history:
Received 31 January 2019
Revised 21 May 2019
Accepted 27 May 2019
Available online 28 May 2019

Keywords:
Degradation evolution
Multiway PLS
Batch process
Campaign

ABSTRACT

In the process industry, various types of degradation occur in processing plants, resulting in significant economic losses. Modeling of degradation is important because it provides quantitative insights for consideration of degradation impacts in the operations of process manufacturing. This paper studies batch processes that show a periodic pattern for the evolution of degradation. A new data structure, the campaign, is applied for data-driven modeling of the periodic batch-to-batch evolution of degradation using a new multiway partial least squares approach, and it is further employed to predict the evolution of degradation in a series of batch runs. The proposed approach is illustrated and applied in a comprehensive industrial case study. The example illustrates the efficacy of the proposed model and presents a fair potential for applications of degradation prediction.

© 2019 The Authors. Published by Elsevier Ltd.
This is an open access article under the CC BY-NC-ND license.
(http://creativecommons.org/licenses/by-nc-nd/4.0/)

1. Introduction

In process manufacturing, degradation refers to detrimental changes of physical conditions in processes that evolve with time, use, or external causes, and has always been one of the main concerns in processing plants (BSI, 2001). Fouling as typical degradation occurs in many production scenarios and results in degraded efficiency, widely in units such as chemical reactors, coal-fired utility boilers, compressors, heat exchangers, etc. (Yeap et al., 2004; Teruel et al., 2005; Cicciotti et al., 2014; Wang et al., 2015; Urrutia, 2016). Other types of degradation in mechanical systems, such as cracking, deformation, and wear, are discussed and illustrated with examples in Martin et al. (1983), Zmitrowicz (2006) and Ciang et al. (2008). For the existing plants that have degradation issues, many solutions have been developed to minimize the negative impact of degradation on production from the perspectives of process control. Among them, various types of degradation models are developed for different types of degradation processes (Yeap et al., 2004; Teruel et al., 2005; Cicciotti et al., 2014; Wang et al., 2015; Wu et al., 2018). Further, control and optimization frameworks that incorporate degradation models are developed and applied in a variety of processes (Radhakrishnan et al., 2007; Xenos et al., 2016; Biondi et al., 2017; Lozano Santamaria and Macchietto, 2018; Dalle Ave et al., 2019; Wu et al., 2019a).

In the field of process control, one of the essential parts of handling degradation issues is modeling of degradation-related mechanisms. Modeling of degradation is usually custom-made. In some scenarios, the difficulties in the online measurement of degree of degradation lead to the use of monitoring and state-estimation approaches for detection or indication of degradation (Cicciotti et al., 2014; Wu et al., 2019b). When a degradation indication is available, it is possible to build a degradation model where the degree of degradation is a function of some influencing factors (Gorjian et al., 2010). This type of factors-based model can simulate the evolution of degradation in different time scales. Some factor-based fouling models have been developed for continuous processes in Teruel et al. (2005) and Radhakrishnan et al. (2007), which can be taken as maintenance tools to show the evolution of fouling in the next months by giving the future operating and scheduling parameters. The degradation model in Zhang et al. (1999) is used for the estimation of fouling in batch reactors, which focuses on the fouling evolution in each single batch run and, therefore, cannot predict fouling in future batches.

Degradation models focus on systems' patterns and dynamics that relate to degradation and, therefore, have been considered as a special type of soft sensors in industrial practice. Like other

* Corresponding author.
E-mail addresses: lars.imsland@ntnu.no, lars.imsland@itk.ntnu.no (L. Imsland).

https://doi.org/10.1016/j.compchemeng.2019.05.038
0098-1354/© 2019 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license. (http://creativecommons.org/licenses/by-nc-nd/4.0/)

soft-sensor applications, modeling of degradation must adapt to various types of processes and process industry data that are characterized by missing values, data outliers, data collinearity, etc (Kadlec et al., 2009; Yin et al., 2015). Degradation models can be classified into first-principle models and data-driven models. The approaches using first-principle models account for the physical and chemical mechanisms of the process, requiring adequate process knowledge. In many cases, first-principle models describing process dynamics are not reliable in practical scenarios due to a lack of complete process understanding and unknown process parameters. In comparison with first-principle approaches, data-driven approaches require less understanding of the process and aim to describe the real process conditions from historical data. These approaches show good performance in many scenarios of applications, and different data-driven methods for soft sensing, such as principal components analysis (PCA), partial least squares (PLS), and artificial neural networks, are reviewed in Kadlec et al. (2009) and Yin et al. (2015). Among those approaches, PCA and its regression model, PLS, or projection to latent structures described in Geladi and Kowalski (1986)and Abdi (2010) are widely applied to the modeling of high-dimensional industrial data. Furthermore, degradation models have been applied with an increasing interest in reliability analysis to evaluate and predict reliability of systems such as time-to-failure and the probability of failures to occur (Gorjian et al., 2010). Classification of degradation models in reliability analysis can be found in Gorjian et al. (2010), where the models are often application-specific using different types of modeling methods.

Batch processes are commonplace in the process industry. These are processes that produce a finite amount of output materials over a finite time period, and is typically repeated. These traits differentiate them from continuous processes and allow operations to adjust recipes and raw materials in each batch run. In many industrial batch processes, the lack of understanding of nonlinear dynamics and limited specific measurements leads to difficulties for building first-principle models and any use of conventional estimation techniques in the modeling of degradation (Zhang et al., 1999). Furthermore, the discrete operation aspect of batch processes gives a three-way matrix form of data that does not fit into modeling methods based on two-dimensional data. To apply multivariate analysis on batch processes, unfolding techniques are developed to unfold the three-dimensional matrix into a two-dimensional matrix leading to multiway PLS and multiway PCA approaches, which are widely used in monitoring and modeling of batch processes (Nomikos and MacGregor, 1994; 1995; Flores-Cerrillo and MacGregor, 2004; Wold et al., 2010; Keivan Rahimi-Adli, 2016). Moreover, further topics in the multiway PLS approaches, such as missing data estimation, multimode, and robust model, are discussed to improve the performance of batch-wise modeling (Nelson et al., 1996; Arteaga and Ferrer, 2002; Nelson et al., 2006; Wang and Srinivasan, 2009; Wang, 2011). On the other hand, the long-term trend of batch-to-batch data draws attention and has been considered for application-specific modeling in literature. Considering batch-to-batch startups that shows nonstationary and nonidentically distributed process data from batch to batch, Yan et al. (2015) adopted a nonparametric signal decomposition techniques to obtain the short-term intrabatch variations for multivariate statistical process monitoring. Wu et al. (2018) developed a degradation model to capture the long-term trend of fouling between batches.

This paper focuses on the modeling of degradation evolution in batch processes, which provides prediction of batch-to-batch degradation. In a degraded batch system, the condition of degradation is a trigger for the maintenance after a certain number of batch runs. A degradation model can simulate the evolution of degradation in different conditions of batch production and, therefore, provides new possibilities for optimization of batch production, such as scheduling of multiproduction and maintenance operations. It is observed that some batch-to-batch degradation follows a periodic pattern growing from zero degradation to high degradation repeatedly, and the concept of campaign is introduced to denote a sequence of batches following such a periodic pattern (Wu et al., 2018). A novel multiway PLS approach is proposed to model the evolution of degradation by considering variations from campaign to campaign, which presents good abilities for the prediction of degradation in future batches. The main contribution in this paper is the proposal of a new model formulation for the prediction of batch-to-batch degradation. We adapt the multiway approach used typically for batch analysis to the modeling of campaign-structured data. We extend our previous work in Wu et al. (2018) in several ways, most importantly by introducing the concept of subcampaigns to improve data alignment. Moreover, an industrial case study, significantly extended from Wu et al. (2018), is illustrating the application of the new campaign-based PLS model to predict the degradation in the form of a fouling indicator.

The paper is structured as follows. Section 2 revisits the batch data structure and the multiway approach and introduces the concept of the campaign for batch processes and its data structure. New data alignment and unfolding methods are introduced to adjust the campaign-structured data and fit it into a multivariate input–output model. Subsequently, a general campaign-based PLS model structure is presented along with a prediction framework based on missing data estimation methods. Section 3 presents a case study about fouling in a chemical batch plant in which the proposed approach is applied to model and predict the evolution of fouling. Moreover, discussions about the application in the fouling example are provided. Finally, conclusions are drawn in Section 4.

2. Methodology

2.1. Revisit of multiway approaches for batch data analysis

Multiway approaches including multiway PLS and multiway PCA are extensions of standard multivariate data analysis methods to handle batch data that are in a three-dimensional structure (Nomikos and MacGregor, 1994; 1995). To illustrate, consider all available batch data from Batch 1 to Batch Q, which consists of batch initial conditions, process trajectories, and batch final qualities, as illustrated in Fig. 1. The initial condition matrix ($Q \times I$) refers to the measurements (total number I) related to recipe-based initialization of each batch run, such as initial temperatures, raw material charges and properties, etc. The process trajectories are several measurements (total number of types J) such as temperatures, pressures, flowrates, etc., being made frequently throughout the duration (T_q) of each batch run, where the subscript $q \in \{1, 2, \ldots, Q\}$ denotes the batch index, leading to a three-dimensional structure ($Q \times J \times T_q$). The batch final quality matrix ($Q \times F$) refers to critical product specifications (total number F) in each batch run. Moreover, the final quality is usually affected by the batch operation which are shown in the form of process trajectories and initial conditions. For the three-dimensional data arising from batch processes, multiway approaches typically apply an unfolding method first to transform trajectories into a two-dimensional matrix. As an example, the unfolding method called batchwise unfolding (BWU) rearranges the presentation of trajectories from batch to batch, in which each of the vertical slices ($Q \times J$) of trajectory data is put side by side to the right to create a new two-dimensional structure as shown in Fig. 1 (a) (Nomikos and MacGregor, 1994; 1995). The initial conditions and the unfolded trajectories are, in the next step, taken into the input matrix of

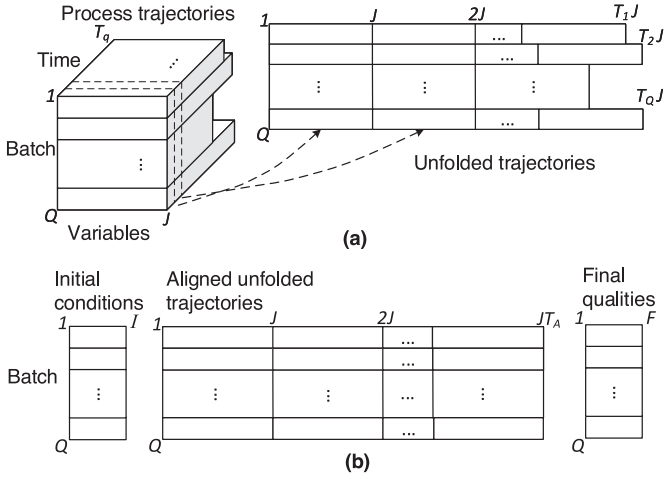

Fig. 1. (a) Batch data structure and batchwise unfolding; (b) alignment of batch data.

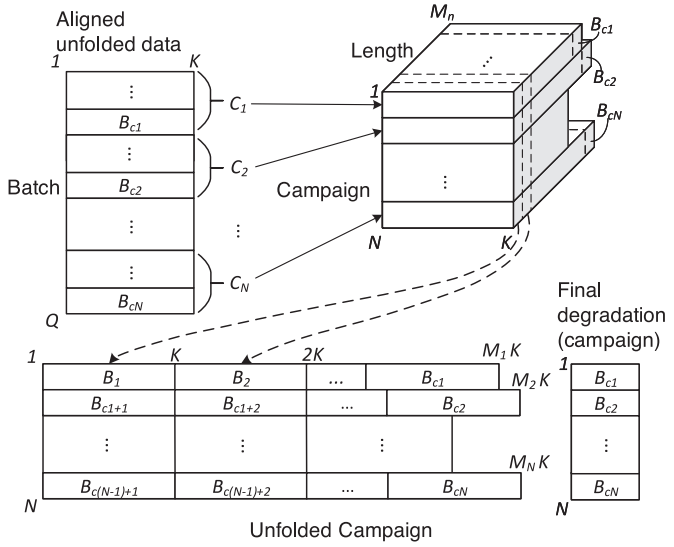

Fig. 2. Structure of campaign data and campaignwise unfolding; B_i is an indicator saying the corresponding data is from Batch i.

both PCA and PLS models. Additionally, PLS models consider an output matrix that includes batch final qualities. These empirical models can be of use to applications of batch monitoring and final product quality prediction, and the online prediction can be further employed for the midcourse correction of ongoing batches (Yabuki and MacGregor, 1997; Flores-Cerrillo and MacGregor, 2002; Wold et al., 2010; Keivan Rahimi-Adli, 2016).

One of the challenges in applying multiway approaches in batch data analysis is the variation of batch duration. The number of trajectory measurements of each batch varies according to batch duration T_q, leading to uneven row lengths in the input matrix according to the unfolding method, as presented in Fig. 1 (a). Typically, changes in operating conditions result in varying batch duration. Data alignment becomes necessary for the modeling of batch data so that variables or scores at any point during a batch can correspond to those at the same state in other batches. A widely used approach for data alignment is to use the time or find a signal that increases monotonically from a given starting point to an end, which is taken as a maturity indicator (Wold et al., 2010; Keivan Rahimi-Adli, 2016). By aligning the unfolded data according to the maturity indicator with the number of new samples in each batch as T_A, the row length for each unfolded trajectories is kept even as Fig. 1 (b) shows. One can also use landmarks for each batch, which describe representative features of trajectory data such as minimum and maximum values, average numbers of process measurements over a period (Wold et al., 2010).

2.2. Campaign structure for degradation evolution

We introduce the notion of a *campaign* to denote a series of batches that repeats in a periodic pattern. In this paper, campaigns are batch series between any two neighboring maintenance actions that restore a degradation mechanism. Usually, the growth rate of degradation in the batch production is not so fast that maintenance has to be done after each batch. Instead, the maintenance is carried out after a series of batches when the accumulated degradation reaches a threshold. The maintenance restores the degradation condition in the batch unit, leading to a periodic pattern where the degree of degradation grows from zero to a high level repeatedly. The evolution of degradation in each campaign shares many similarities with the within-batch dynamics of batch processes. First, both campaigns and batches are processed in a repeated pattern. Each batch run in a normal operation mode follows a certain batch trajectory, which is in a time series, and ends up with a product that meets the quality requirements. Compared to that, each campaign from a multipurpose batch plant consists of a series of batches following a production sequence in each batch unit, and those batches contribute to the growth of degradation until a maintenance operation restores the degree of degradation. Second, both campaigns and batches share similar variations on the length of the series. The duration of each batch varies due to the change of operating conditions, while the length of the batch series in each campaign changes due to different growth rates of degradation.

As stated above, we use the campaign concept to group batch series between two neighboring maintenance operations so that the grouped batch series present a qualitatively similar degradation evolution in each campaign. By rearranging the unfolded and aligned batch data, we propose a campaign data structure that piles the two-dimensional unfolded batch data according to the grouped batch series as illustrated in Fig. 2. B_q indicates the corresponding data are from Batch q, while the subscript cn in B_{cn} denotes the index of the last batch in Campaign n. The aligned and unfolded data are from Fig. 1 (b) with a new total number K. C_n refers to Campaign $n \in \{1, 2, \ldots, N\}$, where N is the total number of the campaigns. M_n is the number of batches in Campaign n. Corresponding to the final quality in batch data, the final degradation of a campaign is defined as the degree of the degradation in the last batch of the campaign that can be measured or indicated. The number of degradation indicators or measurements for each batch is one and can be extended with multiple ones in specific cases. The degree of degradation is often affected by other earlier batch operations within the same campaign due to the evolution mechanism.

2.3. Multiway approach for campaign-based degradation modeling

Because frequent degradation measurements or indicators are typically not available, modeling the batch-to-batch evolution of degradation becomes a favorable option for the purpose of degradation prediction and is the one chosen here. This type of degradation model considers the contribution of each batch to the overall evolution of degradation instead of the continuous-time within-batch dynamics of degradation in each individual batch. The proposed campaign data structure is developed based on the feature of degradation evolution, the repeated pattern from campaign to campaign, and leads to a new modeling of degradation

by integrating the structure of the campaigns. In such a campaign-based degradation model, the campaign data consist of a series of individual batch data. Because the batch data have shown a three-dimensional structure with batch indexes, process variables, and time instants, the campaign data are also multidimensional and have an additional dimension, campaign indexes. Similarly, in regard to the case of batch data, the campaign data have to be unfolded into a two-dimensional structure for multivariate data analysis. An unfolding method called campaignwise unfolding (CWU) is proposed in Wu et al. (2018). With the application of the unfolding method and data alignment on batch data, CWU considers the three-dimensional campaign data based on results of batchwise unfolded data, as Fig. 2 shows, and the unfolded batch data of each campaign is further unfolded into one line by putting the vertical slices ($N \times K$) side by side to the right. Similar to the batchwise unfolding method that enables further modeling and analysis among batches, the campaignwise unfolding method provides the appropriate data structure to model the variations of "trajectories" from campaign to campaign with regard to the evolution of degradation. In such a model, the final degradation for campaigns is taken as the output. The unfolded batch data reflecting batch operations of each campaign are taken as the inputs. Furthermore, the unfolding method can generate many columns in the input matrix of campaign degradation models. Though PLS methods may be able to handle this, it is often not necessary to choose all unfolded data from the same campaign as inputs; for example, some time-dependent measurements (batch trajectories) repeat patterns from batch to batch and are irrelevant to the evolution of degradation.

The number of batches in each batch series of a campaign can be different, and, therefore, the CWU-based unfolded campaign data in Fig. 2 have an uneven row length in the two-dimensional structure, resulting in difficulties for further modeling. Unlike continuous time series in batch data, it is difficult to merge discrete batch samples or decompose one batch sample into several ones for data alignment due to the discrete nature of the batch series. Furthermore, the batch series contributes to the degradation evolution in the form of production sequences, and it does not make sense to remove any batch samples in the middle of the sequences. Wu et al. (2018) proposed to use the last M batches of each campaign for degradation modeling, and, therefore, the model takes the same number of batch samples for each campaign requiring no further alignment. However, this method only enables part of the campaign features to be captured in the model and is hard to determine the starting batch of a new campaign for prediction. A new data alignment method is required to improve the campaignwise modeling of degradation.

A new data resampling method is proposed here for the alignment of campaign data by generating subcampaigns from campaigns. A subcampaign is a fixed-length batch series belonging to a single campaign and is resampled by shifting a M-length window within each campaign batch series, as Fig. 3 shows. Take Campaign C_1 as an example. Many subcampaigns are generated from C_1S_1 to C_1S_{c1-M+1}, where the subscripts in C_1S_1 denote the campaign index and the index of the corresponding subcampaign. The number of subcampaigns from one campaign depends on the lengths of the campaign and the subcampaigns. Some campaigns generate many subcampaigns, while others have less or even no subcampaigns. The length of a subcampaign, M, is application-related, and, normally, M should be small enough to generate a considerable number of subcampaigns from limited amounts of campaign data. Besides, M means the maximum prediction horizon in a subcampaign PLS model, which can be found in the next subsection, and more discussion about M can be also found in the case study section. In degradation models based on the subcampaign unfold data as Fig. 3 shows, the new final degradation is defined as the measurements or indicator of degradation after the last batch of the subcampaign and is taken as the output, which is different from the final degradation of campaigns. Similar to the batch modeling in Fig. 1, initial conditions of one subcampaign refer to degradation-relevant measurements before the starting batch of this subcampaign and are added as inputs in the degradation models. By modeling on the subcampaign data, the starting batch for the prediction of degradation evolution does not suffer the limitations observed by using the last M batches of campaigns for the data alignment; that is to say the first batch of one subcampaign can be assigned to the first batch or any middle one in an ongoing campaign. In addition, the subcampaign concept may help alleviate the "small-data problem[1]". Applying multivariate modeling to a small data set may lead to overfitting and lack of data representations, so the subcampaign resampling method generates larger data sets from a limited historical campaign data leading to an improved solution.

2.4. Degradation prediction using subcampaign PLS model

The PLS approach is employed to model degradation dynamics using the subcampaign data, as Fig. 3 illustrates. The model input consists of a series of unfolded subcampaign data and initial conditions for subcampaigns, which reflect the variations from subcampaign to subcampaign; and the indicator for the final degradation of subcampaigns is selected as the output. The model expression is presented as follows:

$$\begin{aligned} X &= TP^{\top} + E \\ Y &= TC^{\top} + F \end{aligned} \tag{1}$$

where the input matrix $X \in \mathbb{R}^{L\times(M_i+M_u)}$ is a collection of L subcampaign row vectors $x_l \in \mathbb{R}^{(M_i+M_u)\times 1}$:

$$X = \begin{bmatrix} \mathbf{x_1}^{\top} \\ \mathbf{x_2}^{\top} \\ \vdots \\ \mathbf{x_L}^{\top} \end{bmatrix} = [X_i, X_u], X_i = \begin{bmatrix} \mathbf{x_1^i}^{\top} \\ \mathbf{x_2^i}^{\top} \\ \vdots \\ \mathbf{x_L^i}^{\top} \end{bmatrix}, X_u = \begin{bmatrix} \mathbf{x_1^u}^{\top} \\ \mathbf{x_2^u}^{\top} \\ \vdots \\ \mathbf{x_L^u}^{\top} \end{bmatrix} \tag{2}$$

That is, X consists of the initial conditions and unfolded subcampaign data and is divided into blocks $X = [X_i, X_u]$, where X_i is a collection of the initial condition data of subcampaigns $\mathbf{x_l^i} \in \mathbb{R}^{M_i\times 1}$ and X_u is a collection of the unfolded data of subcampaigns $\mathbf{x_l^u} \in \mathbb{R}^{M_u\times 1}$. Moreover, $\mathbf{x_l^u}$ consists of M unfolded batch data $\mathbf{x_l^{u\top}} = [\mathbf{x_{(l,1)}^{ub\top}}, \mathbf{x_{(l,2)}^{ub\top}}, \ldots, \mathbf{x_{(l,M)}^{ub\top}}]$, and $\mathbf{x_{(l,m)}^{ub}}$ represents the vector of the mth unfolded batch data in Subcampaign l. The output matrix $Y \in \mathbb{R}^{L\times M_y}$ is a collection of final degradation indicator vectors $y_l \in \mathbb{R}^{M_y\times 1}$. The the score matrix $T \in \mathbb{R}^{L\times A}$ is a collection of vectors $\boldsymbol{\tau_l}$, and $\boldsymbol{\tau_l}$ corresponds to the values of the scores for the lth subcampaign. The matrices Y and T are presented as follows:

$$Y = \begin{bmatrix} y_1^{\top} \\ y_2^{\top} \\ \vdots \\ y_L^{\top} \end{bmatrix}, T = \begin{bmatrix} \boldsymbol{\tau_1}^{\top} \\ \boldsymbol{\tau_2}^{\top} \\ \vdots \\ \boldsymbol{\tau_L}^{\top} \end{bmatrix} \tag{3}$$

$P \in \mathbb{R}^{(M_i+M_u)\times A}$ and $C \in \mathbb{R}^{M_y\times A}$ are the loading matrices where the dimension of the PLS model is denoted by A. A second matrix $W \in \mathbb{R}^{(M_i+M_u)\times A}$, called âæweight matrix,âg helps calculate the scores T during the parameter estimation of the PLS model, and the score matrix T is defined by $T = XW_s$, where $W_s = W(P^{\top}W)^{-1}$ (Geladi and Kowalski, 1986). Both the input data X and the output data Y are preprocessed by centering and unit variance scaling, and the loading matrices {P, C, W} and score matrix {T} of the

[1] The small-data problem refers to the scenario that the product has a limited production history, which is common in process manufacturing (Tulsyan et al., 2018).

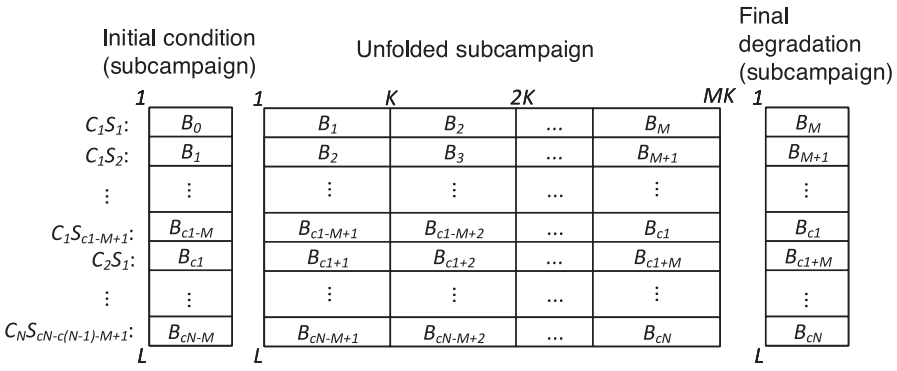

Fig. 3. Subcampaign unfolded data; B_i is an indicator saying the corresponding data is from Batch i.

PLS model are estimated using the nonlinear iterative partial least squares (NIPALS) algorithm (Geladi and Kowalski, 1986; Westerhuis et al., 1998). A multiblock PLS model as a variation of the standard PLS model is applied by dividing X into several blocks, such as X_i and X_u, and assigns weights to different input blocks for improving the interpretability of the model (Westerhuis et al., 1998). Given a new subcampaign data vector x_n, the corresponding output estimate $\hat{y}_n$ is calculated based on the existing PLS model as follows:

$$\begin{aligned} \tau_{\mathbf{n}} &= W_s^{\top}\mathbf{x}_{\mathbf{n}} \\ \hat{y}_n &= C\tau_{\mathbf{n}} \end{aligned} \tag{4}$$

where the score vector $\boldsymbol{\tau}_{\mathbf{n}}$ is calculated from $\mathbf{x}_{\mathbf{n}}$, and $\hat{y}_n$ is calculated from $\boldsymbol{\tau}_{\mathbf{n}}$ according to the PLS model in Eq. (1).

Degradation prediction using missing data estimation methods. In the scenario of degradation prediction, an ongoing campaign is provided with all measurements available for the finished batches, and the target is to predict the evolution of degradation in the planned future batches. The ongoing campaign is further divided into different subcampaigns according to Fig. 3, which contain different amounts of finished batches as Fig. 4 illustrates. The subcampaign PLS model obtained from the historical campaign data is employed to predict the final degradation for those subcampaigns. The measurements related to the future batches of each subcampaign lead to missing data in the input vector $\mathbf{x}$ for the subcampaign PLS model. The new input data vector $\mathbf{x}_{\mathbf{n}}$ is further divided into two parts $\mathbf{x}_{\mathbf{n}} = [\mathbf{x}_{\mathbf{n}}^{*\top}, \mathbf{x}_{\mathbf{n}}^{\#\top}]^{\top}$, the missing data $\mathbf{x}_{\mathbf{n}}^{\#}$, and the observed data $\mathbf{x}_{\mathbf{n}}^{*}$. The prediction of the final degradation of the subcampaign $\hat{y}_n$ is then calculated based only on $\mathbf{x}_{\mathbf{n}}^{*}$. In the literature, several missing data estimation methods, such as the conditional mean regression (CMR) and the trimmed score regression (TSR), have been developed to solve this problem (Nelson et al., 1996; Arteaga and Ferrer, 2002; Nelson et al., 2006). Nelson et al. (1996) proposed and implemented CMR for both PLS and PCA models. In Arteaga and Ferrer (2002), multiple missing data estimation methods are compared in PCA models, and it is concluded that both CMR and TSR show superior performance than others. Furthermore, TSR has the additional advantage of a much smaller size for the inverting of the covariance matrix.

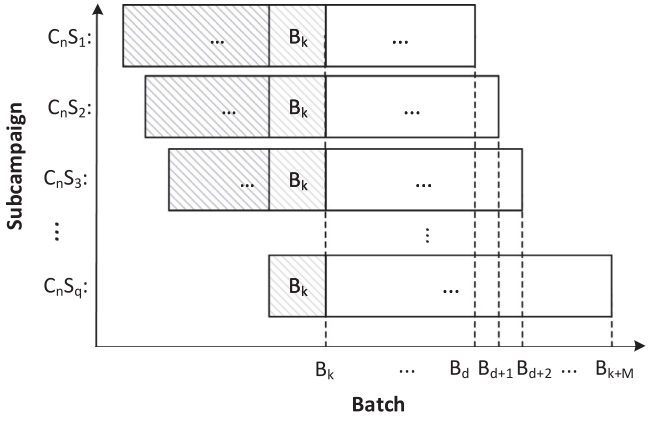

Fig. 4. Different subcampaigns at one given time; each horizontal rectangle represents a subcampaign, where B_k is the present batch, the shaded part represents finished batches, and the unshaded part represents future batches.

In the CMR method, first, the covariance matrix S of the input vector $\mathbf{x}_{\mathbf{n}}$ is calculated using the statistical covariance based on the historical normalized input data X as follows:

$$S = Cov(\mathbf{x}_{\mathbf{n}}) = X^{\top}X/(L-1) \tag{5}$$

The loading matrix W_s and the covariance matrix S are partitioned according to the input vector $\mathbf{x}_{\mathbf{n}}^{\top} = [\mathbf{x}_{\mathbf{n}}^{*\top}, \mathbf{x}_{\mathbf{n}}^{\#\top}]$:

$$W_s^{\top} = [W_s^{*\top},\ W_s^{\#\top}],\quad S = \begin{bmatrix} S_{**}, & S_{*\#} \\ S_{\#*}, & S_{\#\#} \end{bmatrix} \tag{6}$$

where S_{**} is defined as the covariance of the observed part $\mathbf{x}^*$, and other partitioned covariances are defined in a similar way. According to an assumption of normally distributed scores and Eq. (4), the conditional distribution of the score also follows a normal distribution (Nelson et al., 2006), given as:

$$\hat{\tau}_{\mathbf{n}} = E(\tau_{\mathbf{n}}|\mathbf{x}_{\mathbf{n}}^{*}, W_s, S) = W_s^{*\top}\mathbf{x}_{\mathbf{n}}^{*} + W_s^{\#\top}S_{\#*}S_{**}^{-1}\mathbf{x}_{\mathbf{n}}^{*} \tag{7}$$

On the other hand, the TSR method is derived by fitting a regression model between the trimmed score, $\tau^* = W_s^{*\top}\mathbf{x}^*$, and the score $\boldsymbol{\tau}$ in the historical data, and the estimate of the final score is calculated using the regression model as follows (Arteaga and Ferrer, 2002):

$$\begin{aligned} \hat{\tau}_{\mathbf{n}} &= \beta W_s^{*\top}\mathbf{x}_{\mathbf{n}}^{*} \\ \beta &= T^{\top}TW_s^{*\top}W_s^{*}(W_s^{*\top}S_{**}W_s^{*})^{-1} \end{aligned} \tag{8}$$

The estimates of both output y_n and missing input $\mathbf{x}_{\mathbf{n}}^{\#}$ are further calculated from the score estimate $\hat{\tau}_{\mathbf{n}}$ according to Eqs. (1) and (4).

In the next section, a case study about a chemical batch reactor is presented and the subcampaign approach is employed to predict the evolution fouling in the batch production.

3. Case study

A case study from a multipurpose chemical batch plant, which is introduced and described in Wu et al. (2018, 2019b), is considered in this paper. First, a short description of the batch process is provided, and the main degradation in this unit, fouling, along with its indication are described. A subcampaign PLS model is built for

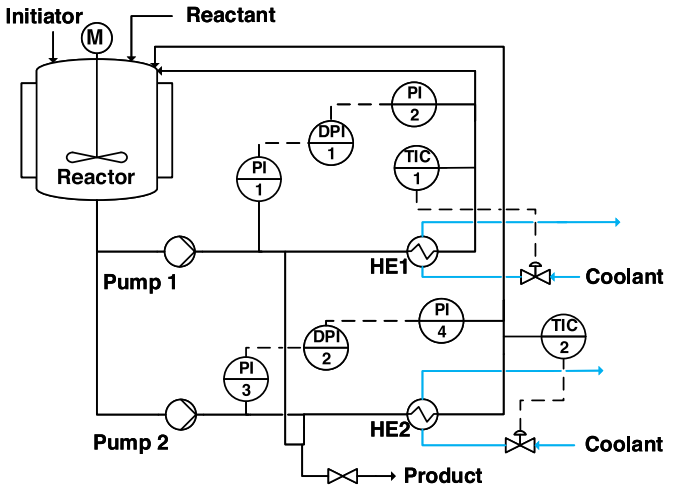

Fig. 5. Batch process schematics: reaction section.

the fouling evolution in the series of batch runs, and the results of the modeling, as well as the fouling prediction, are presented to show the efficacy of the proposed approach.

3.1. Process description

In the considered example, the batch plant produces multiple types of chemical products using different batch recipes. In each batch, raw materials are discharged and blended in a vessel according to the recipe, whereafter the blended materials are discharged into a batch reactor, as shown in Fig. 5, where the polymerization starts with the inflow of the initiator. Because of the exothermic reaction, the reactor is equipped with two parallel recirculation loops in which pumps and heat exchangers are employed to cool the reactor. Once the reaction is finished, the product in the reactor section is further transferred into storage tanks.

Fouling is one of the main factors that result in production degradation in this batch plant. During each batch, residuals of product materials are accumulated in the inner surface of the reactor, the pipes, and the heat exchangers. One of the main impacts of fouling is the reduced heat transfer from the product to the coolant, which results in a longer batch duration due to the decreased cooling capacity. Moreover, the flow resistance grows due to the residuals in the heat exchangers and the pipes, leading to an increase in the pressure drop over the heat exchanger, and too many residuals could even result in a flow block in the recirculation loop. As a result, the units in the reactor section need to be shut down and cleaned once the fouling reaches an unacceptable level, and this frequently repeated scheduling of shutdowns leads to capacity losses in the production.

The fouling exacerbates the bottleneck of the production and, therefore, needs to be considered when doing any scheduling for the batch process. In Wu et al. (2019a), fouling effects on batch duration are considered in short-term scheduling of batch production by integrating fouling models into the optimal scheduling framework. The fouling model, conceptually similar but simpler than the one developed herein, describes the fouling evolution from batch to batch and provides predictions of fouling evolution for the batch scheduling, and is, therefore, essential in the optimization of batch operations when considering fouling effects.

3.2. Fouling indication

Some sort of indicator for the fouling condition is a requisite for the modeling of fouling dynamics. The indicator in this example relies on the availability of specific process measurements and the corresponding inner mechanism. In Wu et al. (2018), a pressure-based key performance indicator (KPI) is adopted to indicate the degree of fouling in the reactor section at the beginning of each batch run. The KPI takes measurements of the pressure drops over the heat exchangers through the differential pressure (DP) sensors, as shown in Fig. 5, and reveals the variation of fouling according to its effects on the flow resistance. The wide range of the operating point brings disturbances on the use of pressure measurement for fouling indication in each batch. As a result, a frequent indication of fouling, which is based on the batch trajectories of the pressure variable, becomes difficult. This less frequent and batch-sampled fouling KPI prevents from the effects of wide operating ranges in batch processes but still presents irregular numbers due to changes in batch recipes. Some recipes have relatively higher or lower KPI values compared with the neighboring batches using other recipes. This is because different batch recipes take different raw materials or different charge amounts of the same raw materials, leading to varying physical properties for both the reactant and the product. The variations in batch recipes cause an inconsistent indication of fouling for each batch and, therefore, need to be considered in the refinement of the KPI, which is essential for further modeling of fouling. By formulating and solving a state-estimation problem regarding the fouling evolution, a recipe-independent fouling KPI is developed in Wu et al. (2019b). The corresponding results are further illustrated in the following paragraph.

The aforementioned fouling indication is applied in a set of historical data. The selected data are from a relatively short and concentrated period (one year) and have less abnormal changes or disturbances during the operations, which makes the system less time-varying and more consistent in regard to the evolution of fouling. Because only one of the two parallel recirculation loops was running during the picked period, the DP measurements of the corresponding heat exchanger are taken to calculate the fouling KPI for each batch run. The batch-to-batch series of the fouling KPI from the historical data is presented in Fig. 6 to show the evolution of fouling in different campaigns, where the red circles denoting the cleaning operation divide the fouling KPI series into many campaigns. The symbols, such as the star, cross, left-pointing triangle, and right-pointing triangle, are used to denote the recipe groups from $RG1$ to $RG4$ for the pressure measurements of batches, and the black dotted line denotes the recipe-independent KPI obtained from Wu et al. (2019b). The length of campaigns varies during the batch production. In practice, the scheduling of shutdowns for cleaning is not only determined by the fouling conditions. Some campaigns are rather short, with cleanings carried out in advance of what would be needed due to other operational issues apart from fouling. Those short campaigns are less representative for the modeling and were, therefore, omitted from the modeling data set.

3.3. Campaign data: data unfolding and alignment

The proposed campaign modeling approach is employed for the prediction of the fouling KPI in future batches of the example reactor. Because of the varying length of campaigns, data alignment is required for the modeling, and the length of subcampaigns M is determined according to the historical data: M needs be large enough to cover the main fouling evolution in the campaign, while it should be small enough to fit most of the campaigns. By setting a fixed length M, the subcampaign unfolding approach generates many subcampaigns from the campaigns.

According to the campaign structure, the historical subcampaign data consist of different layers of data, such as batch-to-batch, within-batch, and subcampaign. The within-batch layer includes trajectories of process variable measurements throughout the batch duration. Among them, measurements, such as

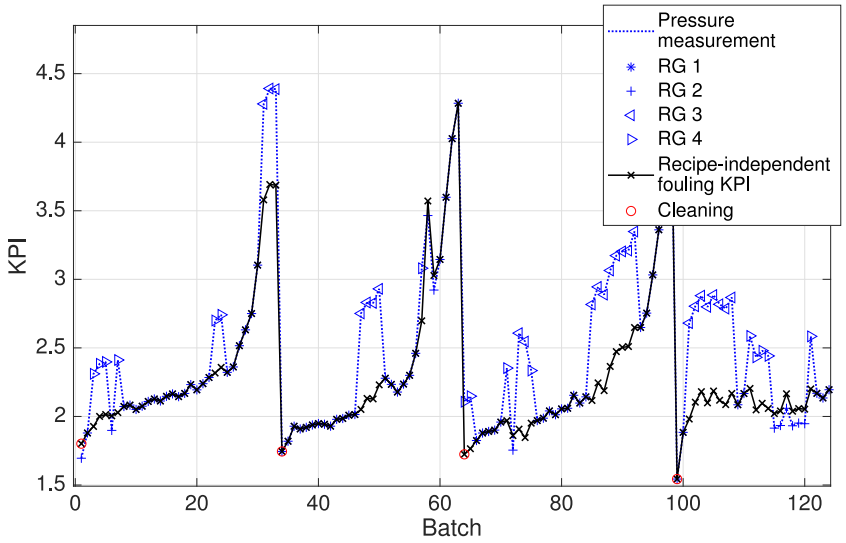

Fig. 6. Batch-to-batch fouling KPI series denoted with batch recipes and cleaning. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

temperatures and flow rates, are controlled to follow the designed batch trajectories and, therefore, are repeated from batch to batch. Pressure measurements, however, are fouling-related and receive a direct impact from the fouling dynamics. The batch-to-batch layer considers the specifications of each batch according to the recipes, such as the charged amounts of raw materials, the set points for the batch control loops, and the landmarks from the trajectories which are relevant to the growth of the fouling after one batch. The fouling KPI indicates the fouling condition at each batch, and the batch-to-batch series of the KPI follows a Markov chain process due to the fouling mechanism. The layer of subcampaign in this example considers the initial fouling KPI and the final fouling KPI of each subcampaign.

3.4. Subcampaign PLS model for fouling prediction

As the purpose of the modeling is to predict the fouling KPI of future batches, the final fouling KPI of subcampaigns, which is defined as the fouling KPI after the end batch of each subcampaign, is selected as the output of the subcampaign fouling model. The batch trajectory data is not included in the input matrix because it enlarges the size of the input matrix greatly and has small variations in the operation that hardly contribute to the output. The recipes during the batch production contribute to the fouling evolution and, therefore, are selected as parts of the inputs for this model. Different recipes share several similar raw materials, while several ingredients are specific to particular recipes. Besides, recipes vary in terms of the amounts of raw material additions. As such, the recipe data matrix that contains charged amounts of raw materials is employed, while the data for the zero charged amounts of particular raw materials in specific recipes is not clear. Lane et al. (2001) developed a data preprocessing method for the recipe data in multivariate modeling. This method treats the data with the zero charged amounts as zero values in the normalized data of corresponding raw materials, and the scaling and centering are employed for each ingredient data according to the types of recipes. Seven different types of raw materials are considered in this example, and the preprocessed recipe data are illustrated in Fig. 7, where the shape of normalized values from raw material *A* to *G* shows the differences between different recipes. The initial fouling KPI for subcampaigns affects the growth of fouling in each subcampaign and, as a result, is selected as one of the inputs. Furthermore, the fouling KPI value of the kth batch in subcampaigns denoted as $\{f_k|k = 1 : M\}$ are relevant to the output f_M, and, therefore, an extracted feature of the fouling KPI series, called D-KPI, is further employed for the subcampaign PLS model by computing the neighboring differences of the batch KPI series $df_k = f_k - f_{k-1}$. The D-KPI series provides the exact fouling growth of each batch in one campaign, which linearly contributes to the final fouling KPI of subcampaigns along with the initial KPI value f_0. To sum up, the input vector $\mathbf{x}$ of the M-length subcampaign model consists of the initial fouling KPI value f_0, the series of the recipe data vector $\{r_k|k = 1 : M\}$, and the series of the D-KPI value $\{df_k|k = 1 : M\}$.

3.5. Model validation

A latent variable software called ProMV (MacGregor et al., 2015; Aspentech, 2018), which is designed for multivariate modeling of continuous or batch processes, is applied to build the PLS model here. The historical data of seven campaigns are employed to validate the proposed method, and the first step is to determine the length of the subcampaign M. Given the average length of campaigns 34 and the shortest length 29, M is set to 25 in this subcampaign PLS model to generate many subcampaigns from a limited number of campaigns and meanwhile to include more batches in one subcampaign for a longer prediction horizon. Cross validation is a popular method giving guidelines for determining the number of components in a PLS model (Wold, 1978), and the leave-one-out method provides a general means for selecting validation sets. Since subcampaigns that belong to one campaign share some input data, it is reasonable to exclude all those subcampaigns from the set of subcampaigns for validation. This leave-one-campaign-out cross validation is applied and generates seven different validation and training sets. The result of the cumulative variance explained in the X and Y spaces with different numbers of principal components is presented in Table 1. The metric Q2Y is the variance explained in Y space accounting for all validation sets, which is used to evaluate the predictive performance to avoid overfitting. R2X and R2Y are the variances explained in the X and Y spaces for all training sets. The Q2Y value is around 69.15% when the number of components increases to eight and is not improving critically by adding more components, and, therefore, the number of the principal component for the PLS model is chosen as eight. The scatter plot of estimated and observed outputs using the

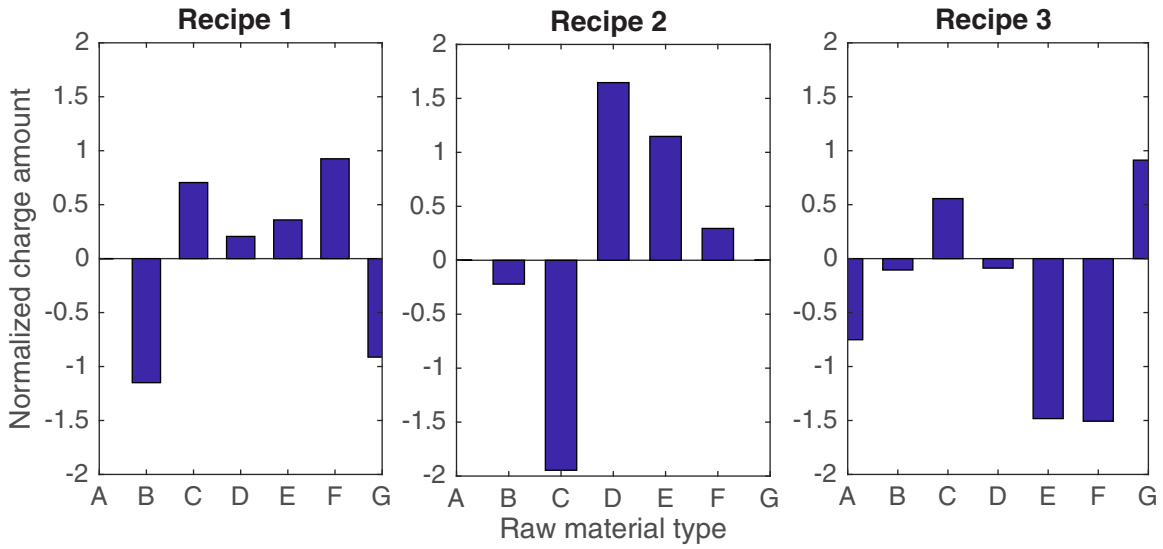

Fig. 7. Normalized charge amounts of raw materials for different recipes.

Table 1
cumulative variance explained in the X and Y spaces.

PC No.	1	2	3	4	5	6
R2Y	49.37%	86.13%	91.12%	94.54%	97.28%	98.36%
Q2Y	−12.55%	23.18%	43.11%	53.19%	57.67%	59.85%
R2X	14.34%	21.08%	32.03%	40.45%	44.82%	49.71%
PC No.	7	8	9	10	11	12
R2Y	99.34%	99.63%	99.76%	99.87%	99.92%	99.96%
Q2Y	65.53%	69.15%	70.17%	70.89%	72.31%	73.00%
R2X	52.53%	55.33%	58.94%	61.78%	63.86%	65.16%

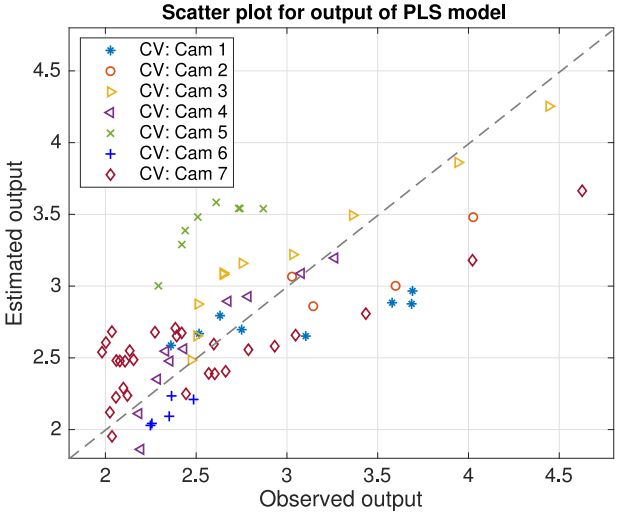

Fig. 8. Validation: output estimates and observations.

eight-component PLS model is presented in Fig. 8, where seven symbols represent the results of each validation data set and the notation *CV:Cam 1* denotes the validation data set consisting of the subcampaigns from Campaign 1. The statistical results using the root mean square error of estimation (RMSEE) are presented in Table 2. Among them, the estimated outputs in Cam 5 are relatively higher than the observed outputs, and some results from Cam 1 and Cam 6 are on the opposite. Cam 7 is special as it has more than 25 subcampaigns due to the long length, in which the

Table 2
Validation performance (RMSSE): 7 cross-validation sets.

CV set	Cam 1	Cam 2	Cam 3	Cam 4	Cam 5	Cam 6	Cam 7
RMSSE	0.50	0.43	0.28	0.16	0.85	0.23	0.42

corresponding training set is much smaller. The performance of Cam 7 is relatively not satisfactory due to lack of training data, while Cam 3 and Cam 4 show good fitting performance. In general, the performance of the fouling model can be improved with more training data.

In the prediction scenario, some parts of the input vector x are missing due to the unfinished future batches in the subcampaign. The recipe data are relatively fixed and can be obtained in advance according to the production schedule. It is always assumed that the initial fouling KPI is known so that the prediction horizon is bounded within M batches. The D-KPI data represent the operation status of each batch and, therefore, cannot be obtained if the batch is not finished, which leads to the missing data in this example. Two missing data estimation approaches are employed for the prediction of the final fouling KPI. Five prediction sets are provided with different portions of missing data in the input vector, and the prediction results are illustrated in Figs. 9 and 10 by taking Cam 3 as examples. The dashed line with the circular marker refers to the observed output in Campaign 3, and the dashed line with the star marker refers to the estimated output, as shown in Fig. 8. The right-pointing triangle marker denotes the prediction with 20 missing batches of the subcampaign, which means that the prediction is obtained without the last 20 batches and, therefore, provides a prediction horizon of 20 batches. This is similar for other predictions with different numbers of missing batches. The prediction with fewer numbers of missing input data or a shorter prediction horizon has a relatively smaller mismatch error compared with the output estimate, as the RMSSEs in Table 3 show. In addition, the prediction error compared with the observed output also becomes smaller when the prediction horizon becomes shorter. The prediction's RMSSE compared with the observed output is around twice as large as the RMSSE compared with the estimated output. That is to say, the mismatch error between the prediction and observed output comes from both the model estimate and the missing inputs. The results also show that the missing data

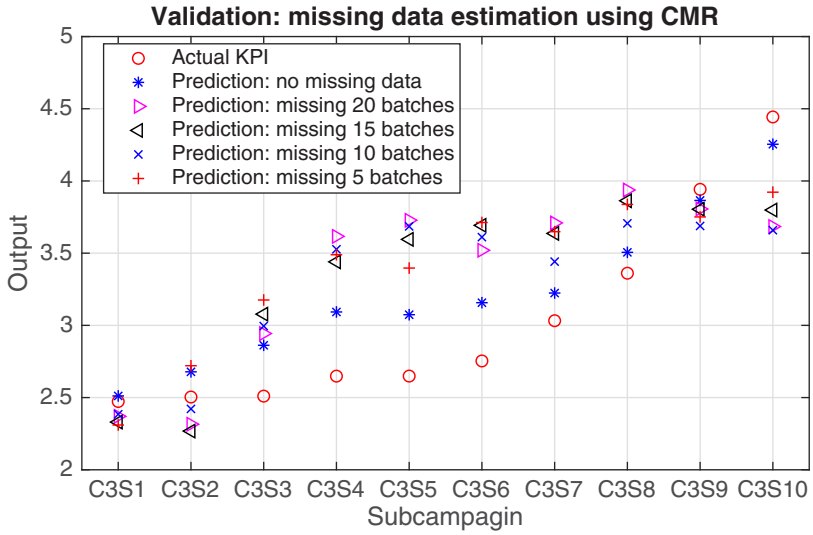

Fig. 9. Fouling KPI prediction using CMR method.

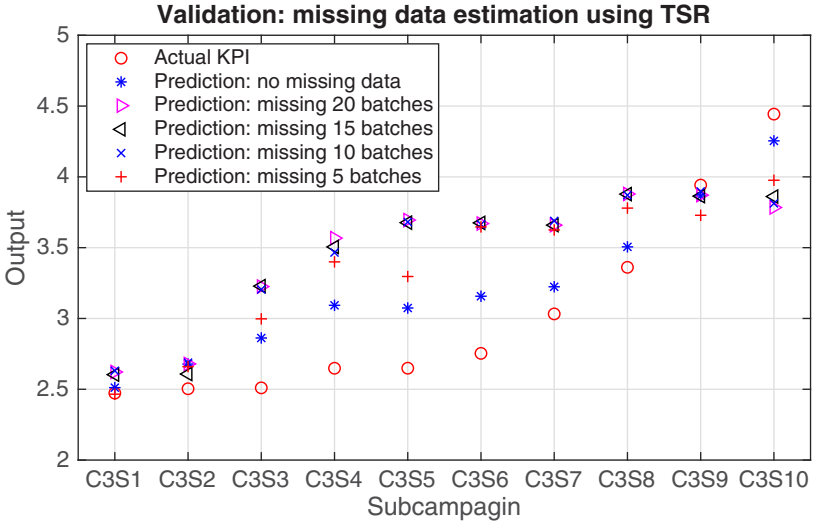

Fig. 10. Fouling KPI prediction using TSR method.

Table 3
Prediction performance (RMSSE) with missing data in the input vector.

RMSSE	Method	Prediction horizon			
		20 batches	15 batches	10 batches	5 batches
Estimated output	TSR	0.32	0.31	0.31	0.22
	CMR	0.37	0.34	0.34	0.28
Observed output	TSR	0.67	0.65	0.65	0.53
	CMR	0.66	0.63	0.62	0.60

estimation method TSR has slightly better performance than CMR by presenting smaller RMSSEs.

The estimated output from the subcampaign PLS model provides the prediction of fouling KPI after certain amount of future batches. One can also calculate the prediction of fouling KPI series within certain amount of future batches using estimates of missing input data. The fouling KPI at the kth batch of one subcampaign f_k can be calculated directly through the summation of the D-KPI data $\{df_j | j = 1 : k\}$ and the initial KPI of a subcampaign f_0, which renders a new perspective to predict the fouling KPI. The missing data estimation method helps to find the score estimate $\hat{\tau}$ from the observed part of the input vector, and the estimate of missing D-KPI $\hat{df}_k$ can be further calculated from the estimated score. As a result, the estimated KPI value $\hat{f}_k$ is calculated for all future batches within the prediction horizon M as Eq. (9) shows.

$$\hat{f}_k = f_0^i + \sum_{j=1}^{k} \hat{df}_j, \ \forall\ k = k_c : M \tag{9}$$

where k_c refers to the index of the ongoing batch in the subcampaign, and batches with an index larger than k_c refer to the future batches in this subcampaign. One subcampaign example, as presented in Fig. 11, shows the estimated KPI series $\{f_k | k = 1 : M\}$ of the subcampaign $C3S5$. The subcampaign starts from Batch 6 to Batch 30 in this campaign. The star marker refers to the estimated output calculated directly from the PLS model, which is close to the value f_{30} calculated from Eq. (9).

Some subcampaigns are resampled from the same campaign. These subcampaigns are different parts of the whole batch series of the campaign and share some batches, as Fig. 3 illustrates. The overlapped batches from different subcampaigns lead to multiple fouling KPI predictions for the same batches. One of the solutions to integrate multiple predictions is to give some weights for those predictions and to calculate the corresponding weighted average

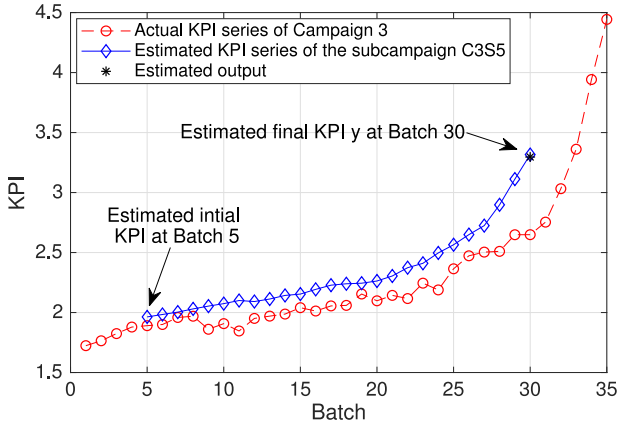

Fig. 11. Illustration of an estimated KPI series of Subcampaign C3S5, with initial KPI and final KPI, compared to the actual KPI series.

value. To avoid notation confusion, a new notation of KPI f_{Bi} is introduced, where the subscript i refers to Batch No.i in one campaign, which is calculated as Eq. (10):

$$\hat{f}_{Bi} = \sum_{j \in J_i} a_{ij} \hat{f}^j_{Bi}, \quad \forall\ i = i_c : N_c \tag{10}$$

where $\hat{f}^j_{Bi}$ denotes the KPI prediction of Batch No.i from the subcampaign j, which is calculated from Eq. (9); a_{ij} is the corresponding weight of f^j_{Bi} and $\sum_{j \in J_i} a_{ij} = 1$; the subcampaign j is in the set J_i and J_i refers to the set of subcampaigns that contain Batch No.i; notation i_c denotes the absolute index of the ongoing batch in the current campaign; and N_c is the last planned batch of this campaign. The weight a_{ij} denotes the reliability of the predictions from different subcampaigns, which can be assigned equally or based on some available prediction indicators such as Hotelling's T^2 and the squared prediction error. In this validation example, the weights are set equal for the predictions from different subcampaigns, and the average values are calculated for those predictions of the fouling KPI of the same batch in the validation set Cam 3. All available subcampaigns in Cam 3 are used to calculate a series of KPI predictions at different periods of this 'ongoing' campaign. The prediction results are illustrated in Fig. 12. Four figures in Fig. 12 provide the evolution of the predictions. Fig. 12 (a) shows three KPI prediction series at the early stage of this campaign, and Fig. 12 (d) shows the prediction series at the final stage of this campaign. A better prediction performance is observed for the prediction series that has more observed batch data.

3.6. Discussion

The proposed campaign-based approach provides a prediction framework to predict the evolution of fouling with flexible lengths of the horizon. The measurements for prediction are incomplete in an ongoing campaign. The missing data estimation method known as data imputation is applied to handle the missing future data by assuming a known correlation between the available data and the missing data, and the correlation in this example comes from the PLS model. The performance of the missing data estimates is highly affected by the regression model, the noise term, ill-conditioning in the inverting matrix, and the degree of information in the measured variables about the unmeasured variables, as Nelson et al. (1996) and Arteaga and Ferrer (2002) comments. In the case study example, the missing input data are D-KPI and show an acceptable correlation with other observed input data according to the PLS model, which helps with the missing data estimation. On the other hand, the small-data problem puts challenges on the multivariate modeling. In this example, the amount of industrial historical data is limited, and the number of observations has been increased leading to a sufficient PLS model. However, the

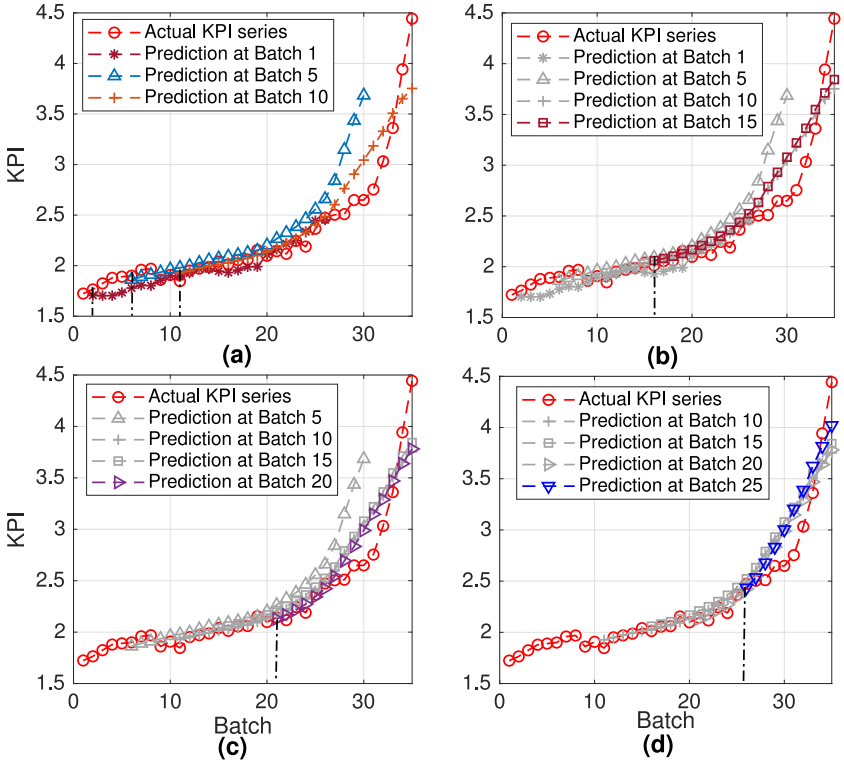

Fig. 12. Illustration of integrated prediction of KPI series for Campaign 3 compared to actual KPI series: (a) prediction made at Batch 1, 5 and 10; (b) prediction made at Batch 15; (c) prediction made at Batch 20; (d) prediction made at Batch 25.

lack of a sufficiently large amount of historical data presents difficulties in the missing estimation because of the high-dimensional regression model and leads to a less reliable estimate for the statistical input covariance in the CMR method. Moreover, both the PLS model and the missing data estimation work as data interpolation for those subcampaigns. The prediction performance becomes poor if the training data set does not cover the pattern in the validation set, which is a common challenge for all data-driven models.

Regarding to applications, maintenance operations are usually triggered when the degree of degradation reach a threshold, which links degradation models to the scheduling of maintenance. In the example, the fouling KPI has some thresholds for different recipes. The last batch of Campaign 3 is using Recipe *RG* 1 and has a threshold of fouling KPI as five. The corresponding predicted KPI from Figs. 10 and 12 reaches around four given the planned 15 future batches ahead, which potentially provide an early warning for the preparation of fouling cleaning. To apply the fouling predictive model to batch scheduling is left as future work.

4. Conclusions

A modeling formulation for the description of the evolution of degradation in batch processes has been presented. The concept of campaign is introduced to describe the periodic pattern for the degradation evolution in batch processes. A multiway approach for the campaign-structured data is developed by the proposal of corresponding data unfolding and alignment methods. The new multiway formulation combined with the mature multivariate modeling tools provides predictions of degradation in future batches, and it further presents a potential for practical applications by integrating it in the optimization of batch production and maintenance. A case study is presented to show an applicable industrial example for degradation dynamics in a chemical batch process. The proposed modeling approach is applied in the case study example and shows acceptable predictions for the fouling KPI of batch series.

Acknowledgement

Financial support is gratefully acknowledged from the Marie Skłodowska Curie Horizon 2020 EID-ITN project “PROcess NeTwork Optimization for efficient and sustainable operation of Europe’s process industries taking machinery condition and process performance into account PRONTO”, Grant agreement No 675215.

References

Abdi, H., 2010. Partial least squares regression and projection on latent structure regression (pls regression). Wiley Interdiscip. Rev. Comput. Stat. 2 (1), 97–106.
Arteaga, F., Ferrer, A., 2002. Dealing with missing data in mspc: several methods, different interpretations, some examples. J. Chemom. J. Chemom. Soc. 16 (8–10), 408–418.
Aspentech, 2018. Aspen promv brochure. URL: https://www:aspentech.com/en/resources/brochure/aspen-promv-brochure.
Biondi, M., Sand, G., Harjunkoski, I., 2017. Optimization of multipurpose process plant operations: a multi-time-scale maintenance and production scheduling approach. Comput. Chem. Eng. 99, 325–339.
BSI, 2001. Maintenance terminology. Standard. BS EN 13306
Ciang, C.C., Lee, J.-R., Bang, H.-J., 2008. Structural health monitoring for a wind turbine system: a review of damage detection methods. Meas. Sci. Technol. 19 (12), 122001.
Cicciotti, M., Xenos, D.P., Bouaswaig, A.E., Thornhill, N.F., Martinez-Botas, R.F., 2014. Online performance monitoring of industrial compressors using meanline modelling. In: Proceedings of the ASME Turbo Expo 2014: Turbine Technical Conference and Exposition. American Society of Mechanical Engineers.
Dalle Ave, G., Hernandez, J., Harjunkoski, I., Onofri, L., Engell, S., 2019. Demand side management scheduling formulation for a steel plant considering electrode degradation. In: Proceedings of the IFAC International Symposium on Dynamics and Control of Process Systems (DYCOPS).
Flores-Cerrillo, J., MacGregor, J.F., 2002. Control of particle size distributions in emulsion semibatch polymerization using mid-course correction policies. Ind. Eng. Chem. Res. 41 (7), 1805–1814.
Flores-Cerrillo, J., MacGregor, J.F., 2004. Multivariate monitoring of batch processes using batch-to-batch information. AIChE J. 50 (6), 1219–1228.
Geladi, P., Kowalski, B.R., 1986. Partial least-squares regression: a tutorial. Anal. Chim. Acta 185, 1–17.
Gorjian, N., Ma, L., Mittinty, M., Yarlagadda, P., Sun, Y., 2010. A review on degradation models in reliability analysis. In: Engineering Asset Lifecycle Management. Springer, pp. 369–384.
Kadlec, P., Gabrys, B., Strandt, S., 2009. Data-driven soft sensors in the process industry. Comput. Chem. Eng. 33 (4), 795–814.
Keivan Rahimi-Adli, 2016. Grey-box Modeling, State Estimation and Optimization of a Semi-batch Reactor. TU Dortmund University, Dortmund, Germany Master’s thesis.
Lane, S., Martin, E., Kooijmans, R., Morris, A., 2001. Performance monitoring of a multi-product semi-batch process. J. Process Control 11 (1), 1–11.
Lozano Santamaria, F., Macchietto, S., 2018. Integration of optimal cleaning scheduling and control of heat exchanger networks undergoing fouling: model and formulation. Ind. Eng. Chem. Res. 57 (38), 12842–12860.
MacGregor, J., Bruwer, M., Miletic, I., Cardin, M., Liu, Z., 2015. Latent variable models and big data in the process industries. IFAC-PapersOnLine 48 (8), 520–524.
Martin, P., Strutt, J., Kinkead, N., 1983. A review of mechanical reliability modelling in relation to failure mechanisms. Reliab. Eng. 6 (1), 13–42.
Nelson, P.R., MacGregor, J.F., Taylor, P.A., 2006. The impact of missing measurements on PCA and PLS prediction and monitoring applications. Chemom. Intell. Lab. Syst. 80 (1), 1–12.
Nelson, P.R., Taylor, P.A., MacGregor, J.F., 1996. Missing data methods in PCA and PLS: score calculations with incomplete observations. Chemom. Intell. Lab. Syst. 35 (1), 45–65.
Nomikos, P., MacGregor, J.F., 1994. Monitoring batch processes using multiway principal component analysis. AIChE J. 40 (8), 1361–1375.
Nomikos, P., MacGregor, J.F., 1995. Multi-way partial least squares in monitoring batch processes. Chemom. Intell. Lab. Syst. 30 (1), 97–108.
Radhakrishnan, V., Ramasamy, M., Zabiri, H., Do Thanh, V., Tahir, N., Mukhtar, H., Hamdi, M., Ramli, N., 2007. Heat exchanger fouling model and preventive maintenance scheduling tool. Appl. Therm. Eng. 27 (17–18), 2791–2802.
Teruel, E., Cortes, C., Diez, L.I., Arauzo, I., 2005. Monitoring and prediction of fouling in coal-fired utility boilers using neural networks. Chem. Eng. Sci. 60 (18), 5035–5048.
Tulsyan, A., Garvin, C., Undey, C., 2018. Machine-learning for biopharmaceutical batch process monitoring with limited data. IFAC-PapersOnLine 51 (18), 126–131.
Urrutia, J., 2016. Fouling in Emulsion Polymerization Reactors. University of the Basque Country UPV/EHU Ph.D. thesis.
Wang, D., 2011. Robust data-driven modeling approach for real-time final product quality prediction in batch process operation. IEEE Trans. Ind. Inform. 7 (2), 371–377.
Wang, D., Srinivasan, R., 2009. Multi-model based real-time final product quality control strategy for batch processes. Comput. Chem. Eng. 33 (5), 992–1003.
Wang, Y., Yuan, Z., Liang, Y., Xie, Y., Chen, X., Li, X., 2015. A review of experimental measurement and prediction models of crude oil fouling rate in crude refinery preheat trains. Asia-Pacific J. Chem. Eng. 10 (4), 607–625.
Westerhuis, J.A., Kourti, T., MacGregor, J.F., 1998. Analysis of multiblock and hierarchical PCA and PLS models. J. Chemom. 12 (5), 301–321.
Wold, S., 1978. Cross-validatory estimation of the number of components in factor and principal components models. Technometrics 20 (4), 397–405.
Wold, S., Kettaneh-Wold, N., MacGregor, J., Dunn, K., 2010. Batch process modeling and MSPC. In: Comprehensive Chemometrics. Elsevier, pp. 163–197.
Wu, O., Bouaswaig, A.E.F., Schneider, S.M., Moreno Leira, F., Imsland, L., Roth, M., 2018. Data-driven degradation model for batch processes: a case study on heat exchanger fouling. In: Computer Aided Chemical Engineering, 43. Elsevier, pp. 139–144.
Wu, O., Dalle Ave, G., Harjunkoski, I., Imsland, L., Schneider, S.M., Bouaswaig, A.E.F., Roth, M., 2019a. Short-term scheduling of a multipurpose batch plant considering degradation effects. In: Proceedings of the European Symposium on Computer-Aided Process Engineering (ESCAPE-29).
Wu, O., Imsland, L., Brekke, E., Schneider, S.M., Bouaswaig, A.E.F., Roth, M., 2019b. Robust state estimation for fouling evolution in batch processes using the em algorithm. In: Proceedings of the IFAC International Symposium on Dynamics and Control of Process Systems (DYCOPS).
Xenos, D.P., Kopanos, G.M., Cicciotti, M., Thornhill, N.F., 2016. Operational optimization of networks of compressors considering condition-based maintenance. Comput. Chem. Eng. 84, 117–131.
Yabuki, Y., MacGregor, J.F., 1997. Product quality control in semibatch reactors using midcourse correction policies. Ind. Eng. Chem. Res. 36 (4), 1268–1275.
Yan, Z., Huang, B.-L., Yao, Y., 2015. Multivariate statistical process monitoring of batch-to-batch startups. AIChE J. 61 (11), 3719–3727.
Yeap, B., Wilson, D., Polley, G., Pugh, S., 2004. Mitigation of crude oil refinery heat exchanger fouling through retrofits based on thermo-hydraulic fouling models. Chem. Eng. Res. Des. 82 (1), 53–71.
Yin, S., Li, X., Gao, H., Kaynak, O., 2015. Data-based techniques focused on modern industry: an overview. IEEE Trans. Ind. Electron. 62 (1), 657–667.
Zhang, J., Morris, A., Martin, E., Kiparissides, C., 1999. Estimation of impurity and fouling in batch polymerisation reactors through the application of neural networks. Comput. Chem. Eng. 23 (3), 301–314.
Zmitrowicz, A., 2006. Wear patterns and laws of wear–a review. J. Theoretical Appl. Mech. 44 (2), 219–253.