Computers and Chemical Engineering 128 (2019) 188–200

Contents lists available at ScienceDirect

Computers and Chemical Engineering

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

Forecasting of process disturbances using *k*-nearest neighbours, with an application in process control[☆]

Francesco Borghesan[a,*], Moncef Chioua[b], Nina F. Thornhill[a]

[a] Imperial College London, South Kensington Campus, London SW7 2AZ, UK
[b] ABB Corporate Research Center, Wallstadter Strasse 59, Ladenburg 68526, Germany

ARTICLE INFO

Article history:
Received 18 January 2019
Revised 2 May 2019
Accepted 4 May 2019
Available online 27 May 2019

Keywords:
Plantwide disturbance
Time series prediction
Nearest neighbours
Process control
MPC
Buffer tank

ABSTRACT

This paper examines the prediction of disturbances based on their past measurements using *k*-nearest neighbours. The aim is to provide a prediction of a measured disturbance to a controller, in order to improve the feed-forward action. This prediction method works in an unsupervised way, it is robust against changes of the characteristics of the disturbance, and its functioning is simple and transparent. The method is tested on data from industrial process plants and compared with predictions from an autoregressive model. A qualitative as well as a quantitative method for analysing the predictability of the time series is provided. As an example, the method is implemented in an MPC framework to control a simple benchmark model.

1. Introduction

A disturbance is an undesired, transitory deviation of those inputs of the controlled system that are not manipulable by the controller. In process plants, such disturbances can affect the quality of the product, or they can cause a malfunction of the site machinery and accelerate its wear (Yuan and Qin, 2014). Further, a disturbance originating from one unit can propagate to other units because of material, energy, or information and control interconnections. When a disturbance propagates through a section of a plant and affects more process variables, such disturbance is defined as a plantwide disturbance.

The paper is about the prediction of process disturbances. It solves the problem, provides general guidelines, and suggests a possible application in process control. Specifically, we show how the disturbance prediction can be obtained using two *k*-nearest neighbour methods. Previous uses of *k*-nearest neighbours for time series prediction have typically predicted just one sample a few steps ahead. Instead, this paper shows it is possible to achieve the more challenging task of predicting the future evolution of the time series. A condition for a good prediction is the regularity and predictability of the time-series. The paper shows a qualitative and quantitave method to analyse the predictability. Further, it compares the prediction performance of the *k*-nearest neighbours method with the one obtainable with an autoregressive model, and highlights the capability of the *k*-nearest neighbours to learn new disturbance patterns. Thereafter, a simple possible application of the method in process control is provided, by implementing the disturbance prediction in an MPC framework. The aim of the MPC with disturbance forecast is to stop the propagation of a disturbance coming from the inflow of a stirred tank heater, while keeping the level within tight constraints.

2. Background and motivation

Forecasting the time series of a disturbance is an open question and, to this scope, literature provides many methods such as ARIMA models and Artificial Neural Networks (De Gooijer and Hyndman, 2006). The methods are classified into first principle models, statistical methods and data-driven methods. Difficulties in choosing the method arise mainly when the disturbances are caused by non-linear effects such as limit cycles, or when they arise because of random events.

Process control is an interesting application for time series forecasting because providing information regarding the future evolution of a disturbance can improve the performance of a controller. If a disturbance is measured before it enters and affects the

[☆] 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.

[*] Corresponding author.
E-mail address: f.borghesan@imperial.ac.uk (F. Borghesan).

https://doi.org/10.1016/j.compchemeng.2019.05.009

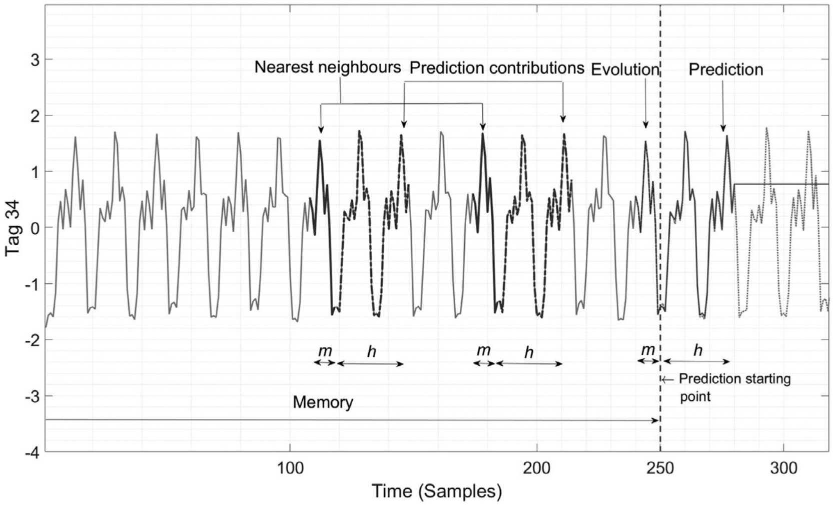


Fig. 1. An example of a persistent disturbance and representation of terminology.
Source: ©PSE 2018, reprinted with permission from 13th International Symposium on Process Systems Engineering.

controlled system, literature speaks about *previewing* a disturbance. The principle of previewing is that if there is a transport delay of i samples between the instant the disturbance is measured and the instant the disturbance enters in the system, the controller can know the future evolution over i samples of the disturbance. While previewing is an established practice with real and commercial applications in process control (Nunokawa et al., 1997; Yoshitani and Hasegawa, 1998; Birla and Swarup, 2015), truly predicting a disturbance is not. Indeed, feedforward control is used routinely in process systems to compensate for measured disturbances when they are detected (Liptak, 2005; Camacho and Bordons, 2008), but the standard implementation of feedforward control does not include a prediction of the future evolution of the disturbance.

Cairano et al. (2014) improved the control of the battery management of a hybrid electric vehicle by predicting the disturbance, the driver's driving style, with a data-driven Markov chain. Klenske et al. (2016) improved the control of an electrically actuated telescope proposing a model based on Gaussian processes that learns to fit nearly-periodic disturbances. However, these methods adapt their models based on the last measured samples and do not exploit the whole history of the measured disturbance.

To address these issues, Borghesan et al. (2018a) proposed k-nearest neighbours methods to forecast the future trend of persistent disturbances. Such methods were able to provide good predictions of time trends coming from a refinery. This article develops and analyses the method in more detail, starting by assessing when a time series is first of all predictable. The prediction performances are analysed for their robustness to variations in the characteristics of the time series such as changes in mean values or of the pattern in the case of a persistent disturbance. The present paper extends also the former work for the choice of the tuning parameters of the method, which improves the prediction performance significantly. Whereas (Borghesan et al., 2018a) focused on persistent disturbances, this article addresses also the case of time series characterized by abrupt disturbance phenomena. Finally, it compares the predictions performance of the k-nearest neighbours method with the prediction obtained by an autoregressive model.

The paper provides a simple application of the prediction, implementing the k-nearest neighbours method in an MPC framework. The controller has to stop the propagation of a plantwide disturbance by minimizing the variation of the outflow of a tank affected by persistent and abrupt disturbances of the inflow.

3. Disturbance prediction

3.1. Introduction

The k-nearest neighbours method supposes that the current time series segment will evolve in future like a past time series segment (not necessarily a recent one) evolved previously (Kantz and Schreiber, 2004). The task therefore is to identify past segments of the time series which are similar to the present one according to a certain norm. At current sample time N, the algorithm considers a measurement signal made of N samples $\mathbf{y}_M(N) = [y(1)\ y(2)\ldots y(N)]$, termed 'memory'. The size N of the memory increases as more samples $y(\cdot)$ are recorded. The segment made of the last m samples of $\mathbf{y}_M(N)$ is $\mathbf{y}_E(N) = [y(N-m+1)\quad y(N-m+2)\ldots y(N)]$. This segment represents the current evolving disturbance pattern and is called $\mathbf{y}_E(N)$ 'evolution', while m is known as the 'embedding dimension' (Kantz and Schreiber, 2004). The algorithm searches within the memory $\mathbf{y}_M(N)$ for the k time series intervals of length m that are most similar to $\mathbf{y}_E(N)$. The similarity is measured in this paper with the Euclidean distance, D (Eq. (1)). These k time series intervals $\mathbf{y}_j,\ \ j = 1,\ldots,k$ are the k-nearest neighbours and are those k time series intervals within $\mathbf{y}_M(N)$ with smallest Euclidean distances:

$$D\big(\mathbf{y}_E(N), \mathbf{y}_j\big) = \sqrt{\sum_{i=1}^{m}\big(y_E(i) - y_j(i)\big)^2} \tag{1}$$

As depicted in Fig. 1, each nearest neighbour $\mathbf{y}_j$ is followed by a time series of length h, where h is the desired prediction horizon. These time series of length h are called the 'prediction contributions' $\mathbf{y}_{P\ j}, j = 1,\ldots,k$. They show how their corresponding nearest neighbour evolved over an interval of length h. Therefore, they are the basis for building the prediction vector, as will be described in Section 3.2.1. Since all prediction contributions must stay within the memory $\mathbf{y}_M(N)$, there is a constraint for the starting points r_j of the nearest neighbours $\mathbf{y}_j$

$$\begin{aligned}&\mathbf{y}_j = \big[y\big(r_j\big)\ \ y\big(r_j+1\big)\ldots\ \ y\big(r_j+m\big)\big],\\ &j = 1,\ldots,k,\\ &r_j \in [1; N-h-m+1]\end{aligned} \tag{2}$$

Consequently $\mathbf{y}_{Pj}$ is defined as

$$\begin{aligned}\mathbf{y}_{Pj} &= \left[y\left(r_j+m+1\right) \quad \dots \quad y\left(r_j+m+h\right)\right] \\ j &= 1,\dots,k\end{aligned} \tag{3}$$

3.2. k-Nearest neighbours

3.2.1. Two versions of the algorithm

Unweighted and weighted versions of the algorithm differentiate themselves at this step. For the unweighted version, the prediction is an average of the prediction contributions. The prediction is a vector $\hat{\mathbf{y}}(N) = [\hat{y}(N+1)\dots\ \hat{y}(N+h)]$ where:

$$\hat{y}(N+i) = \frac{1}{k}\sum_{j=1}^{k} y_{P\ j}(i), i = 1,\dots,\ \ h \tag{4}$$

In the weighted version of the algorithm, the nearest neighbours contribute to the prediction proportionally to their distance from the evolution. First a weight w_j for each nearest neighbours is calculated. The weight $w_j = 1$ if the nearest neighbour $\mathbf{y}_j$ has the smallest distance with $\mathbf{y}_E(N)$. Conversely, $w_j = 0$ if the corresponding nearest neighbour $\mathbf{y}_j$ is the most distant from $\mathbf{y}_E(N)$.

$$w_j = \begin{cases} \dfrac{\max\limits_{\ell=1\dots k}\left(D(\mathbf{y}_E(N),\mathbf{y}_\ell)\right) - D(\mathbf{y}_E(N),\mathbf{y}_j)}{\max\limits_{\ell=1\dots k}\left(D(\mathbf{y}_E(N),\mathbf{y}_\ell)\right) - \min\limits_{\ell=1\dots k}\left(D(\mathbf{y}_E(N),\mathbf{y}_\ell)\right)} & \text{with } j = 1,\dots,k \quad \text{if } k \ge 2 \\ 1 & \text{with } j = 1 \quad\quad\ \ \text{if } k = 1 \end{cases} \tag{5}$$

The prediction vector $\hat{\mathbf{y}}(N) = [\hat{y}(N+1)\dots\ \ \hat{y}(N+h)]$ is a weighted average of the prediction contributions, where:

$$\hat{y}(N+i) = \frac{1}{\sum_{j=1}^{k} w_j}\sum_{j=1}^{k} w_j\ y_{P\ j}(i) i = 1,\dots,\ \ h \tag{6}$$

The weighted method has been developed to guarantee a higher robustness and precision of the prediction, since it is able to partially ignore the nearest neighbours that are too distant from the evolution. This is interesting when the algorithm works over an interval of time and the possibility of finding exactly k good nearest neighbours can change over time.

3.2.2. A fixed number of neighbours versus a maximal neighbours distance

Concerning the number of nearest neighbours to be found, a possibility is setting a fixed number k, independently from the instant of time N in which the prediction should start. Kantz and Schreiber (2004) proposed to choose the maximal distance $D_{\max}$ between evolution and a neighbour, for which the neighbour can be considered as one of the nearest neighbours that will evolve as the evolution $\mathbf{y}_E(N)$.

Indeed, if the distance $D(\mathbf{y}_E(N), \mathbf{y}_j)$ is too big, then it is likely that the time segment $\mathbf{y}_j$, chosen as nearest neighbour of $\mathbf{y}_E(N)$, will not evolve in the future as the evolution $\mathbf{y}_E(N)$ will do because the chosen nearest neighbour does not capture the same dynamic occurring to the evolution. In this case, the time segment $\mathbf{y}_j$ can be defined as false nearest neighbour. Choosing constantly a number k of nearest neighbours, without taking into account their distance from the evolution, has the risk of selecting false nearest neighbours.

On the other hand $D_{\max}$ cannot be too small. For example, some time series segments might be affected by noise in the measurement. This implies that although such time series segments are relatively distant from the evolution because of noise, they reflect the same dynamic. In a such case, averaging more prediction contributions (without considering false nearest neighbours) can have the beneficial effect of finding the expected value of the prediction.

Using the approach of $D_{\max}$, k is chosen at each sample N as

$$\begin{aligned} k(N, D_{\max}) &= |Y_{NN}(N, D_{\max})| \\ Y_{NN}(N, D_{\max}) &= \{\mathbf{y}_1,\dots,\mathbf{y}_k : D(\mathbf{y}_E(N),\mathbf{y}_j) \le D_{\max}\} \end{aligned} \tag{7}$$

In the above expression, $|\cdot|$ has been used to indicate the number of items in the set. Setting a fixed $D_{\max}$ is different from setting a fixed k only if $D_{\max}$ and k are constant over a certain interval of time. Instead, considering a prediction starting at a certain instant of time, certain values of $D_{\max}$ correspond to a specific value of k.

3.3. Measuring accuracy

3.3.1. Measuring the accuracy of the prediction for varying values of k and m at an instant of time N

For a given starting point of the prediction N, the tuning parameters of the methods are m and k. Varying m and k causes the average prediction distance to change. The prediction distance E_{av} is the Euclidean distance between the prediction and the corresponding segment of the actual future time series (not known in advance to the algorithm) divided by the chosen prediction horizon h so that the value does not depend on the length of the prediction horizon.

$$E_{av}(N,k,m) = \frac{1}{h}D(\hat{\mathbf{y}}(N,k,m),\ [y(N+1)\dots y(N+h)]) \tag{8}$$

The smaller $E_{av}(N)$ is, the more accurate the prediction is.

3.3.2. Measuring the accuracy of the prediction for varying values of k and m over an interval of time

Eq. (8) measures the error made at one sample time N. However, the optimal parameters m and k change for different sampling times. Consequently, it is important as well to measure the performance of a fixed m and k over an interval of time. We introduce in this paper the measure

$$E_{av\ \mathrm{int}}^{k_{\max}}(N_1,N_2,k,m) = \frac{1}{N_2-N_1}\sum_{N=N_1}^{N_2} E_{av}(N,k,m) \tag{9}$$

Such a measure is an average of the prediction error over an interval.

Similarly, it is possible to fix a value $D_{\max}$ and m over a certain interval. In this case k might change but depends on a fixed $D_{\max}$ as Eq. (7) described.

$$E_{av\ \mathrm{int}}^{D_{\max}}(N_1,N_2,D_{\max},m) = \frac{1}{N_2-N_1}\sum_{N=N_1}^{N_2} E_{av}(N,k(N,D_{\max}),m) \tag{10}$$

To not overload the notation, we use the following expression when the discussion refers to both $E_{av\ \mathrm{int}}^{k_{\max}}$ and $E_{av\ \mathrm{int}}^{D_{\max}}$ and the main focus is the interval of time $N_1 \to N_2$.

$$E_{av\ \mathrm{int}}(N_1,N_2) \tag{11}$$

3.4. Training of the optimal parameters

3.4.1. Finding the tuning parameters through an optimization problem

A rule of thumb is to set $m = \lfloor\frac{1}{f_0}\rfloor$, where $\lfloor\cdot\rfloor$ is the floor function that provides the smaller or equal integer of a real number, and f_0 is the main frequency of the persistent disturbance. In this case m is close to the length of one instance of the repeating pattern of the persistent disturbance. The number of nearest neighbours k should be $k = \lfloor \frac{N}{m} \rfloor$, which means equal to the number of occurrences of the repeating pattern. This strategy will be called *base strategy* in the rest of the paper.

The present paper proposes to find the parameters through a training algorithm based on a search grid. This involves enumerating the possible combinations within a range of the parameters k and m, or $D_{\max}$ and m depending on the strategy used, and choosing the pair of parameters that minimises $E_{av\ \mathrm{int}}(N_1^{\mathrm{tr}}, N_2^{\mathrm{tr}})$ with $N_2^{\mathrm{tr}} < N - h$, where N is the current instant of time. The interval $N_1^{\mathrm{tr}} \rightarrow N_2^{\mathrm{tr}}$ is the training interval and corresponds to values $y(N_1), y(N_1+1) \ldots y(N_2)$ that have been already measured at instant N. Since these values are known, the training algorithm can compare what the prediction would be for each instant of the training interval and what it was in reality. The condition $N_2^{\mathrm{tr}} < N - h$ is set because the algorithm had to measure the values of the time series to calculate the value of $E_{av}(N_2)$ used to calculate thereafter $E_{av\ \mathrm{int}}(N_1^{\mathrm{tr}}, N_2^{\mathrm{tr}})$. The length of the interval $N_1^{\mathrm{tr}} \rightarrow N_2^{\mathrm{tr}}$ is set to be long enough to cover some instances of the pattern of the disturbance. The training should not be conducted at every sample N, rather at discretion when, for example, new disturbances patterns are visible or when the noise level of the measurement is changing because of a change of the operating point of the plant.

3.4.2. Evaluating the accuracy of the training process for the choice of the tuning parameters

The parameters obtained from the training are $(k_{\mathrm{opt}}^{\mathrm{tr}}, m_{\mathrm{opt}}^{\mathrm{tr}})$ or $(D_{\max\ \mathrm{opt}}^{\mathrm{tr}}, m_{\mathrm{opt}}^{\mathrm{tr}})$, depending whether a fixed k or $D_{\max}$ was defined. To measure how accurate the training was, the prediction obtained from such parameters should be evaluated over a test interval $N_1^{\mathrm{test}} \rightarrow N_2^{\mathrm{test}}$. The term of comparison is the value of $E_{av\ \mathrm{int}}(N_1^{\mathrm{test}}, N_2^{\mathrm{test}})$ obtained with the parameters $k_{\mathrm{opt}}^{\mathrm{test}}$, $m_{\mathrm{opt}}^{\mathrm{test}}$, $D_{\max\ \mathrm{opt}}^{\mathrm{test}}$ that are the optimal ones for the testing interval $N_1^{\mathrm{test}} \rightarrow N_2^{\mathrm{test}}$.

The metric γ is the ratio between the value of $E_{av\ \mathrm{int}}(N_1^{\mathrm{test}}, N_2^{\mathrm{test}})$ obtained using the parameters $k_{\mathrm{opt}}^{\mathrm{tr}}$, $m_{\mathrm{opt}}^{\mathrm{tr}}$, $D_{\max\ \mathrm{opt}}^{\mathrm{tr}}$ and the one obtained using $k_{\mathrm{opt}}^{\mathrm{test}}$, $m_{\mathrm{opt}}^{\mathrm{test}}$, $D_{\max\ \mathrm{opt}}^{\mathrm{test}}$. We define two versions $\gamma_{k_{\max}}$ and $\gamma_{D_{\max}}$ of γ, according to the strategy used:

$$\gamma_{k_{\max}} = \frac{E_{av\ \mathrm{int}}^{k_{\max}}(N_1^{\mathrm{test}}, N_2^{\mathrm{test}}, k_{\mathrm{opt}}^{\mathrm{test}}, m_{\mathrm{opt}}^{\mathrm{test}})}{E_{av\ \mathrm{int}}^{k_{\max}}(N_1^{\mathrm{test}}, N_2^{\mathrm{test}}, k_{\mathrm{opt}}^{\mathrm{tr}}, m_{\mathrm{opt}}^{\mathrm{tr}})}$$

$$\gamma_{D_{\max}} = \frac{E_{av\ \mathrm{int}}^{D_{\max}}(N_1^{\mathrm{test}}, N_2^{\mathrm{test}}, k_{\mathrm{opt}}^{\mathrm{test}}, m_{\mathrm{opt}}^{\mathrm{test}})}{E_{av\ \mathrm{int}}^{D_{\max}}(N_1^{\mathrm{test}}, N_2^{\mathrm{test}}, k_{\mathrm{opt}}^{\mathrm{tr}}, m_{\mathrm{opt}}^{\mathrm{tr}})} \tag{12}$$

3.5. Assessing the predictability of a time series

To distinguish a time series with some recurring and deterministic patterns from a stochastic time series, the predictability of the time series should be assessed. This can be done with a qualitative method or a quantitative one.

3.5.1. Qualitative method to assess the predictability

A practical qualitative method to estimate the predictability of a time series is to plot its phase space plot. The phase space plot associates to a value $y(N)$ of the time series the corresponding value $y(N - delay)$. It possible also to draw a 3D plot where two values $y(N - delay_1)$ and $y(N - delay_2)$ are associated to $y(N)$. These plots highlight the presence of recurrent patterns in the time series and consequently their self predictability, explained in Section 3.1.

3.5.2. Quantitative method to assess the predictability

A quantitative method we propose derives from the Harris index described in Desborough and Harris (1992). The Harris index was developed for control loop performance assessment. The principle is that if the time series of the error of a controller is predictable, then the control has poor performance. Instead of being used on the time series of the error of a controller, here the procedure is used on the time series of a disturbance to assess the predictability of the time series. The procedure is to tune an autoregressive (AR) model of the time series.

$$\hat{y}(N+h) = a_0 + a_1 y(N) + a_2 y(N-1) + \cdots + a_m y(N-m+1) \tag{13}$$

where the horizon h and the order m of the AR model can be chosen as described in Thornhill et al. (1999).

To tune the coefficients, a section of the time series with constant characteristics as the mean value must be provided. The coefficients of the model are found so that the residuals $r(N) = y(N) - \hat{y}(N)$ of the prediction are minimized for the training time series. The variance of the residuals is compared to mean square of the time series and this gives the Harris index.

$$H_{index} = \frac{\sigma_r^2}{\mathrm{mse}(y)} \tag{14}$$

where σ is the variance of the residuals and mse(y) is the mean square of the time series. If the residuals (and their variance) are large compared to the values of the time series (and consequently to their mean square), then the AR model is not able to capture fully the dynamic of the time series and, consequently, the time series has a low predictability. A value of $H_{index} = 0$ corresponds to a highly predictable time series and low quality controller. Therefore, a predictability index can be derived as

$$P_{index} = 1 - H_{index} \tag{15}$$

where $P_{index} = 1$ indicates that the time series is highly predictable and $P_{index} = 0$ indicates that the time series is not predictable.

A high value of the predictability index is a guarantee that the prediction will work. In case of a persistent disturbance, a value of predictability index smaller than 0.5 makes questionable the use for an application in process control.

For abrupt disturbances instead, a low value of the predictability index means that the abrupt event is not predictable before occurring. However, once it starts, its future evolution might be predicted as Section 5.3 will show.

3.6. The autoregressive model as reference of comparison

The widely used AR model, employed for the predictability index in Section 3.5.2, can be used also as comparison for the k-nearest neighbours methods presented in this paper.

To compare the performance between the two methods, the same intervals of time are used for the training of the two methods. As Section 3.4.1 described, when the k-nearest method is trained on an interval $N_1^{\mathrm{tr}} \rightarrow N_2^{\mathrm{tr}}$, a memory, starting from the first recorded instant, should be provided to the algorithm. Consequently, to compare fairly the training done on the two methods, the AR model uses the interval $1 \rightarrow N_2^{\mathrm{tr}}$ for the training.

The training and the prediction of the model is done using the routines in Matlab *estimate* and *forecast*. Once the prediction with the trained autoregressive model is obtained, the quality of the prediction can be obtained similarly to what has been explained in Sections 3.3.1 and 3.3.2.

$$E_{av}^{\mathrm{AR}}(N, AR_{\mathrm{model}}) = \frac{1}{h} D(\hat{\mathbf{y}}(N, AR_{\mathrm{model}}),\ [y(N+1) \ldots y(N+h)]) \tag{16}$$

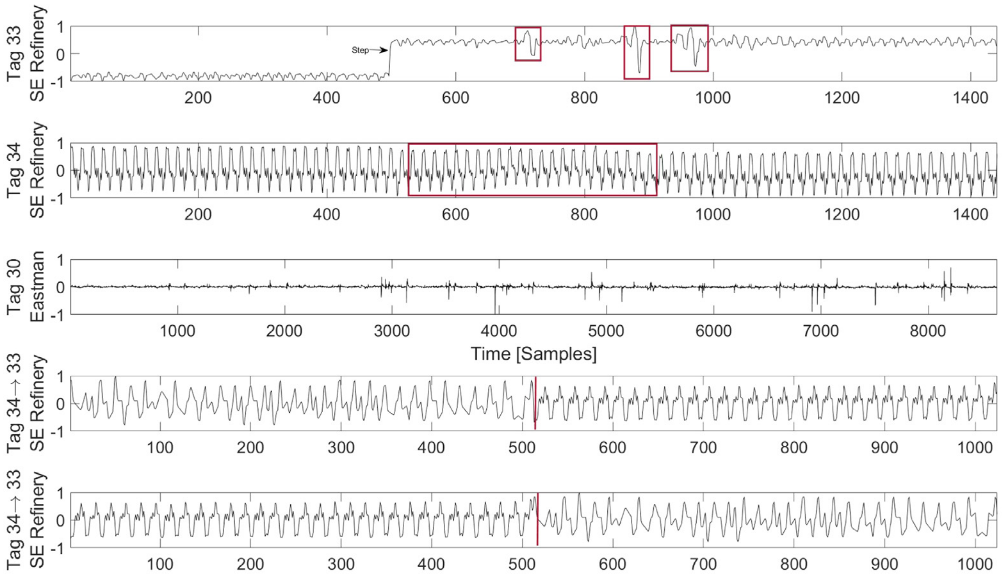


Fig. 2. The time series considered for the section regarding the results of the prediction methods.

$$E_{av\ \mathrm{int}}^{\mathrm{AR}}(N_1, N_2, AR_{\mathrm{model}}) = \frac{1}{N_2 - N_1} \sum_{N=N_1}^{N_2} E_{av}^{\mathrm{AR}}(N, AR_{\mathrm{model}}) \tag{17}$$

where AR_{model} refers to the autoregressive model with its coefficients and previous time series elements.

As in Section 3.4.2 described, a coefficient γ^{AR} can be introduced. If in Section 3.4.2 the aim of γ is to understand how good the parameters for the k-nearest neighbours are, here the aim of γ^{AR} is to understand how good is still, over the testing interval, the trained autoregressive model $AR_{\mathrm{opt}}^{\mathrm{tr}}$ compared to the model $AR_{\mathrm{opt}}^{\mathrm{test}}$ that would result from the training on the testing interval.

$$\gamma_{\mathrm{AR}} = \frac{E_{av\ \mathrm{int}}^{\mathrm{AR}}(N_1^{\mathrm{test}}, N_2^{\mathrm{test}}, AR_{\mathrm{opt}}^{\mathrm{test}})}{E_{av\ \mathrm{int}}^{\mathrm{AR}}(N_1^{\mathrm{test}}, N_2^{\mathrm{test}}, AR_{\mathrm{opt}}^{\mathrm{tr}})} \tag{18}$$

4. The real industrial disturbances

Fig. 2 shows the time series used to analyse the prediction methods. The upper and second panel of the figure show the Tag 33 and Tag 34 from a SE Asian refinery. These time series have been presented previously, for instance in Thornhill (2005).

Tag 33 has a persistent and regular oscillation. However, the repeating pattern is not so regular as the one of Tag 34. Further, there is a step change around sample 500, as shown in the arrow in the top panel of Fig. 2. The first 500 samples have different values than the later samples, even though the pattern of oscillation is similar. Therefore, one can expect lower predictability just after the step change since there will be no nearest neighbours in the memory on which to base the prediction. There are also three abrupt transient disturbances within the red rectangles shown on the figure that also will present challenges for prediction.

Tag 34 has a very regular and persistent oscillation, and it is expected to be highly predictable. However, the local mean value is not constant between sample 500 and 900, and this will reduce its predictability to an extent.

The third panel in Fig. 2 shows Tag 30 of dataset from the Eastman Chemical Company described in Thornhill et al. (2003). This time series is affected by phenomena of abrupt disturbances that take the form of short spikes of varying amplitude with durations of 3 to 5 samples.

The time trend of Tag 30 is affected by noise between the abrupt disturbances, and random noise is inherently unpredictable. Moreover, the occurrence of an abrupt disturbance is not predictable. However, the results will show that once the abrupt phenomenon starts, the k-nearest neighbours algorithm can detect the phenomenon and provide useful predictions.

The fourth panel shows a signal created by concatenating a section of the Tag 33 with one of the Tag 34. The signal in the fifth panel is built concatenating the same two time series but in the opposite order compared to the order of the fourth panel. These two time series, built on-purpose, are used in Section 5.2 to show the effects of a time changing disturbance pattern on the prediction performances.

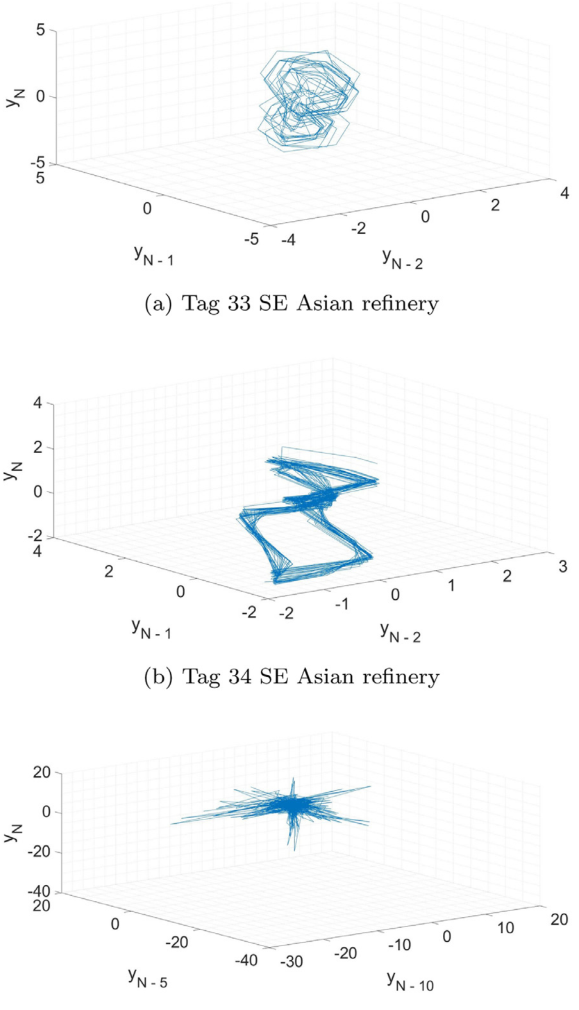
4.1. *Qualitative assessment the predictability of the industrial disturbances*

Section 3.5.1 described how to assess qualitatively the predictability of a time series through the phase plots. Fig. 3 shows such plots for the three main time series considered.

The plot in Fig. 3b, corresponding to Tag 34, is far more well defined than the other two plots. Tag 34 shows how the k-nearest neighbours method can give good predictions because the trajectory tends to evolve in the same way from any given point. Tag 33 shows also some repeating trajectories but, qualitatively, the pattern is not well defined. The confused pattern at the centre of the plot for Tag 30 suggests that Tag 30 has generally low predictability. However, there are spikes radiating from the centre. Once the k-nearest neighbours method has identified a spike, then Fig. 3c shows that a return to the centre can be predicted.

4.2. *Quantitative assessment the predictability of the industrial disturbances*

Section 3.5.2 described how to assess quantitatively the predictability of time series. The method is applied to the three industrial disturbances. The prediction horizon is set to $h = 15$ and the order is $m = 30$. Table 1 shows the predictability index for the three main time series considered. The quantitative analysis reflects what was already understandable from the qualitative analysis. The time series of the Tag 34 is extremely regular and predictable. However, the time series of Tag 33 and especially the time series of Tag 30 are harder to predict. Nevertheless, Fig. 3c



(a) Tag 33 SE Asian refinery

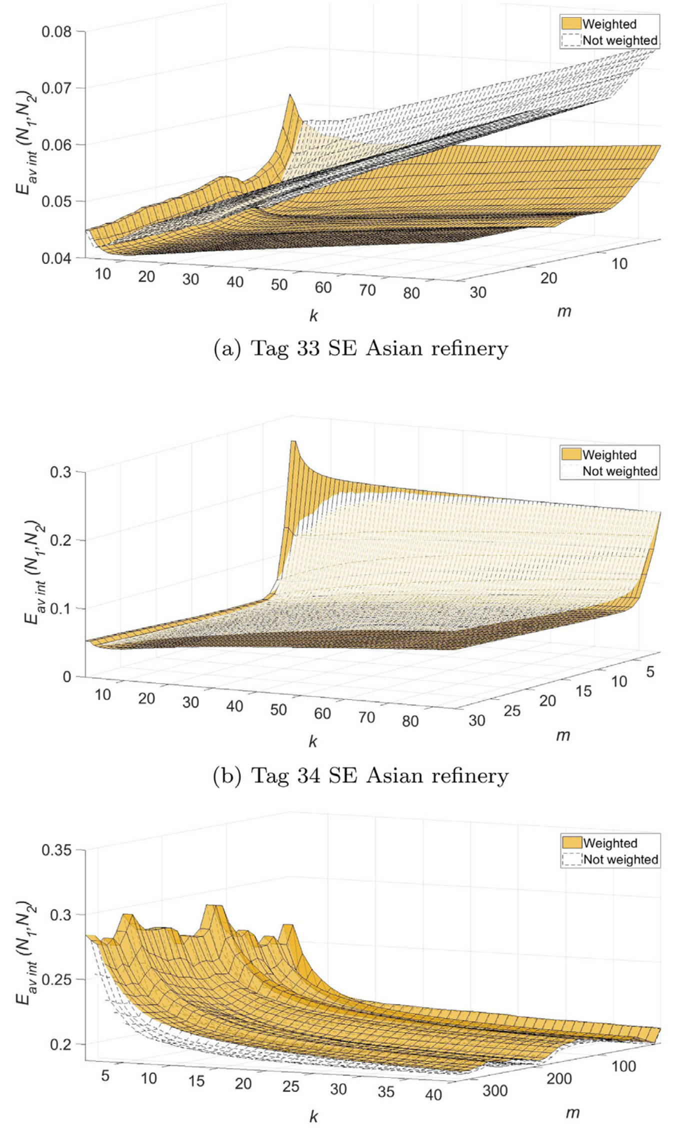
(b) Tag 34 SE Asian refinery

(c) Tag 30 Eastman Chemical Company

Fig. 3. Phase space plots of sections of the considered time series.

Table 1
Predictability index for the three main time series considered with $h = 15$ and $m = 30$.

	Tag 33	Tag 34	Tag 30
H_{index}	0.610	0.984	0.001



(a) Tag 33 SE Asian refinery

(b) Tag 34 SE Asian refinery

(c) Tag 30 Eastman Chemical Company

Fig. 4. Effect of parameters k and m on prediction of Tags 33 and 34 with $N_1 = 400, N_2 = 1425$ and $h = 15$. Effect of parameters k and m on prediction of Tag 30 with $N_1 = 6000, N_2 = 7000$ and $h = 15$.

suggested that the evolution of the abrupt spiky features of Tag 30 can be predicted, once the spike has been detected.

5. Results: prediction

This section examines the influence of the number of nearest neighbours k, and the embedding dimension m, on the quality of the predictions. Further, it compares the performance of the unweighted and weighted method explained in Section 3.2.1, of using a fixed k or a constant D_{Max} as described in Section 3.2.2, and finally compares the k-nearest neighbours method with the predictions made by an autoregressive model.

5.1. Effect of k and m on $E_{av\ \text{int}}$

Fig. 4 shows the effects of k and m on $E_{av\ \text{int}}$ over an interval of time $N_1 \rightarrow N_2$. Indeed, it plots the values of $E_{av\ \text{int}}$ for several combinations of k and m using the weighted and unweighted method for the three main considered time series. In all cases, the length of the interval $N_1 \rightarrow N_2$ is 1000 samples.

For the Tag 33 and 34 from the SE Asian refinery the prediction horizon is $h = 15$, roughly the period of the oscillation which is 17. Tag 30 is affected by abrupt disturbances and once an abrupt disturbance starts, it is possible to predict its future values until the end of the disturbance pattern. The abrupt disturbance pattern of Tag 30 lasts generally 15 samples. Therefore, $h = 15$ as well for Tag 30.

For such sensitivity analysis, the maximal embedding dimension m has been chosen as roughly two times the period of one oscillation. Since the period of oscillation is 17 samples for the Tags 33 and 34 then $m \approx 35$. The reason is that the most recent cycles are likely to be the most important to understand the future trend of a disturbance. However, there is no oscillation in Tag 30 but there are roughly six phenomena of abrupt disturbances every one thousand samples, although not at regular intervals (with a periodicity of about $\frac{1000}{6} \approx 165$). Given a periodicity of about 165 samples, the maximal possible value of m is 330, following the same principle used for the Tags 33 and 34.

The maximal number of neighbours has been set as the number of samples N in the maximal memory N_2 over the interval divided by the length of one oscillation. Indeed, the maximal number of nearest neighbours that can be found in a series is equal to the number of oscillations that have occurred in the memory. Consequently $k_{\max} = \frac{1420}{17} \approx 85$ for Tags 33 and 34 and $k_{\max} = \frac{7000}{165} \approx 50$.

5.1.1. The effect of m on $E_{av\ int}$

Fig. 4 shows that the value of $E_{av\ \mathrm{int}}$ can be high for very small values of m regardless of the value of k. This is the case of the two time series Tag 33 and 34 coming from the SE Asian refinery characterized by a persistent disturbance. This is because an embedding dimension that does not capture a significant portion of the period of oscillation does not correctly identify the shape of the disturbance. For example, two points might have similar values in the ascending or descending phase of an oscillation but the prediction contributions $\mathbf{y}_P$ could be very different, thereby worsening the prediction. For Tag 34, more predictable, high values of $E_{av\ \mathrm{int}}$ are present for circa $m \leq 5$. For Tag 33, less predictable, high values of $E_{av\ \mathrm{int}}$ are present for circa $m \leq 7$. Consequently, given in these cases a duration of pattern of circa 17 samples, we can say that m should be higher than a third of the duration of the repeating pattern for persistent disturbances. This limit is inserted in the search grid algorithm for the training of optimal parameters which has been described in Section 3.4.1.

However, Fig. 4c shows that the prediction error for Tag 30 seems to be largely independent from the value of m, with a slight tendency of improvement for small values of m. The reason of this slight improvement is that in case of an abrupt phenomena only a short time segment is needed to evaluate the distance between evolution and nearest neighbours. However, the measure $E_{av\ \mathrm{int}}$ is an average of the values E_{av} for the different instants N. Except for few instants, precisely during the abrupt phenomena, the time series shows noise. Consequently, it is not very relevant which are the nearest neighbours picked for the prediction and consequently their embedding dimension. Since the aim for the Tag 30 is to predict the short abrupt phenomena, then the embedding dimension should not be chosen too large. Indeed, if the length of the evolution is too long, then the part of the evolution occurring before the abrupt event contributes considerably more in calculating the distance $D(\mathbf{y}_E(N), \mathbf{y}_j)$ (from Eq. (1)) than the short abrupt event.

5.1.2. The effect of k on $E_{av\ int}$

For all the time series considered, choosing very few nearest neighbours is not good for the performance. Even Tag 34, which seems to have from Fig. 4c low values of $E_{av\ \mathrm{int}}$ even for very small of k, benefits from having $k \geq 4$. Indeed the minimum value of $E_{av\ \mathrm{int}} = 0.38$ for Tag 34 is obtained for $k = 6, m = 10$. For the same m but $k = 2$, $E_{av\ \mathrm{int}} = 0.42$ instead. The reason is that even persistent disturbances have some randomness. By averaging over more prediction contributions, the resulting prediction is closer to the true value.

Especially for noisy time series, such as Tag 33 and even more Tag 30, k cannot be too small. By using many nearest neighbours, the resulting prediction is filtered and is closer to the time series underneath the hidden signal. For Tag 30 the graph suggests that the a high number of nearest neighbours should be used for the prediction. The presence of noise is the likely explanation of this behaviour. In this specific case k must be higher then 15 as Fig. 4c suggests.

However, after a threshold, high values of k are not beneficial for more regular time series like Tags 33 and 34. Increasing k too much results in choosing time series intervals, called false neighbours, that are considered as nearest neighbours even though they are not very similar to the evolution $\mathbf{y}_E(N)$ and do not develop in the same way. To conclude, the minimal values of E_{av} are roughly given for values of $4 \leq k \leq 10$ for Tag 34 and $4 \leq k \leq 20$ for Tag 33 using the weighted algorithm. For the training of optimal parameters, as in Section 3.4.1, we recommend therefore a lower limit $k \geq 4$.

Table 2
Range of the parameters for the search grid optimization algorithms.

	SE Asian refinery			Eastman Chemical Company		
	Tag 33 and Tag 34			Tag 30		
	k	$D_{\max}$	m	k	$D_{\max}$	m
Min	4	0	5	4	0	1
Discretization step	1	0.001	1	1	0.01	10
Max	85	0.5	30	50	1	50
Prediction horizon				15		

5.1.3. Effects of weighted and unweighted method on E_{av}

Figs. 4a and b show that the weighted method has the advantage of making the value of $E_{av\ \mathrm{int}}$ less sensible to the choice of k for Tags 33 and 34 respectively. This is important because, given a value of k (and m), this value is robust even when the characteristics of the disturbance is changing.

However, the unweighted method is generally better for Tag 30 for a certain value of k because the hypothesis on which the weighted method is based is not valid any more in presence of noise. Indeed, the weighted method assumes that the closer a nearest neighbour is to the evolution, the closer the corresponding prediction contribution will be to the future values of the time series.

5.2. Evaluation of E_{av} over time and different strategies to choose k and m

The kind of analysis in Section 5.1 can be done on past data. Instead, the interest of this section is analysing how good is the predicting performance using different techniques to find the parameters for the k-nearest neighbours method.

Section 3.4.1 explained that a possible choice is to set $m = \lfloor \frac{1}{f_0} \rfloor$ and $k = \lfloor \frac{N}{m} \rfloor$. However, this is a rule of thumb and, furthermore, the choice of f_0 for a signal affected by abrupt disturbances is not trivial. This strategy for the choice of m and k is compared here with the new strategies explained in Section 3.4.1. The analysis is done over a testing interval of time as explained in Section 3.4.2. To provide terms of comparison, the best predictions available at each instant of time N using the weighted and unweighted algorithm are used. Furthermore, the advantages of the k-nearest neighbours are highlighted in comparison to the autoregressive model.

5.2.1. The searching grid, training and testing interval

The strategies are based on grid search algorithms, which means that the parameters are chosen by testing combinations of the parameters that are within a range. Table 2 shows the extremes of such ranges and the discretization step.

The training time has been chosen so that several instances of the patterns of the disturbances are present. For the tags from the SE Asian refinery the length of the interval $N_1^{\mathrm{testing}} \to N_2^{\mathrm{testing}}$ has been of 70 samples. For Tag 30, the length of the training interval was of 985 samples.

The testing interval for the tags from the SE Asian refinery includes the transitory variations, like the step change for Tag 33, that have been described in Section 4. For the Tag 30 the test-

Table 3
Values of $E_{av\ \mathrm{int}}$ obtained with the considered strategies, values of the metric γ to measure the accuracy of the training and values of the metric Ξ to evaluate the performance of weighted method in comparison to the unweighted one.

		Tag 33	Tag 34	Tag 30	Tag 33 → 34	Tag 34 → 33
Opt	Weighted	0.031	0.025	0.168	0.026	0.112
	Ξ	−3.13%	0.00%	+3.07%	−16.1%	+9.80%
Base	Weighted	0.044	0.070	0.195	0.068	0.153
	Ξ	−17.0%	−21.4%	+1.56%	−33.3%	−10.5%
Opt k	Weighted	0.041	0.039	0.186	0.047	0.157
	Ξ	−10.9%	−7.14%	−1.59%	−29.9%	−5.42%
$\gamma_{k_{\max}}$	Weighted	1.000	0.974	1.000	0.766	0.892
	Unweighted	0.913	0.976	0.989	0.642	0.867
Opt $D_{\max}$	Weighted	0.042	0.042	0.193	0.052	0.184
	Ξ	−2.33%	0.00%	0.00%	−11.86%	0.00%
$\gamma^{D_{\max}}$	Weighted	1.000	0.952	1.000	0.731	0.788
	Unweighted	0.977	0.976	1.000	0.661	0.793
AR	–	0.878	0.160	0.199	0.148	0.162
γ^{AR}	–	0.056	0.375	0.938	0.338	0.864

ing interval encompasses eight instances of abrupt disturbances of varying amplitude.

Furthermore this section analyses also the composed time series called Tag 33 → 34 and Tag 34 → 33, which have been described in Section 4. In these two cases, the parameters obtained from the training on the first section of the time series are tested on the second half. This simulates the case when the pattern of the disturbance is changing and the method must learn how to predict this new pattern.

5.2.2. Description of the results

Table 3 shows the values of $E_{av\ \mathrm{int}}$ for the time series considered over their testing interval with different strategies. The strategies are listed below

- ‘OPT’: the best prediction available at each sample time N with a value of $k_{\min} \le k \le k_{\max}$ and $m_{\min} \le m \le m_{\max}$. This is a term of comparison, since it is not possible at an instant N to know already the optimal parameters k and m for calculating the prediction
- ‘BASE’: the prediction obtained using the parameters $k = \lfloor \frac{N}{m} \rfloor$ and $m = \lfloor \frac{1}{f_0} \rfloor$
- ‘OPT K’: The prediction obtained using the weighted method training the algorithm to find the fixed parameters k and m that minimized $E_{av\ \mathrm{int}}(N_1^{\mathrm{tr}}, N_2^{\mathrm{tr}})$
- ‘OPT $D_{\max}$’: The prediction obtained using the weighted method and training the algorithm to find the fixed parameters $D_{\max}$ and m that minimized $E_{av\ \mathrm{int}}(N_1^{\mathrm{tr}}, N_2^{\mathrm{tr}})$
- ‘AR Model’: The prediction obtained using the AR model fitted on the interval $1 \rightarrow N_2^{\mathrm{tr}}$

For each of the strategies considered (excluding the AR model) the values $E_{av\ \mathrm{int}}$ using the weighted method are given. Further, the ratio

$$\Xi = \frac{E_{av\ \mathrm{int}}^{\mathrm{weighted}} - E_{av\ \mathrm{int}}^{\mathrm{unweighted}}}{E_{av\ \mathrm{int}}^{\mathrm{unweighted}}} \tag{19}$$

is provided. This ratio shows the improvement on $E_{av\ \mathrm{int}}$ using the weighted method instead of the unweighted one. A value of $\Xi < 0\%$ indicates that that the weighted method had better performance compared to the unweighted method for the interval of time considered. Further, the index Ξ shows how large the improvement was. The opposite occurs for $\Xi > 0\%$.

The table show also the metric γ that has been introduced in Section 3.4.2 for evaluating the accuracy of the training process. Observing this table allows the types of analysis listed.

- Comparing the performance of the base case strategy for tuning k and m with the new strategies explained in Section 3.4.1
- Comparing the performance obtained setting a constant value of k and setting a constant value of $D_{\max}$
- Comparing the results obtained using the trained parameters and the ones obtained with the optimal parameters for the test interval
- Comparing the performance of the weighted and unweighted method
- Analysing the effect of a change of the disturbance pattern on the performance
- Observing the behaviour of the k-nearest neighbour method with the linear AR model

5.2.3. Comparison between the base strategy and training the tuning parameters

The impact of the choice of the parameters depends on the predictability of the signal. If the signal is highly predictable, then the strategies explained in Section 3.4.1 has a high impact on the quality of the prediction compared to the base strategy. Indeed, Tag 34 is highly predictable and one can obtain very good results by careful tuning. Tags 33 and 30 are less inherently predictable, but also robust to tuning parameters.

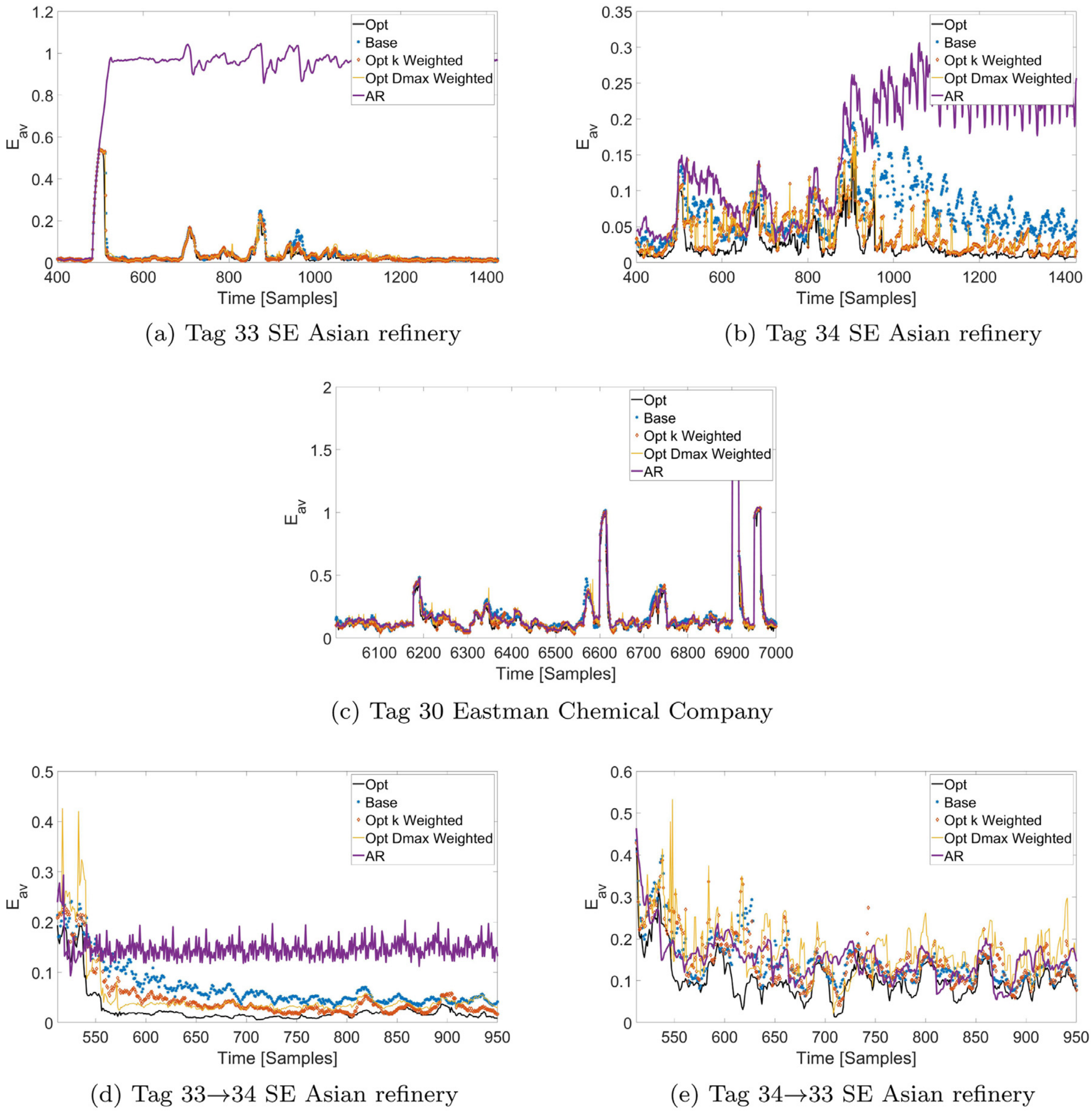
5.2.4. Comparison between a fixed number of neighbours versus a maximal neighbours distance

The results show that setting a certain value of k instead of a certain value of $D_{\max}$ makes no difference in most of cases or it is even better in some of the cases. The major difference is in the computational time needed to obtain the optimal parameters. The method that uses $D_{\max}$ needs to compute far more combinations of the parameters for the search grid algorithm. Therefore, the authors would not recommend this approach.

5.2.5. Comparison between using the trained parameters and the optimal parameters for the test interval

For Tag 33, 34 and 30, where the pattern in the time series is repeating itself, there is not a significant difference of performance between using the parameters obtained from the training of the algorithm and best results obtainable for the interval considered. Table 3 shows indeed high values of γ for these Tags. Consequently the conducted training has been enough to nearly reach the optimal performances available from the strategy.

Instead, there is a decrease of performance in case of the time series built on purpose Tag 33 → 34 and 34 → 33, especially in the case of Tag 33 → 34. However, γ for the Tag 33 → 34, using the weighted method, is 0.76. In comparison the AR model shows a γ of just 0.33. Therefore the k-nearest neighbour method shows a far higher robustness.



(a) Tag 33 SE Asian refinery

(b) Tag 34 SE Asian refinery

(c) Tag 30 Eastman Chemical Company

(d) Tag 33→34 SE Asian refinery

(e) Tag 34→33 SE Asian refinery

Fig. 5. Optimal distances achievable over time for different strategies.

5.2.6. Comparison between the weighted and unweighted method

The prediction performance obtainable with the unweighted method, if k and m are allowed to change over time, is in some cases slightly better than the performance with the weighted method. This is can be seen from the values $E_{av\ \mathrm{int}}$ corresponding to the strategy 'Opt'. However, the strategy 'Opt' is a term of comparison. It is not possible to know at an instant N what are the optimal parameters to be chosen for the k-nearest neighbour method.

Looking at the ratio Ξ (Eq. (19)) for the different strategies, it is possible to see that the weighted method improves the prediction, except for Tag 30, where the 'Base unweighted' strategy is slightly better than the weighted one.

Therefore, we recommend using the weighted method because it provides generally better results.

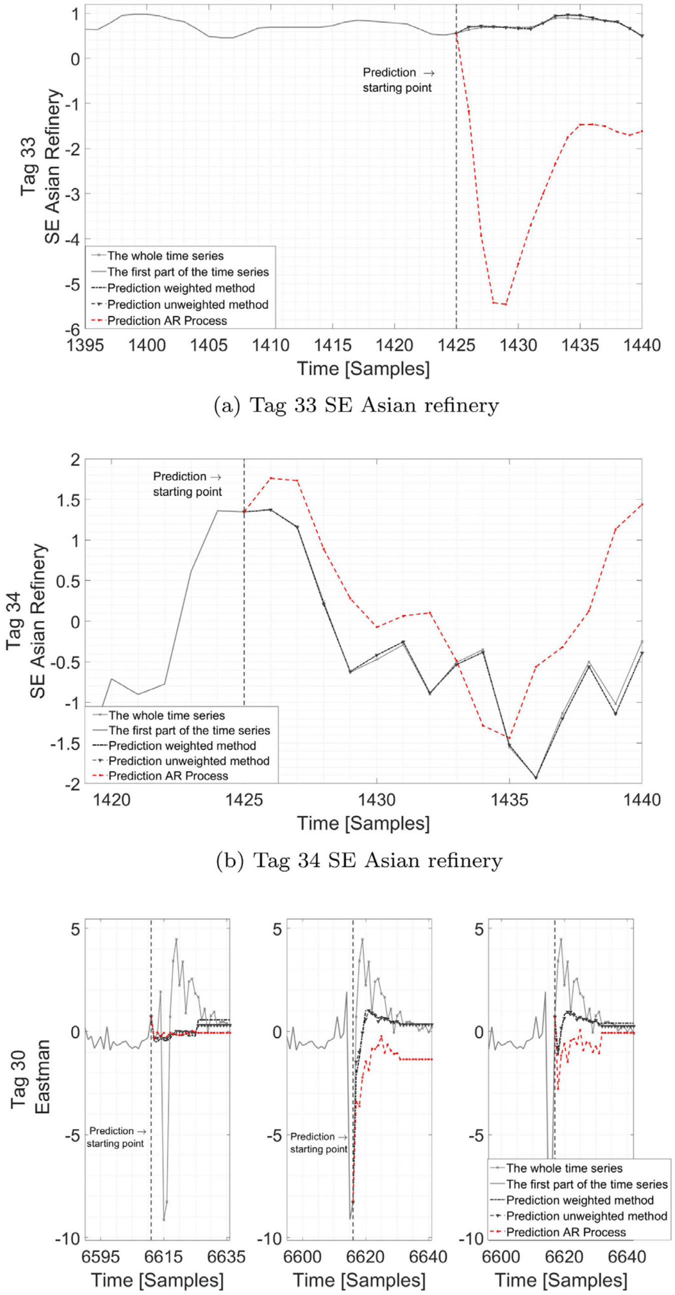
5.2.7. Comparing the behaviour of the k-nearest neighbour method with the linear AR model

Table 3 shows that the values of $E_{av\ \mathrm{in}}$ are far higher for the predictions obtained by the autoregressive model than the k-nearest neighbours for Tags 33, 34 and 33 → 34. Furthermore, the values of γ^{AR} are very low, meaning that a new training of the model would be necessary. The values of γ for the k-nearest neighbours are high instead.

The topic will be described further in the next Section 5.2.8. However, this observation highlights already one of the advantage of the k-nearest neighbours, which is the capability to learn new patterns, without new training and human intervention for the choice of the parameters.

5.2.8. Variation of the values of E_{av} over time

Fig. 5 plots the values of E_{av} over time using the strategies listed in Section 5.2.2. Given the analysis done in Section 5.2.6, only the weighted method is plotted. From Fig. 5, it is visible that the best performance available, indicated as 'OPT', vary over time. Furthermore, some strategies might appear better than other in some intervals of time and worse in other intervals. Therefore, a performance metric that takes an interval of time into account is needed. This is the reason behind the metric $E_{av\ \mathrm{int}}$ explained in Section 3.3.2.



(a) Tag 33 SE Asian refinery

(b) Tag 34 SE Asian refinery

(c) Tag 30 Eastman Chemical Company

Fig. 6. Examples of predictions obtained with the weighted and unweighted k-nearest neighbours methods using the parameters obtained from the training and with the AR model.

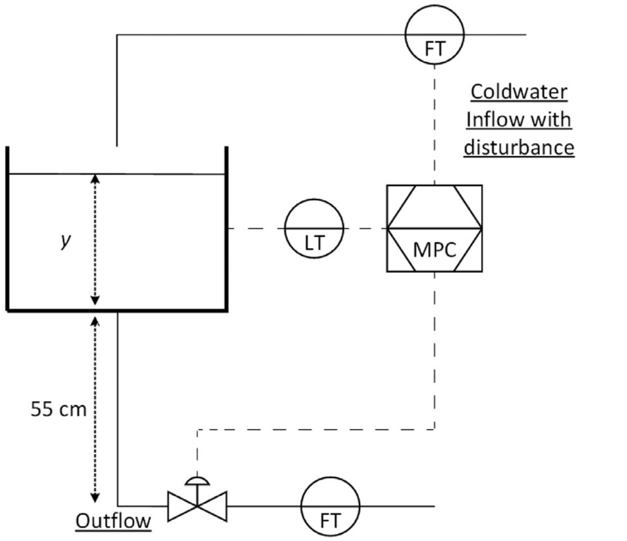


Fig. 7. Case study. The MPC receives measurements of level and of cold water disturbance and sends its output to the outflow valve.
Source: ©IEEE 2018, reprinted with permission from Proceedings of the 26th Mediterranean Conference on Control and Automation (MED 2018), Zadar, Croatia, pp-19–22.

Regarding the k-nearest neighbours, some plots exhibit a peak of E_{av} when the time trend has events or disturbance patterns that do not resemble events present in the memory of the algorithm. This is the case of the step present in the Tag 33 for example (Fig. 5a). This event highlights also the time that the k-nearest neighbours algorithm needs to acquire new data at the new operating condition and be able to provide reliable predictions again.

The AR model instead after a transient is not able to learn a new model unless it is retrained. Consequently an offset might be visible on the value of E_{av} for the AR model. Of course, an AR model is intended for stationary time series, but in process plants it might be common that the time series of the measured disturbances are not stationary and in this case the k-nearest neighbours has an advantage.

Finally, the different strategy overlap for the Tag 30, which is the less predictable of the all time series considered. Instead, for the other Tags 33 and 34, and their composite time series 33 → 34 and 34 → 33, there is a visible distinction between the different strategies. Highly predictable time series benefit more from the improvements in training strategies.

5.3. Examples of prediction at one instant of time

To conclude this Section, Fig. 6 provide examples of possible prediction obtainable using the parameters and the AR model resulting from the training process. For the Tags 33 and 34 the prediction starts at instant $N = 1425$, using a prediction horizon $h = 15$. This instant of time lays in the region where the AR models for the two time series show a deterioration of the prediction performance. For Tag 30 from the Eastman Chemical Company $N = 6615$ and $h = 15$.

As regards the two Tags 33 and 34, the AR model shows qualitatively the deterioration of the performance of the AR model compared to the prediction obtained by the k-nearest neighbour method.

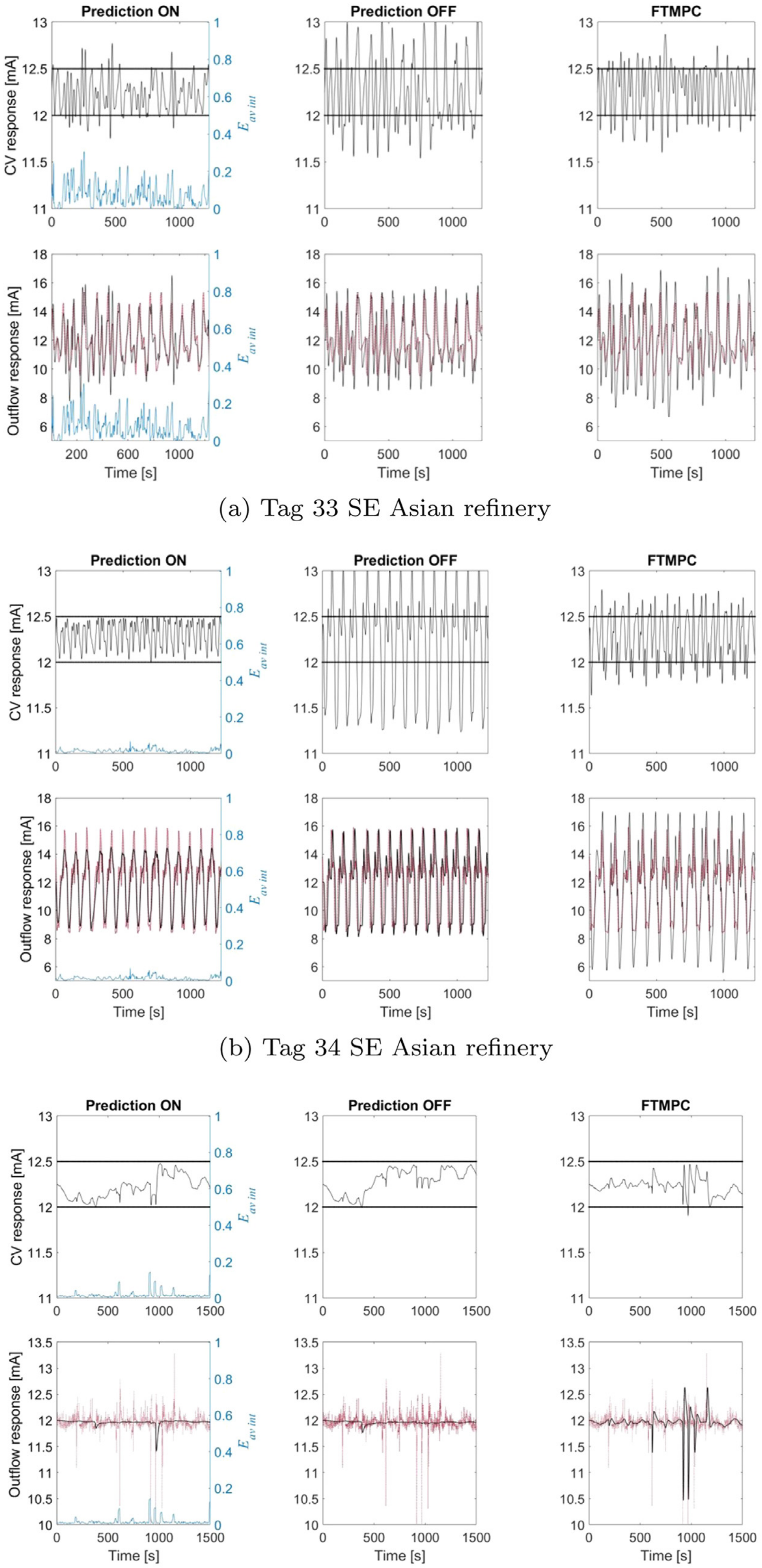
As regards Tag 30, Fig. 6c shows that neither the k-nearest neighbours nor the autoregressive model predict the abrupt disturbance just before it happens. However, once it happens they can forecast the future trend and, nonetheless, the k-nearest neighbours shows better performance.

6. An application of disturbance forecasting in process control

This section provides a simple example of an application of the disturbance forecasting in process control. The model and the structure of the controller is described briefly. Thereafter the results are shown.

6.1. Description of the case study

Fig. 7 shows the considered case study, which is a simulator of a laboratory continuous stirred tank heater



(a) Tag 33 SE Asian refinery

(b) Tag 34 SE Asian refinery

(c) Tag 30 Eastman Chemical Company

Fig. 8. Results of the response of the level (CV) and of the outflow using real disturbances from a SE Asian refinery and from a Eastman Chemical Company plant(the inflow of the tank is dashed).

(zenodo.org/record/888135) with an MPC controller (Borghesan et al., 2018b). The CSTH consists of a simulated tank filled with water, a cold-water inflow and a water outflow. The level is the only controlled variable and the outflow valve of the tank is the only manipulated variable available to the controller. The aim of the controller is to keep the level within two tight constraints despite the variations of the inflow which is affected by a disturbance. The inflow of the tank is considered constant in the nominal case and the disturbance d_{CW}, assumed measurable, is added to the nominal value of the cold water. The outflow should be kept constant as much as possible so that the disturbance is not propagated to a possible second unit connected downstream.

The simple relationship between disturbance, controlled variable and manipulated variable has the aim to highlight how the response of the controller changes by simply providing a disturbance forecast to the controller.

The controller can use the data about $d_{CW}(N)$ in three possible ways:

- Predicting the future values of $d_{CW}(N+h), h \geq 0$ where N is the current discrete instant of time and h is the prediction horizon of the disturbance
- Setting $d_{CW}(N+h) = d_{CW}(N), \forall h \geq 0$. An MPC that uses this strategy is also called Frozen Time MPC (FTMPC) (Cairano et al., 2014)
- Ignoring the presence of the disturbance. This is equivalent to setting $d_{CW}(N+h) = 0, h \geq 0$

In the first case, the algorithm used for the prediction is the weighted algorithm with the parameters k and m obtained from the training process described in Section 3.4.1.

6.2. Results

Fig. 8 shows the trend over time of the controlled variable (level) and outflow from the tank using the MPC strategies described in Section 6.1 for segments of the Tag 33, 34 and 30. The strategies are the MPC capable of predicting the disturbance (Prediction ON), the MPC that ignores the presence of a disturbance in its model (Prediction OFF), and the frozen time MPC (FTMPC).

The lower panels relative to the outflow show also the disturbing inflow with a red dashed line. The panels on the left hand side, corresponding to the strategy ‘Prediction ON’, show also the values of E_{av} over time with a blue line. The real industrial data sets in Fig. 2 have to be adapted for use as disturbances in the CSTH simulation. Tags 33 and 34 are from a refinery and an oscillation period of 17 min. A realistic disturbance for the laboratory-scale CSTH simulator would have much shorter timescales, however. Therefore, the disturbances to be applied are derived from Tags 33 and 34 by using a data set with exactly the same waveform but with a sampling interval of 5 s. Similarly a data set representing abrupt disturbances has been derived from Tag 30 by using a sampling interval of 1 s.

The implementation of the prediction improves the performance of the controller in terms of violation of the constraints and it reduces at the same time the variation of the outlet flow, reducing the disturbance propagation to a possible process unit situated downstream. This result is true in the case of persistent oscillating disturbances as well as in the case of abrupt spiky disturbances.

The benefits are higher for the Tag 34 compared to the Tag 33. The reason is that the predictions that are obtainable for the Tag 34 are more accurate. Indeed, the values of E_{av}, plotted on the panels on the left hand side, are higher for Tag 33 than for Tag 34. This is because of the higher predictability of Tag 34 compared to the one of Tag 33.

Finally, a comparison between the two solutions used normally in industrial MPCs, ‘Prediction OFF’ and ‘FTMPC’, is interesting. The MPC without any information regarding the disturbance value violates the given constraints with both persistent disturbances (Tag 33 and Tag 34). The FTMPC violates less the constraints but its control action is aggressive. Instead, when abrupt spiky disturbances appear the FTMPC violates once the constraints of the controlled variable at around the instant 1000s and moves aggressively the manipulated variable.

7. Conclusions

This paper focuses on the prediction of persistent and abrupt process disturbances. It proposes two versions of the k-nearest neighbours method to predict the future values of process disturbances. The methods have been able to provide good predictions of three signals of real plants, affected by persistent oscillating disturbances or spiky abrupt ones. The paper shows also how to assess the predictability of a time series, analyses the impact of the parameters k, $D_{\max}$ and m on the predictions. The paper provides guidance in their choice and compares different strategies to obtain the parameters. It concludes that the weighted method is more robust, and that using a fixed number of neighbours k in place of a fixed distance $D_{\max}$ does not deteriorate the performances and it is computationally faster. Furthermore, when a strong self predictability is present, an algorithm that optimizes the choices of the tuning parameters is beneficial for the quality of the prediction. On the other hand, when a strong self predictability is not present, the method is robust to a wide range of values of the parameters.

The paper analysis also the effect of changing characteristics of the disturbance in the prediction performance and does a comparison of the k-nearest neighbours method with the predictions obtained with an autoregressive model. This analysis shows that the k-nearest neighbour is better able to learn in an unsupervised way changing characteristics of the disturbance.

Thereafter, this paper presents a simple application of the disturbance prediction in process control, showing how the prediction of the disturbance can work in conjunction with an MPC controller. The aim of the controller is to reduce the variations of the manipulated outflow, keeping the level of the water within two constraints. The considered disturbance are the three real time series mentioned before. For all the time series, of persistent oscillating type as well as of abrupt spiky type, the method provides consistent advantages in terms of reduction of constraint violations and propagation of the disturbance.

References

Birla, N., Swarup, A., 2015. Optimal preview control: a review. Optim. Control Appl. Methods 36 (2), 241–268.

Borghesan, F., Chioua, M., Thornhill, N.F., 2018a. Forecast of persistent disturbances using k-nearest neighbour methods. In: Computer Aided Chemical Engineering, Volume 44, pp. 631–636.

Borghesan, F., Chioua, M., Thornhill, N.F., 2018b. An MPC with disturbance forecasting for the control of the level of a tank with limited buffer capacity. In: 2018 26th Mediterranean Conference on Control and Automation (MED), pp. 1–9.

Cairano, S.D., Bernardini, D., Bemporad, A., Kolmanovsky, I.V., 2014. Stochastic MPC with learning for driver-predictive vehicle control and its application to HEV energy management. IEEE Trans. Control Syst. Technol. 22, 1018–1031.

Camacho, E., Bordons, C., 2008. Model Predicitve Control. Springer Verlag.

De Gooijer, J.G., Hyndman, R.J., 2006. 25 years of time series forecasting. Int. J. Forecast. 22 (3), 443–473.

Desborough, L., Harris, T., 1992. Performance assessment measures for univariate feedback control. Can. J. Chem. Eng. 70 (6), 1186–1197.

Kantz, H., Schreiber, T., 2004. Nonlinear Time Series Analysis, Second ed. Cambridge University Press, Cambridge.

Klenske, E.D., Zeilinger, M.N., Schoelkopf, B., Hennig, P., 2016. Gaussian process based predictive control for periodic error correction. IEEE Trans. Control Syst. Technol. 24 (1), 110–121.

Liptak, B.G., 2005. Feedback and feedforward control. Instrument Engineers' Handbook Volume Two: Process Control and Optimization, Fourth CRC Press. chapter 2

Nunokawa, S., Itoh, T., Suenaga, O., Katayama, T., 1997. Design of temperature control system with preview action for coke oven plant. In: IFAC Proceedings Volumes, 30, pp. 97–102.

Thornhill, N.F., 2005. Finding the source of nonlinearity in a process with plant-wide oscillation. IEEE Trans. Control Syst. Technol. 13 (3), 434–443.

Thornhill, N.F., Cox, J.W., Paulonis, M.A., 2003. Diagnosis of plant-wide oscillation through data-driven analysis and process understanding. Control Eng. Pract. 11, 1481–1490.

Thornhill, N.F., Oettinger, M., Fedenczuk, P., 1999. Refinery-wide control loop performance assessment. J. Process Control 9, 109–124.

Yoshitani, N., Hasegawa, A., 1998. Model-based control of strip temperature for the heating furnace in continuous annealing. IEEE Trans. Control Syst. Technol. 6 (2), 146–156.

Yuan, T., Qin, S.J., 2014. Root cause diagnosis of plant-wide oscillations using Granger causality. J. Process Control 24 (2), 450–459.