An online outlier detection method based on wavelet technique and robust RBF network

Weixing Su¹, Yunlong Zhu¹, Fang Liu² and Kunyuan Hu¹

Abstract
We focus on the issue of outlier detection for time-series data in a process control system (PCS), since outlier detection is a critical step before performing data-based system analysis. Several published articles have proved that a wavelet transform (WT) technique can be used to detect outliers in time-series data, but the standard WT detection method, as well as any other univariate outlier detection technique, does not distinguish between the sudden change caused by the changes of inputs and the fluctuations caused by outliers in PCS. In order to improve this shortcoming of the conventional WT method for the data in a PCS, a new algorithm combining the wavelet technique with a robust radial basis function (RBF) network is proposed here. In this method, a robust RBF network (RBFN) training algorithm is proposed, which can train the RBFN online using the original data as a training set without the need of clean data and thus fits the application of online detection. Furthermore, a hidden Markov model is adopted as an analysis tool to accomplish online automatic detection without pre-selecting the threshold. We compare the performance of our proposed method with the conventional wavelet method and the AR model method to demonstrate its validity through simulation and experimental applications to the data pretreatment process in an electric arc furnace electrode regulator system.

Keywords
Outlier detection, process control system, RBFN, time series, wavelet

Introduction
For most process control systems (PCSs), nearly all the online data-based system analysis methods, such as online parameter estimation (Samantaray et al., 2010; Young and Chotai, 2001), online system monitoring (Smith et al., 2008) and online control strategy making (Miller and Pankov, 2007; Zhang et al., 2010), need ‘clean’ data to provide reliable and useful information. It is futile to perform data-based analysis when data are contaminated with outliers because the outliers can lead to model misspecification, biased parameter estimation and incorrect control command. So data preprocessing is a necessary prerequisite step prior to performing data-based system analysis.

In a PCS, the outliers are defined as the input–output time series that does not meet the input–output model of controlled object. They may be generated by a different mechanism corresponding to the normal data and may be due to some sensor faults, process disturbances and/or instrument degradation. In this paper, we address the problem of detecting outliers sampled from the PCS. More accurately speaking, we are concerned with the data sampled from the regulating system, which have a fixed structure and some slow time-varying parameters. These data have their own characteristics. The first is a non-stationary property, which means the data are seriously oscillating during the regulation process, and because of some slow time-varying parameters in PCS, the statistical quantities of the data, such as the mean and the variance, are changing continuously. The second is a real-time property, which attributes to the short control cycle and the long control process. Since most PCSs have two such characteristics described above, it is quite natural and necessary to construct an effective outlier detection method with good dynamic characteristics and excellent real-time properties.

The task of outlier detection has been tackled by several researchers in a variety of ways, such as the statistical (Barnet and Lewis, 1994), deviation-based (Takeuchi and Yamanishi, 2006), distance-based (Knorr and Ng, 1998, 1999) and density-based approach (Breunig et al., 2000). The statistical approach requires *a priori* knowledge of the probability distribution of the data. A test is then used to verify whether an object (data at time *t*) is significantly larger (or smaller) in relation to the distribution of the data. A test is then used to verify whether an object (data at time *t*) is significantly larger (or smaller) in relation to the distribution of the data. A test is then used to verify whether an object (data at time *t*) is significantly larger (or smaller) in relation to the distribution of the data. A test is then used to verify whether an object (data at time *t*) is significantly larger (or smaller) in relation to the distribution of the data. A test is then used to verify whether an object (data at time *t*) is significantly larger (or smaller) in relation to the distribution of the data.

¹Institute of Automation, Chinese Academy of Sciences, Laboratory of Information Service and Intelligent Control, Shenyang, China
²Northeastern University, School of Information Science and Engineering, Shenyang, China

Corresponding author:
Weixing Su, No. 114 Nana Street, Shenyang, 110016, China.
Email: suweixing@sia.cn
approach proposed by Knorr and Ng (1999) calculates the distance between the point and its neighbours, and considers the point an outlier if its nearest neighbours are sufficiently distant. The density-based approach can be seen as a new kind of distance-based approach. Breunig et al. (2000) proposed a density-based outlier detection approach by evaluating the degree of isolation the object has with respect to the surrounding neighbourhood. According to the attributes of each classic outlier detection approach and the characteristics of the time series in the PCS, it is unreliable to detect outliers in a PCS based only on the distances, the densities, the means and/or the variances of the data. Recently, some new ideas have been introduced to the outlier detection field, such as the cluster analysis method (Almeida et al., 2007; Wang and Chiang, 2008), the support vector machine (SVM) method (Miao et al., 2009), the artificial neural network (ANN) methods (Hawkins et al., 2002; Yang et al., 2009) and the Bayesian network (BN) method (Janakiram et al., 2006). In practice, the cluster analysis method often suffers from the choice of an appropriate parameter of cluster width; the SVM method identifies outliers from the data measurements collected after a long time window and is not performed in real time; the exiting ANN methods require clean sample data for training but here we cannot get any; the BN method needs the accurate prior knowledge while it is impossible for the PCS. In other words, most existing outlier detection methods based on the above ideas are essentially for offline operation and it is generally hard to filter outliers and keep a track of a changing process model simultaneously.

The wavelet analysis technique has been widely used for outlier detection recently due to its inherent time-frequency property that allows splitting signals into different components at several frequencies, and in the work of Mallat and Hwang (1992), the WT technique is applied to detect outliers for the first time, in which the large wavelet coefficients characterize non-stationary transients can be considered outliers from a statistical point. So, this method is particularly adapted to non-stationary transient extraction from stationary. However, the recognition of outliers in PCS cannot simply depend on whether the data are stationary or not because the normal signal, which is undesired to be detected out as outlier, may also be non-stationary during the regulation process in the PCS.

In this paper, considering the limitations of the WT technique and the characteristics of the data in PCS, an online outlier detection method based on a robust radial basis function network (RBFN) and iterative wavelet is proposed. In the proposed method, our contributions are shown as follows:

1) **The robust RBFN training algorithm, which can model the time-series online without the need for a clean training sample data set, is proposed in this paper.** Unlike the conventional RBFN, a robust RBFN can work without clean mechanisms. In other words, the outliers need not be excluded from the training set since model estimation with missing data maybe lead to poor accuracy, so a robust training algorithm based on a weighted objective function is proposed to resist the disturbance of outliers.

2) **The shortcoming of the conventional WT technique in the outlier detection field is overcome by introducing the robust RBFN model.** In the next section, the shortcoming of the conventional WT technique is pointed out by an example. As well as any other univariate outlier detection technique, the shortcoming is that the standard wavelet detection method does not distinguish between the sudden change caused by the changes of inputs and the fluctuations caused by outliers in PCS. Because of this, we built the model of the data first using the proposed robust RBFN, and then the iterative wavelet technique is used to analyse the estimation residual in order to separate a subset of (small) noise-representative wavelet coefficients from the (large) outliers (one must notice that this method assumed that the estimation residual is white noise when the datum is normal). These improvements on the conventional WT method can further motivate the wavelet technique towards its use as a new and general outlier detection method.

3) **The hidden Markov model (HMM) is introduced as a tool to analyse the wavelet coefficients.** The reason we used HMM here is that we want to find a detection algorithm, which can detect outliers online and without pre-selecting the threshold. The double-chain structure of the HMM just can do it. In the HMM-based detection method, the state hidden chain is used to denote the detection results of the data (two states represent outlier and normal data respectively), and the Viterbi algorithm is used to obtain the optimal state of hidden chain.

The paper is organized as follows: the conventional wavelet detection principle and the existing problem for data in PCS are illuminated below. This is followed by the introduction of the proposed detection method including a robust RBFN, the iterative wavelet and the HMM-based analysis method. Then, its performance and comparison results with conventional wavelet are analysed, followed by the concluding remarks.

**The conventional wavelet detection principle and the existing shortcoming for time series of PCS**

The theorem proposed by Mallat and Hwang (1992) characterizes a particular class of isolated outliers based on the behaviour of the wavelet transform modulus maxima (WTMM). Mallat and Hwang (1992)’s theorem: let \( f(x) \) be a tempered distribution whose wavelet transform (WT) is well defined over \( [a, b] \), and let \( x_0 \in [a, b] \). We suppose that there exists a scale \( s_0 > 0 \), and a constant \( C \), such that for \( x \in [a, b] \) and \( s < s_0 \), all the modulus maxima of \( W_f(s, x) \) belong to a cone defined by

\[
|x - x_0| \leq C s
\]  

(1)

Then, for any point \( x_1 \in [a, b] \), \( x_1 \neq x_0 \), \( f(x) \) is the uniform Lipschitz \( n \) in a neighbourhood of \( x_1 \). Let \( \alpha < n \) be a non-integer. The function \( f(x) \) is the Lipschitz \( \alpha \) at \( x_0 \), if and only
if there exists a constant $A$ at each modulus maxima $(s, x)$ in the cone defined by Equation (1)

$$|W_f(s, x)| \leq A s^\alpha$$

(2)

The theorem of WTMM mentioned above shows that there are some relationships between wavelet coefficients and the Lipschitz $\alpha$: when $\alpha > 0$, the amplitude of WTMM point enlarges along with the increasing of scaling $s$; when $\alpha < 0$, it decreases along with the increasing of scaling $s$; when the $\alpha = 0$, it does not change with scaling $s$. So the conventional wavelet-based outlier detection theorem is based on these relationships and can be described as: the Lipschitz of outlier is smaller than 0 and the wavelet coefficient of outlier will become larger with the decreasing of scaling s.

However, as for the seriously oscillating signals, just as the oscillation in the regulation process of PCS, the conventional wavelet-based outlier detection theorem is no longer adapted. The main reason can be summarized as: for the oscillation in the regulation process of PCS, the normal data have the same Lipschitz with outlier and it is smaller than 0, and the conventional wavelet-based outlier detection method mentioned above cannot distinguish outliers at all. In Figure 1, we present some samples of the time series obtained from the electric arc furnace electrode regulator system with some outliers in it. Although the introduced outliers are apparent enough and can even be distinguished by the naked eye, it is meaningful to provide some objective criteria, which are suitable for application automatically by the computer.

Figure 1(a) shows the 500 sample data to be detected. The first 50 data represent the damped oscillation process of PCS. After this process, eight outliers are included in the last sample data. Figure 1(b) shows the wavelet coefficients of sample data at a different frequency (with the dashed line indicating a frequency of 40 and the solid line represents a frequency of 30). As Figure 1(b) shows, both in the damped oscillation process and at the occurrence of the outliers, the wavelet coefficients keep reducing along with the increasing of scaling. This phenomenon illustrates that the conventional wavelet-based outlier detection method cannot perform outlier detection for the process data of the PCS.

Furthermore, in the process of WT, the time window and the scale window are coupled with each other; each of them cannot reach a high resolution in the meantime. This means that when the scale decreases (equivalent to the frequency increasing), the shocks near the outliers will become more severe, and seriously affect the accurate positioning of the outliers. This problem is illustrated by Figure 2. Figure 2(a) is the detecting data, Figures 2(b) and (c) are the wavelet decompositions of the data under different frequencies ($f = 80$ in Figure 2b and $f = 20$ in Figure 2c). It can be clearly seen from these two figures that when the scale is smaller (with $s = 1/f = 1/80$), the outlier cannot be positioned accurately with the influence of shocks; when the scale becomes larger (with $s = 1/f = 1/20$), the shocks near the outliers are no longer apparent but the curve is similar to the curve of the original data, which cannot simply be used a threshold value to determine the location of the outliers.

Therefore, it can be concluded from the above two examples that, for the process data in the PCS, the conventional WTMM detection principle is not applicable.

The new outlier detection method for the data in PCS

In view of the shortcoming of the conventional wavelet technique, we propose a new outlier detection method for the data

![Figure 1](image-url)
in a PCS. The detection process of the method is described in Figure 3.

First, the estimates of the data to be detected are calculated by the robust radial basis function (RBF) model and the fitting residuals are obtained after that. These residuals are white noise when the data are normal, and the sum of white noises and the fitting residuals when the data are outliers. For the oscillation process of the PCS, since the sudden shifts caused by change of inputs meets the model of the controlled object, the residuals are also white noises. In this way, the interference of damped oscillation to the outlier detection is eliminated and thus the shortcoming of the conventional wavelet-based technique mentioned previously is overcome when solving the wavelet coefficients using wavelet decomposition. Finally, HMM is used to analyse wavelet coefficients and obtain the detection result that is the abnormal degree of the data. This degree is also used to determine the role of the data in the next step, updating of the robust RBF model. It means that the data with great degree of abnormality will play less important role in the next update of the robust RBF model and the influence of outliers for model updating is restrained to the greatest extent.

The robust RBFN training algorithm

Naturally, the training of the RBFN used clean sample data, but in the actual industrial control process, obtaining clean data directly from the control system is impossible (which is the purpose of our research – getting clean data by outlier detection), so we have no choice but to use data contaminated by noise and outliers to perform mathematical modelling before the detection. On the other hand, removing the outliers from the modelling dataset is not rational either, since model estimation with missing data might lead to a biased model.
However, the outliers will no doubt also influence the accuracy of the model, so it becomes very attractive to eliminate this impact on the modelling effect.

In this section, a robust training algorithm named the weighted RBFN training algorithm, which is an improvement on the conventional RBFN training algorithm, is proposed to solve the above-mentioned problem. The basic idea of the algorithm is this: the data detected are used as training data to update the RBFN online in real time, then the new RBF model is used to estimate the data to be detected the next time, because the training data are the data already detected and the testing data are the data to be detected.

The conventional RBFN training algorithm. Given the controller input signal $u_t$ at time $t$ and the output signal $y_t$ of the generalized controlled object, the process data at time $t$ is expressed as $[u_t, y_t]$. Supposing that the PCS is a $P$ order system (it means that the output of the system $y_t$ is related to $P$ process data before time $t$), the input of conventional RBFN at time $t$ should be $x_t = [u_{t-P}, y_{t-P}, \ldots, y_{t-1}, y_t]$ and the output should be the estimate of actual output $y_t$ of the control system at time $t$.

The conventional RBFN online training algorithm (Kurban and Besdok, 2009) is based on the following objective function:

$$\min \sum_{i=1}^{t} r^{t-i} \cdot [y_t - \hat{y}_t]^2$$

with forgetting factor $r$. Here, we use $\varphi(\cdot): R^d \rightarrow R^l$ to denote a RBF, which maps the input space into a higher dimensional space. The vector $w \in R^l$ represents the weight vector in the output layer. The symbol $e_t \in R$ represents the estimated error. Then, we write the expression of the output $y_t$ as:

$$y_t = w e(x_t) + e(t)$$

Thus the conventional online RBFN training algorithm can be shown as below:

**Step 1.** Initialize the variables $C_0^j$ and $D_0^j$ ($C_j$ is the covariance of hidden layer output and the PCS output at time $t$; $D_j^j$ is the output covariance of hidden layer node $i$ and node $j$ at time $t$). Here $C_0^j$ and $D_0^j$ are the initial values of $C_j$ and $D_j^j$, and each of them equals 0), the superscript represents the time and the subscript $i = 1, \ldots, d$; $j = 1, \ldots, d$; each of them denotes the number of the hidden nodes.

**Step 2.** Given the input data $x_t$ at time $t$, compute the output vector of the hidden layer $\varphi(x_t) = [\varphi_1, \ldots, \varphi_d]$.

**Step 3.** Update the variables $C_j$ and $D_j^j$ by Equation (5).

$$C_j = (1-r)C_j^{t-1} + r \cdot y_t \cdot y_t^T$$

$$D_j^j = (1-r)D_j^{t-1} + r \cdot \varphi_i \cdot \varphi_j$$

**Step 4.** Solve the equations $C_j = \sum_{i=1}^{d} w_i D_j^i$ ($j = 1, \ldots, d$) and obtaining the output weight $w \in R^l$ at time $t$, where $D_j^j = D_j^j$.

It can be easily seen that if the sample $[u_t, y_t]$ at time $t$ is an outlier, its corresponding $C_j$ and $D_j^j$, $i = 1, \ldots, d$, $j = 1, \ldots, d$ may be seriously biased, which will result in the negative impact on the convergence of other samples and finally lead the training into fiasco. Therefore, when outliers exist, the training process should manage to avoid the influence that comes from the outliers.

Weighted RBFN training algorithm. An effective way of preventing the prediction error of the learned model, which is caused by the outlier, is to impose a weight on the prediction error of the outlier. Thus, the weighted objective function is

$$\min \sum_{i=1}^{t} r^{t-i} \cdot b_i \cdot [y_t - \hat{y}_t]^2$$

where $b_i$ is used to measure the abnormal degree of the data and thus judge whether outliers happen; $b_{10}$ is calculated by Equation (18). Introducing $b_i$ into Equation (5) at step 3, the equation becomes:

$$C_j = (1-r)C_j^{t-1} + r \cdot b_i \cdot y_t \cdot y_t^T$$

$$D_j^j = (1-r)D_j^{t-1} + r \cdot b_i \cdot \varphi_i \cdot \varphi_j$$

Observing Equation (7), we can find that the weight $b_i$ seems a penalty factor, which makes $C_j$ and $D_j^j$ not seriously biased when $[u_t, y_t]$ is an outlier. However, unlike the conventional penalty factor, here $b_i$ is time-varying according to the intensity of an outlier. So, the weight here is more reasonable and flexible. Because the weights for the normal samples can be fixed at 1 as shown in Equation (6), the learning process under the normal samples in the proposed weighted robust training algorithm is the same as that in the conventional training algorithm. The value of forgetting factor $r$ is in $[0, 1]$ and determined by the time-varying speed of the parameter of controlled object. When the speed is faster, $r$ will become smaller, and vice versa.

Moreover, here we adapt to the fuzzy partition of the input space and the minimum distance criterion, which was proposed by Alexandridis et al. (2003), to determine the centre vector of each radial basis function. In the algorithm, the hidden layer nodes of RBFN are automatically determined and the centre vector is also automatically updated, but the process of these need large amount of calculation and data cache. So in order to meet the online detection, we have made a simplification to the algorithm of Alexandridis et al. (2003), i.e. determine the hidden nodes offline (by using Alexandridis et al.’s algorithm), then fix the hidden nodes and only dynamically update the centre vector during the online detection process. In this way, the purpose of reducing the computation is achieved.
The iterative wavelet transform technique

It is known that the conventional wavelet decomposition contains the convolution, which needs a large amount of computation and does not fit for online application. To meet the demand of online outlier detection, an iterative WT proposed by Zhang et al. (1998) is considered in this section, by which the wavelet is expressed as an exponential form, and the wavelet decomposition can be calculated by iteration. This improved recursive WT (IRWT) is presented to meet the real-time demands of analysing the process data in PCS.

Defining a function

\[ \Psi_1(t) = \left( \frac{\sigma r^3}{3} - \frac{\sigma r^4}{6} + \frac{\sigma r^5}{15}\right)e^{-(\sigma + i \omega_0)t}u_0(t) \]  

(8)

Set \( \Psi(t) = \Psi_1^*(-t) \) as the fundamental wave and the function of it is

\[ \Psi(t) = \left( \frac{\sigma r^3}{3} - \frac{\sigma r^4}{6} + \frac{\sigma r^5}{15}\right)e^{(\sigma + i \omega_0)u_0(-t)} \]  

(9)

where * denotes conjugate, \( u_0(t) = \begin{cases} 0, & t < 0 \\ 1, & t \geq 0 \end{cases} \). Here we set \( \sigma = 2\pi/\sqrt{3}, \omega_0 = 2\pi \) to ensure that the basic wavelet meets the admissibility condition. According to the definition of the WT, because \( \omega_0 = 2\pi \), the imaginary part is exactly equal to zero and thus only left the real part, the wavelet coefficients of the residual data obtained from the robust RBFN, can be calculated by the following equation:

\[ W_f(kT, f) = \sqrt{f} \sum_{n=1}^{\infty} \varepsilon(nT)\Psi_1^*(f(nT-kT)) \]

\[ = \sqrt{f} \sum_{n=1}^{\infty} \varepsilon(nT)\Psi_1(f(kT-nT)) \]  

(10)

where \( T \) is the sample period, \( k \) and \( n \) are integers, \( f = 1/s \) and \( \varepsilon(nT) \) is the fitting residual obtained from the robust RBFN. Equation (10) is expressed in convolution as:

\[ W_f(kT, f) = T \sqrt{f} \varepsilon(\cdot) * \Psi_1(\cdot) \]  

(11)

Taking the Z-transform, the expression above becomes

\[ W_f(Z) = T \sqrt{f} \varepsilon(Z) * \Psi_1(Z) \]  

(12)

\( \Psi_1(Z) \) in Equation (12) can be easily calculated and expressed as follows:

\[ \Psi_1(Z) = \frac{\delta_1 Z^{-1} + \delta_2 Z^{-2} + \delta_3 Z^{-3} + \delta_4 Z^{-4} + \delta_5 Z^{-5}}{\lambda_1 Z^{-1} + \lambda_2 Z^{-2} + \lambda_3 Z^{-3} + \lambda_4 Z^{-4} + \lambda_5 Z^{-5}} \]  

(13)

where,

\[ c = e^{-\sigma T (\sigma + i \omega_0)} \]

\[ \delta_1 = \left( \frac{\sigma r^3}{3} - \frac{\sigma r^4}{6} + \frac{\sigma r^5}{15}\right) \cdot c \]

\[ \delta_2 = \left( \frac{\sigma r^3}{3} - \frac{\sigma r^4}{6} + \frac{\sigma r^5}{15}\right) \cdot 2 - \left( \frac{\sigma r^3}{3} - \frac{\sigma r^4}{6} + \frac{\sigma r^5}{15}\right) \cdot 5 + \left( \frac{\sigma r^3}{3} - \frac{\sigma r^4}{6} + \frac{\sigma r^5}{15}\right) \cdot 26 \cdot c^2 \]

\[ \delta_3 = \left( \frac{\sigma r^3}{3} - \frac{\sigma r^4}{6} + \frac{\sigma r^5}{15}\right) \cdot (-6) + \left( \frac{\sigma r^3}{3} - \frac{\sigma r^4}{6} + \frac{\sigma r^5}{15}\right) \cdot 22 \cdot c^3 \]

\[ \delta_4 = \left( \frac{\sigma r^3}{3} - \frac{\sigma r^4}{6} + \frac{\sigma r^5}{15}\right) \cdot 2 + \left( \frac{\sigma r^3}{3} - \frac{\sigma r^4}{6} + \frac{\sigma r^5}{15}\right) \cdot 5 + \left( \frac{\sigma r^3}{3} - \frac{\sigma r^4}{6} + \frac{\sigma r^5}{15}\right) \cdot 26 \cdot c^4 \]

\[ \delta_5 = \left( \frac{\sigma r^3}{3} - \frac{\sigma r^4}{6} + \frac{\sigma r^5}{15}\right) \cdot 6 \]

\[ \lambda_1 = -6c, \lambda_2 = 15c^2, \lambda_3 = -20c^3 \]

\[ \lambda_4 = 15c^4, \lambda_5 = -6c, \lambda_6 = c^6 \]

Thus, Equations (12) and (13) lead to

\[ W_f(Z) = \sqrt{f} T \varepsilon(Z) \cdot (\delta_1 Z^{-1} + \delta_2 Z^{-2} + \delta_3 Z^{-3} + \delta_4 Z^{-4} + \delta_5 Z^{-5}) \]  

(14)

According to the displacement characteristics of the Z-transform, we obtain

\[ W_f(kT, f) = \sqrt{f} T \varepsilon((k-1)T, f) + \delta_1 \varepsilon((k-2)T, f) + \delta_2 \varepsilon((k-3)T, f) + \delta_3 \varepsilon((k-4)T, f) - \lambda_1 W_f((k-1)T, f) - \lambda_2 W_f((k-2)T, f) - \lambda_3 W_f((k-3)T, f) - \lambda_4 W_f((k-4)T, f) - \lambda_5 W_f((k-5)T, f) - \lambda_6 W_f((k-6)T, f) \]  

(15)

Equation (15) is the expression of iterative WT. It only utilizes the historical data. Thus the wavelet coefficients can be implemented in real time.

HMM detection method

As far as we know, many existing outlier detection methods need a pre-selected threshold. These methods identify outliers by examining whether some measurement is bigger (or smaller) than the threshold. However, in practice, it is hard to know the exact threshold before detection and thus nearly impossible to make detection work accurate and robust. Aiming at this problem, an analysis method based on HMM without a pre-selected threshold is proposed in this section.

We now formally define the elements of an HMM, and explain how an HMM analyses the wavelet coefficients. An HMM is characterized by the following (Rabiner, 1989; Bilmes, 2006):

1) \( N \), the number of the states in the HMM. Here \( N = 2 \), which means there are two states: outlier and normal.

2) The state transition probability distribution is \( A = (a_{ij})_{N \times N} \), where

\[ a_{ij} = P(S_i = j | S_{i-1} = i) \quad i, j \in S \]  

(16)

with \( S_i \) denoting the state at time \( t \), and \( S = \{0, 1\} \) representing the set of all the states, where ‘0’
represents an outlier and ‘1’ is considered normal. So 
\(a_i\) is the probability that the state for the previous
time is \(i\) while state for the later time is \(j\), and thus the
online updating method of \(A\) is
\[
\begin{align*}
\hat{a}_{01} &= \frac{N(a_{01})}{N(a_{01} + a_{00})}, \\
\hat{a}_{11} &= \frac{N(a_{11})}{N(a_{11} + a_{10})}, \\
\hat{a}_{00} = 1 - \hat{a}_{11}, \\
\hat{a}_{10} = 1 - \hat{a}_{11}
\end{align*}
\]
(17)
where \(N(a_{mn}) = \sum_{i=1}^{t} u_{i}, u_{i} = \begin{cases} 1, & \text{if } S_{i} = m \text{ and } S_{i-1} = m \\ 0, & \text{otherwise} \end{cases} \)

Taking \(N(a_{01})\) for example, \(N(a_{01}) = \sum_{i=1}^{t} u_{i} = \begin{cases} 1, & \text{if } S_{i} = 1 \text{ and } S_{i-1} = 0 \\ 0, & \text{otherwise} \end{cases} \), so \(N(a_{01}) + N(a_{00}) = N(a_{01}) + N(a_{00})\). With time \(t\) increasing, \(N(a_{ij})\) and \(A = (a_{ij})_{N \times N}\) can update constantly.

3) The observation symbol probability distribution is
\(B = (b_{tj})_{1 \times N}, k = S_{j}, S_{j} \in S_{i},\) where
\[
\begin{align*}
b_{01} &= \exp \left( -\frac{1}{2} (W_{t}(t, f) - W_{ave})^{2} W_{var}^{-1} \right) \\
b_{00} = 1 - b_{01}
\end{align*}
\]
(18)
with the variance of wavelet coefficients \(W_{var}\), and the mean of wavelet coefficients \(W_{ave}\), which is equal to zero since the signal analysed by the wavelet is white noise when the data are normal, \(W_{t}(t, f)\) is equal to \(W_{t}(kT, f)\), which means that \(t = kT, k = 1, 2, \ldots\). Here the expression of \(b_{01}\) in Equation (18) is a normalized result of the Gaussian probability density function \(N(W_{t}(t, f)|W_{ave}, W_{var})\). In addition, the \(b_{01}\) in Equation (6) can be computed by Equation (18), which means that \(b_{01}\) is equal to the observation symbol probability \(b_{01}\) in which \(r\) and \(i\) denote the same time. In this way, the detection results of HMM influence the residue estimation by giving a different weight coefficient \(b_{01}\) in Equation (6) during the training of the RBFN, which means that the detected outliers will be have a lighter weight and will not cause too much impact on the accuracy of the RBFN training. The update process of \(W_{var}\) is as follows: at the beginning, \(W_{var}\) is set an initial value. After a detection process, if the result is that the data is normal, \(W_{var}\) is updated by the function \(W_{var} = (1 - r)W_{var}^{-1} + rW_{t}(t, f)^{2}\), where \(W_{var}^{-1}\) is the \(W_{var}\) at time \(t\); conversely, if the result shows that the data is an outlier, the value of \(W_{var}\) is not changed.

4) The initial state distribution \(\pi = (\pi_{i}), \ i \in S_{i}\) is the initial value of state transition matrix \(A = (a_{ij})_{N \times N}\). In this paper, we set \(\pi_{i} = 0.5\) for all \(i \in S_{i}\). In fact, the initial state distribution can be chosen randomly or on the basis of any available model, which is appropriate to the data. With the passage of time, \(A = (a_{ij})_{N \times N}\) would be constantly updated and its value will be closer and closer to the true distribution of the abnormal and normal situations in the data to be detected. Anyway, it must meet \(\sum_{i=1}^{N} \pi_{i} = 1, \ \pi_{i} \geq 0\).

Given the above illustrations about the parameters of an HMM, we can use this HMM as an analysis tool to detect outliers. Here, we see the detection process as one of the three basic problems of an HMM (Bilmes, 2006). This problem is that given an observation sequence and a model, how do we choose a corresponding state sequence? A technique based on dynamic programming methods is proposed in Lou (1995), and is called the real-time Viterbi algorithm.

The algorithm is
\[
\begin{align*}
\phi_{t}(1) &= a_{11} \cdot b_{11}; \\
\phi_{t}(0) &= a_{00} \cdot b_{00} = a_{00} \cdot (1 - b_{11})
\end{align*}
\]
(19)
The symbols \(\phi_{t}(1)\) and \(\phi_{t}(0)\) represent the detective measurement for normal data and outlier, respectively, and produce Equation (20).
\[
\begin{align*}
\phi_{t}(1) &\geq \phi_{t}(0), \ S_{t} = 1 \\
\phi_{t}(1) &< \phi_{t}(0), \ S_{t} = 0
\end{align*}
\]
(20)

Experimentation and application

Experimentation and comparison

We prepared two kinds of data sets: 1) generated by the mechanism model of the electric arc furnace electrode regulator system in order to simulate the real PCS, and 2) generated by a time-varying model improved from the model of Alexandridis et al. (2003).

The first data set (see Figure 4a) is a data sequence with 10% white noise. As Figure 4(a) shows, at the beginning there is a segment of data representing the controller regulating process (from one step to 50 steps). This segment of data are normally oscillations; following them are 14 outliers. These outliers are added artificially according to these principles: the amplitudes of the outliers added in the signals are neither larger than the biggest value of the signals, so that they are easily detected, nor smaller than the amplitude of the white noise as to be submerged by noise.

Figures 4(b) and (c) show the residual errors calculated by the robust RBFN and the wavelet coefficients of the residual errors (scale \(s = 1/80\), respectively. The robust RBFN used here adopts the robust modelling approach proposed in this paper, which updates the model online using the data of previous time (having been detected), and the hidden layer nodes number of the robust RBFN is 6. Figure 4(d) shows the detection results using our proposed approach. The horizontal axis shows the the time while the vertical axis shows the states of the HMM (‘1’denotes the normal data and ‘0’ is an outlier). It can be observed from Figure 4(d) that our proposed detection method is able to detect all of outliers in the data set.

In order to illustrate the superiority of the proposed method, a comparison with the autoregressive (AR) model detection method proposed by Takeuchi and Yamanishi (2006) is made. Figure 5 shows the detection results of the first set of data using the AR model detection method.

In Figure 5, the vertical axis denotes the negative logarithm of the Gaussian probability of the residual, and the residual is obtained from the AR model fitting for the data to be detected.
A higher value of the negative logarithm means the likelihood of outliers is greater. It can be seen from Figure 5 that the shock process signals at the beginning are mistakenly detected as outliers and the second outlier (marked by an oval) is almost not detected. These results indicate that the outlier detection method based on the AR model is not ideal for control process data and its detection results are worse than the proposed method.

In order to verify the repeatability of the algorithm, two datasets containing 1000 noise-free model data obtained from the electric arc furnace model mentioned above are considered. To each of them is added 14.28% random white noise repeatedly. The experimental statistical results of error rate and missing rate are listed in Table 1. From the 10 experimental statistical results, the average error rate is 0.05% and the average missing rate is 0.03%, and it proves that the proposed detection method has a better repeatability.

The second data set is a data sequence obtained by an improved model of Alexandridis et al. (2003), given by:

\[
    y(k) = \frac{y(k-1)y(k-2)y(k-3)u(k-2)(y(k-3)-1) + u(k-1)}{1 + 0.2\sin(2\pi k/25) + y(k-2)^2 + y(k-3)^2}
\]  

(21)

with

**Figure 4.** Signals of an electrode and the detection results.

**Figure 5.** The detection results for signals of an electrode by autoregressive (AR) model detection method.
\[ u(k) = \begin{cases} \sin \left( \frac{2\pi k}{250} \right), & k \leq 500 \\ 0.8 \sin \left( \frac{2\pi k}{250} \right) + 0.2 \sin \left( \frac{2\pi k}{250} \right), & k > 500 \end{cases} \] (22)

Observing Equation (21), it is easy to find a time-varying term \( 0.2 \sin \left( \frac{2\pi k}{250} \right) \) being added to the denominator. So the second data set is used to demonstrate that our proposed method not only adapts to the oscillating data but can also detect outliers that come from a time-varying data sources.

Figure 6 shows the curve of the second data set with 10% white noise and eight outliers. Comparing Figure 6(a) with Figure 4(a), we can find the image in Figure 6(a) is more oscillating. Like the former example, Figures 6(b) and (c) also show the residual errors calculated by a robust RBFN (the number of hidden layer nodes is 5) and the wavelet coefficients of residual errors (scale \( s = 1/80 \), respectively, and the detection results, shown in Figure 6(d), are that all of the outliers can be detected.
Figure 7 shows the detection results using the AR model detection method proposed by Takeuchi and Yamanishi (2006). It can be seen from the figure that several normal points near the true outliers are detected as outliers. On the other hand, Takeuchi and Yamanishi (2006)'s method needs to set the detection threshold first, but it is obvious that choosing an appropriate and accurate threshold for the unknown data is not easy work. This defect will seriously affect the practical application of the method.

Furthermore, some statistic work for evaluating the execution time and the running speed of our proposed approach have been employed in order to verify its real-time performance; some results are shown in Table 2. The target data used here are two sets of the data mentioned above, and the hardware platform for detection is a 2.66 GHz dual core CPU with 1GB RAM, the configuration of which is not high. We tested 10 times to obtain the longest time for each set and computed the average time for each datum. In 10 tests, the

Figure 8. Detection for signals of a real process control system (PCS) by the proposed detection method.
statistical results of the first data set, the longest processing time is 4.59 s and the average processing time for each datum is 4.15 ms, for which is much smaller than the control cycle of the PCS (50 ms), for the test results of the second data set. It means that the operation process for the data just collected can finish completely within a control cycle. So we can conclude that our proposed detection approach is suitable for online running, since the average time for each datum is much smaller than the control cycle.

**Application**

We applied our proposed approach to the real data set sampled from an electric arc furnace electrode regulator system. Figure 8(a) shows the real data with big noise and some outliers marked either by ellipse-meaning outliers in the regulation process or by passphrase-denoting outliers in the stationary process. Similarly to the previous section, Figures 8(b) and (c) display the residual errors calculated by a robust RBFN (here the number of hidden layer nodes is 7) and the wavelet coefficients of residual errors (scale s = 1/80) respectively. The detection results are shown in Figure 8(d). Observing Figure 8(a), this is not clear enough to find the obvious outliers at the area between the 50th step and the 80th step (enclosed by a rectangle in Figure 8a), but in Figures 8(b) and (d), such a place shows the obvious characteristics of outliers. In order to make readers ‘see’ these outliers clearly, we show the detail from the 51st step to the 68th step by adding a local enlarged image in Figure 8(a). From this, we can find that the data from the 51st step to the 68th step change continuously, but from the 69th step to the 74th step, there are abnormal serious oscillations and actually these points are outliers. The simulation and analysis above further verify that our proposed detection approach is valid and more suitable for the data in PCS than the conventional wavelet analysis technique.

**Conclusions**

We have proposed a scheme for detecting outliers from the non-stationary time series sampled from the PCS. One of its features is that it used the outlier probability of each datum as the weight of the objective function of a robust RBFN, which makes the robust RBFN avoid the influence of outliers. We investigated a fitting residual by the wavelet analysis technique, and then made a decision using an HMM. The HMM has two structural chains for online deciding whether a datum is an outlier, which can avoid pre-selecting the threshold. In the wavelet analysis part, we introduce an iterative wavelet decomposition technique to decompose the residual signals online. Thereby, we were able to deal with the outlier detection problem online.

We have also illustrated the efficiency and accuracy of the proposed outlier detection algorithm by comparing it with the conventional wavelet-based method and AR model detection method. The results show that the proposed method is able to detect outliers in a non-stationary data sequence while the conventional wavelet method is not, and has a higher accuracy for detecting the process data of the PCS. Furthermore, the analysis on run time and the detection effect of the practical process data confirms that the method is ‘simple’ enough to meet the real-time performance of the practical PCS and has significant practical application to some extent.

**Funding**

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

**References**


