Just-in-time learning based soft sensor with variable selection and weighting optimized by evolutionary optimization for quality prediction of nonlinear processes

Article in Chemical Engineering Research and Design · February 2019

DOI: 10.1016/j.cherd.2019.02.004

7 authors, including:

Bei Pan
Beijing Institute of Technology

Huaiping Jin
Kunming University of Science and Technology

Accepted Manuscript

Title: Just-in-time learning based soft sensor with variable selection and weighting optimized by evolutionary optimization for quality prediction of nonlinear processes

Authors: Bei Pan, Huaiping Jin, Li Wang, Bin Qian, Xiangguang Chen, Si Huang, Jiangang Li

PII: S0263-8762(19)30050-4
DOI: https://doi.org/10.1016/j.cherd.2019.02.004
Reference: CHERD 3513

To appear in:

Received date: 3 August 2018
Revised date: 29 January 2019
Accepted date: 3 February 2019

Please cite this article as: Pan B, Jin H, Wang L, Qian B, Chen X, Huang S, Li J, Just-in-time learning based soft sensor with variable selection and weighting optimized by evolutionary optimization for quality prediction of nonlinear processes, *Chemical Engineering Research and Design* (2019), https://doi.org/10.1016/j.cherd.2019.02.004

Just-in-time learning based soft sensor with variable selection and weighting optimized by evolutionary optimization for quality prediction of nonlinear processes

Bei Pan[a], Huaiping Jin[a,*] jinhuaiping@gmail.com, jinhuaiping@126.com, Li Wang[b], Bin Qian[a], Xiangguang Chen[c], Si Huang[a], Jiangang Li[a]

[a]Department of Automation, Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China

[b]Department of Disaster Prevention Apparatus, Institute of Disaster Prevention, Langfang 065201, China

[c]Department of Chemical Engineering, School of Chemistry and Chemical Engineering, Beijing Institute of Technology, Beijing 100081, China

[*]Corresponding author. Tel.: +86 15877986943.

Highlights

- A novel JIT learning soft sensing framework, namely JIT-VSW, is proposed.
- A mixture weighted similarity (MWS) is defined by combing multiple WED similarity measures.
- Input variable selection and weighting are integrated to improve the JIT learning performance.
- A mixed integer optimization approach is used to simultaneously optimize the selection of input variables and MWS parameters.
- Three real-world applications are used to demonstrate the effectiveness and superiority of JIT-VSW.

Abstract

Just-in-time (JIT) learning based soft sensors have been widely used for predicting product quality variables of nonlinear processes. They dynamically build online local models by selecting the samples most relevant to the query data from a historical database whenever an estimate is requested. However, building high-performance JIT soft sensors remains challenging due to difficulties defining similarity measures and the selection of input variables for facilitating efficient relevant sample selection and model building. In this study, we propose a novel soft sensing framework, referred to as JIT learning with variable selection and weighting (JIT-VSW). In this framework, a mixture weighted similarity (MWS) measure is defined by combining multiple weighted Euclidean distance (WED) based similarity measures. The MWS measure enables variable weighting embedded in WED measures to account for the relevance between input and output variables and facilitates the handling of highly complex process characteristics through the mixture-type similarity measure. Meanwhile, a wrapper optimization approach using evolutional algorithms is proposed for input variable selection. Further, the selection of input variables and the determination of MWS parameters, i.e., weights assigned to input variables and mixture coefficients of WED similarity measures, are formulated as a mixed integer optimization problem and solved simultaneously by using the mixed integer genetic algorithms (MIGA). The effectiveness and superiority of JIT-VSW are verified through three real-world applications.

Key words: soft sensor; just-in-time learning; mixture weighted similarity measure; input variable selection; mixed integer genetic algorithms; locally weighted partial least squares.

1 Introduction

Online real-time measurements of key product quality variables are vital for facilitating advanced process control, monitoring, and optimization to improve product quality, energy efficiency, and costs in process industries. Although measurement using hardware instruments is desirable, online quality measurements are not always available due to technological or economic limitations, e.g., the large delays for laboratory analysis and high investment costs of hardware devices. Fortunately, soft sensor technology has developed as a promising solution to this problem (Fortuna et al., 2007; Haimi et al., 2013).

The core of a soft sensor is a mathematical model that describes the relationship between a set of secondary variables (i.e., easy-to-measure variables) and one primary variable (i.e., a difficult-to-measure variable). Generally, soft sensors can be categorized into two groups: first-principle models and data-driven models. The most reliable approach to achieve soft sensor development is through the use of first-principle models; however, such models are not available in many practical applications where in-depth physical or chemical information of the process are often severely lacking. Alternatively, data-driven models have gained popularity in soft sensor applications due to the minimal requirements of mechanical knowledge and the availability of large amounts of process data (Kano and Nakagawa, 2008; Kadlec et al., 2009; Ge et al., 2013; Ge et al., 2017). Thus, data-driven soft sensor modeling is considered in this research.

Traditionally, data-driven soft sensor modeling mainly focuses on global approaches such as partial least squares (PLS) (Shao et al., 2015), independent component regression (ICR) (Ge and Song, 2014), support vector machine (SVM) (Kaneko and Funatsu, 2014; Jin et al., 2015), artificial neural networks (ANN) (Gonzaga et al., 2009) and Gaussian process regression (GPR) (Xiong et al., 2017; Mei et al., 2017). These methods aim to build a single regression model with a universal generalization performance given the underlying assumption of a constant operation phase or mode. However, industrial processes are often characterized by strong nonlinearity, time-varying behavior, and multiplicity of phases or modes, thereby leading to inaccurate predictions from global soft sensors. Moreover, due to high model complexity, global soft sensors suffer from difficulty performing optimizations and model updates when new data become available. To tackle these issues, many attempts have been made to develop local learning based soft sensors (Kadlec and Gabrys, 2011; Jin et al., 2014; Liu et al., 2015; Wang et al., 2015).

Recently, JIT learning, one popular learning paradigm of local learning, has been widely employed for soft sensor modeling (Liu et al., 2012; Liu et al., 2012; Kano and Fujiwara, 2013; Kim et al., 2013; Jin et al., 2015; Yuan et al., 2016a; Yuan et al., 2016b). Following the divide-and-conquer philosophy, the JIT approach attempts to dynamically construct local predictive models only when an estimation to the query data is requested. In general, there are three main characteristics of JIT soft sensors. First, all input and output data are stored into a database. Second, a local model is built from samples most relevant to the query point only when estimation is required and the output estimation is obtained. Third, the constructed local model is discarded after finishing the estimation. One substantial advantage of JIT modeling is its strong ability of handling process nonlinearity, which is common in industrial processes. Another advantage of JIT is its inherent adaptive nature, which is achieved by simply adding new measurements into the database. In addition, JIT can effectively deal with multimode/multiphase process characteristics by selecting the most similar samples to the current process state. However, the development of high-performance JIT soft sensors remains challenging.

One major issue of building accurate JIT soft sensors lies in the definition of similarity functions, which has become a hot topic in recent years. Conventionally, similarity measures are based on Euclidean or Mahalanobis distances (Kano and Fujiwara, 2013). Because the consideration of data observations as points in space leads to distance and angle measures, Cheng and Chiu (2004) proposed combining the two measures in order to enhance the predictive capability of JIT. Another similarity measure is defined based on the correlation among variables (Fujiwara et al., 2009). Moreover, non-Gaussian similarity measures are proposed to cope with the non-Gaussian characteristics of process data (Fan et al., 2014; Xie et al., 2014). Despite the availability of various similarity measures, the Euclidean distance

(ED) based similarity is still dominant in JIT learning due to its simplicity and comprehensibility. By considering the importance of input variables to the output variable, the weighted Euclidean distance (WED) similarity can be defined, which can further enhance the prediction performance of JIT soft sensors (Kim et al., 2013). When the weights assigned to input variables are the same, the WED similarity degenerates as an ED measure.

However, for the definition of WED similarity, the determination of weights is key yet problematic. Traditionally, some attempts have been made to define the weights as the regression coefficients of the input variable by using a linear modeling technique (Shigemori et al., 2011; Kim et al., 2011). However, such weight assignment does not always function well, especially for strong nonlinearity between input and output variables. More importantly, the weights resulting from linear models may be ill-suited for other modeling techniques. Therefore, it is appealing to wrap the weight determination with local modeling techniques. In fact, the determination of weights is essentially an optimization problem. Thus, we attempt to solve this issue by using evolutional optimization approaches, which have gained significant success in both machine learning and engineering (Marler and Arora, 2004; Jin and Sendhoff, 2008; Rangaiah and Bonilla-Petriciolet, 2013).

Moreover, a drawback of many current similarity measures is that only one single similarity measure is employed to select similar samples. Such similarity measures may not effectively accommodate complex processes because they can only deal with partial process characteristics. Therefore, a certain similarity criterion combining multiple simple similarity measures would be useful to address as many process characteristics as possible, i.e., multiple simple similarity measures are integrated into one mixture form using a weighting strategy. Again, the definition of mixed weight should be seriously considered. Similarly, this issue will be solved by evolutionary optimization in this study.

Another issue of JIT soft sensor modeling concerns input variable selection. To build a high-performance JIT soft sensor model, the input variables should be highly relevant to the target variable. However, in many real-world applications, collinearity or redundancy often exists among the available process variables due to the redundant installation of hardware instruments. In soft sensor modeling, the blind utilization of all input variables may increase model complexity and, even worse, deteriorate the predictive capability of soft sensors when involving irrelevant input variables. Therefore, the appropriate selection of input variables for model construction is crucial for enhancing the estimation accuracy and computational efficiency (George, 2000; Jian et al., 2017).

In general, commonly used methods for input variable selection are categorized into two classes: filter and wrapper methods. Filter techniques select input variables using a statistical measure of the degree of dependence between the candidates and output variables; for example, the linear correlation coefficient and mutual information criterion. Although filter approaches are model-free and easy to implement, the performance of input variable selection filters is largely dependent on the statistical dependency measure. Moreover, the resulting input sets may be ill-suited for the selected local modeling technique. In contrast, wrapper methods deal with input variable selection as optimization of the model structure and select the input sets that yield the optimal performance of the predictive model. In detail, wrapper approaches achieve input variable selection by forward selection or backward elimination until the model performance is no longer enhanced. However, considering that the number of possible subsets of input variables grows exponentially with the input dimension, the computational requirements of model evaluation may be formidable in the high-dimensional case. One appealing solution to this problem is utilizing random search methods such as evolutional optimization algorithms, which can efficiently avoid exhaustive search by mimicking the principles of biological evolution.

In this study, a novel JIT modeling framework, referred to as JIT-VSW (JIT learning with variable selection and weighting), is proposed, using the locally weighted partial least squares (LWPLS) technique for local modeling. In this method, a novel mixture weighted similarity (MWS) criteria, incorporating feature weighting and a combination of multiple WED similarity measures, is defined to achieve efficient relevant sample selection. In addition, as the determination of MWS parameters and the selection of input variables are essentially real and binary optimization problems, respectively, the simultaneous optimization of the two sub problems are formulated as a mixed integer optimization problem and further solved using the MIGA algorithm. Overall, the main contributions of this research are outlined as follows:

(1) A novel MWS similarity measure is defined to select similar samples for JIT learning. MWS accounts for the different influences of input variables on the output variable by assigning different weights to inputs. Moreover, MWS adopts a mixture of WED similarity measures instead of the traditional single-type similarity measure. Due to these two improvements, MWS enables more efficient handling of complex process characteristics, and thus facilitates more efficient relevant sample selection.

(2) We propose simultaneously dealing with the parameter optimization of the MWS similarity measure and input variable selection using the mixed integer optimization approach. The simultaneous optimization of real and binary decision variables can effectively avoid interactions between optimization of MWS parameters and that of the input selection. Moreover, this optimization approach is generally applicable to both linear and nonlinear modeling techniques.

(3) Based on the newly defined MWS similarity measure and the MIO approach for parameter optimization, a novel JIT modeling framework is developed, which can be used for soft sensor modeling of nonlinear process systems, as well as handling nonlinear regression problems in other areas.

The rest of the paper proceeds as follows. Section 2 briefly explains the principles of LWPLS and GA methods. Section 3 presents the details of the proposed JIT-VSW soft sensor approach. Section 4 reports three real-world application cases to demonstrate the efficiency of JIT-VSW by comparing it with four other soft sensor methods. Finally, the conclusions are drawn in Section 5.

2. Preliminaries

In this section, LWPLS and genetic algorithms (GA) are briefly introduced.

2.1 Locally weighted partial least squares

The LWPLS algorithm is one of the most popular JIT learning methods. It has been conceived as a promising soft sensing technology because of its advantage in dealing with changes in process characteristics as well as nonlinearity (Kim et al., 2011). Consider a dataset $\{\mathbf{x}_\mathrm{n}, \mathbf{y}_\mathrm{n}\}_{n=1}^{N}$ where the *n*th sample of input and output variables is denoted by

$$\mathbf{x}_n = [x_{n1}, x_{n2}, \dots, x_{nM}]^\mathrm{T} \tag{1}$$

$$\mathbf{y}_n = [y_{n1}, y_{n2}, \dots, y_{nL}]^\mathrm{T} \tag{2}$$

where *N* is the number of samples; *M* and *L* are the numbers of input and output variables, respectively; the superscript T denotes the transpose of a vector or matrix; and $\mathbf{X} \in \Re^{N\times M}$ and $\mathbf{Y} \in \Re^{N\times L}$ represent the input and output matrices whose *n*th rows are $\mathbf{x}_n^\mathrm{T}$ and $\mathbf{y}_n^\mathrm{T}$, respectively.

In LWPLS modeling, $\mathbf{X}$ and $\mathbf{Y}$ are stored in a database and the predictive model is built online. When an output estimation is required for a query sample $\mathbf{x}_q$, the similarity ω_n between $\mathbf{x}_q$ and $\mathbf{x}_n$ is calculated, and a local PLS model is built by weighting samples with a similarity matrix. The detailed procedure for predicting $\mathbf{y}_q$ is described as follows.

1. Determine the number of latent variables *R* and set *r*=1.

2. Calculate a similarity matrix $\mathbf{\Omega}$:

$$\mathbf{\Omega} = \mathrm{diag}\{\omega_1, \omega_2, \dots, \omega_N\} \tag{3}$$

where $\mathrm{diag}\{\cdot\}$ denotes a diagonal operator.

3. Calculate $\mathbf{X}_r$, $\mathbf{Y}_r$, and $\mathbf{x}_{q,r}$ as

$$\mathbf{X}_r = \mathbf{X} - \mathbf{1}_N[\bar{x}_1, \bar{x}_2, \dots, \bar{x}_M] \tag{4}$$

$$\mathbf{Y}_r = \mathbf{Y} - \mathbf{1}_N[\bar{y}_1, \bar{y}_2, \dots, \bar{y}_L] \tag{5}$$

$$\mathbf{x}_{q,r} = \mathbf{x}_q - [\bar{x}_1, \bar{x}_2, \dots, \bar{x}_M]^\mathrm{T} \tag{6}$$

$$\bar{x}_m = \sum_{n=1}^{N} \omega_n x_{nm} \Bigg/ \sum_{n=1}^{N} \omega_n \tag{7}$$

$$\bar{y}_l = \sum_{n=1}^{N} \omega_n y_{nl} \Bigg/ \sum_{n=1}^{N} \omega_n \tag{8}$$

where $\mathbf{1}_N \in \Re^N$ is a vector of ones

4. Set $\bar{\mathbf{y}}_q = [\bar{y}_1, \bar{y}_2, \dots, \bar{y}_L]^\mathrm{T}$.

5. Derive the *r*th latent variable of $\mathbf{X}$

$$\mathbf{t}_{r,} = \mathbf{X}_r \mathbf{w}_r \tag{9}$$

where $\mathbf{w}_r$ is the eigenvector of $\mathbf{X}_r^\mathrm{T}\mathbf{\Omega}\mathbf{Y}_r\mathbf{Y}_r^\mathrm{T}\mathbf{\Omega}\mathbf{X}_r$, which corresponds to the maximum eigen value.

6. Derive the rth loading vector of $\mathbf{X}$ and the regression coefficient vector.

$$\mathbf{p}_r = \frac{\mathbf{X}_r^{\mathrm{T}}\boldsymbol{\Omega}\mathbf{t}_r}{\mathbf{t}_r^{\mathrm{T}}\boldsymbol{\Omega}\mathbf{t}_r} \tag{10}$$

$$\mathbf{q}_r = \frac{\mathbf{Y}_r^{\mathrm{T}}\boldsymbol{\Omega}\mathbf{t}_r}{\mathbf{t}_r^{\mathrm{T}}\boldsymbol{\Omega}\mathbf{t}_r} \tag{11}$$

7. Derive the rth latent variable of $\mathbf{x}_q$.

$$t_{q,r} = \mathbf{x}_{q,r}^{\mathrm{T}}\mathbf{w}_r \tag{12}$$

8. Update $\hat{\mathbf{y}}_q = \hat{\mathbf{y}}_q + t_{q,r}\mathbf{q}_r$.

9. If $r = R$, end; otherwise, update

$$\mathbf{X}_{r+1} = \mathbf{X}_r - \mathbf{t}_r\mathbf{p}_r^T \tag{13}$$

$$\mathbf{Y}_{r+1} = \mathbf{Y}_r - \mathbf{t}_r\mathbf{q}_r^T \tag{14}$$

$$\mathbf{x}_{q,r+1} = \mathbf{x}_{q,r} - \mathbf{t}_{q,r}\mathbf{p}_r \tag{15}$$

10. Set $r \leftarrow r + 1$, and go to step 5.

The estimation performance of LWPLS strongly depends on the definition of the similarity between the query data $\mathbf{x}_q$ and the historical samples $\mathbf{x}_n$. Popular similarity measures are defined based on Euclidean distance, Mahalanobis distance, and angle.

2.2 Genetic algorithms

GA (Whitley, 1994) is one of the most influential evolutionary optimization algorithms inspired by the basic biological evolution theory “survival of the fitness in natural selection”. The basic idea of GA is to express decision variables as chromosomes via binary or real-valued coding and then search the optimal solution through three main operations: crossover, mutation and selection. During the crossover process, the offspring population is generated by crossing pairs of chromosomes in the current population. Mutation randomly changes some parts of the chromosomes to ensure the diversity of the new population. Moreover, the selection procedure aims to select the elite children with the best fitness values and send them directly to the next generation. By iterating the above evolution process, the optimal solution can be obtained.

3. Proposed JIT-VSW soft sensing framework

To construct a high-performance JIT learning soft sensor model, we propose a novel JIT-VSW soft sensor method, where the input variable selection and weighting are simultaneously considered. The resulting JIT-VSW modeling framework is illustrated in Fig. 1. Although there are many aspects influential to the JIT learning performance, JIT-VSW focuses on enhancing the estimation accuracy of JIT models by improving the similarity measure definition and input variable selection. In detail, a mixture weighted similarity measure (MWS) is defined by integrating feature weighting and a mixture of multiple simple similarity measures, which leads to the real-valued optimization of MWS parameters. Moreover, input variable selection can be viewed as a binary optimization problem. Thus, by combining the two sub optimization problems, a mixed integer optimization problem is formulated and further solved using an evolutional optimization approach. Details of the JIT-VSW soft sensing framework will be described in the following sections.

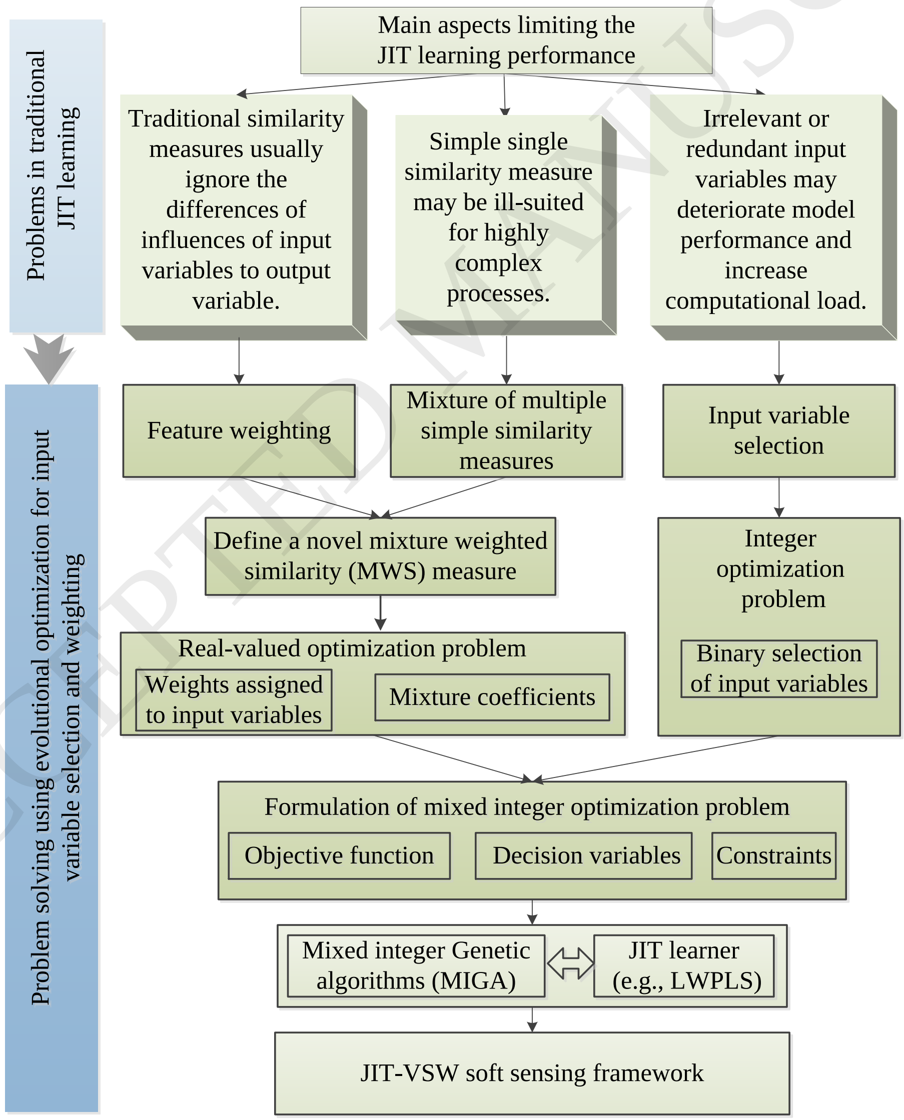


Fig. 1. Illustration of the JIT-VSW soft sensor modeling framework.

3.1 Definition of the mixture weighted similarity measure

JIT learning has no separate training phase, but when estimation of an output variable is requested, a sample subset most relevant to the query data is obtained to build a local model and produce the output variable estimation. The Euclidean distance (ED) based similarity measure is generally used to evaluate similarity between the query point and historical data. However, the ED criterion ignores the different effects of input variables on the output variable. An efficient solution to this issue is to consider feature weighting during the distance calculation, thereby generating the weighted ED (WED) similarity measure as follows:

$$\omega_n = \exp\left(-\frac{d_n}{\sigma_d \varphi}\right) \tag{16}$$

$$d_n = \sqrt{\left(\mathbf{x}_n - \mathbf{x}_q\right)^{\mathrm{T}} \mathbf{M} \left(\mathbf{x}_n - \mathbf{x}_q\right)} \tag{17}$$

$$\mathbf{M} = \mathrm{diag}(\theta_1, \theta_2, \dots, \theta_M) \tag{18}$$

where σ_d is the standard deviation of $d_n (n = 1,2, \cdots, N)$; φ is a localization parameter; and θ_m is the weight for the *m*th input variable.

Although the WED similarity measures can generate more accurate predictions than ED similarity measures, their performance is still restricted by their inherent simple structure. This is mainly because industrial processes often encounter strong nonlinearity and multi-mode and/or multi-phase features, whereas traditional single-type similarity measures cannot accommodate all operation states of complex processes. To tackle this issue, we propose defining a mixture weighted similarity (MWS) measure by integrating multiple single-type WED similarity measures through weighted averaging.

The concept of an MWS similarity measure is inspired by the principle of Gaussian mixture models (GMM), which enable non-Gaussian data description by mixing multiple Gaussian components. Thus, we guess that if a mixture of simple similarity measures is used for relevant sample selection, JIT learning may achieve performance enhancement, which is verified through the application examples in Section 4.

The MWS similarity measure is defined as

$$\omega_{mix,n} = \xi_1 \omega_{1,n} + \xi_2 \omega_{2,n} + \cdots + \xi_K \omega_{K,n} \tag{19}$$

where $\xi = \{\xi_1, \xi_2, \dots, \xi_K\}$ denotes a set of mixture coefficients satisfying $\sum_{k=1}^{K} \xi_k = 1$; $\omega_{k,n}$ denotes the similarity calculated from the kth WED similarity measure that is equipped with $\Theta_k = \{\theta_1^k, \theta_2^k, \dots, \theta_M^k\}$, and K is the number of component WED similarity measures. In the case of $K = 1$, MWS degenerates into the single-type WED criterion.

After determining the MWS structure, one major difficulty in implementing the MWS similarity measure for relevant sample selection is the definition of mixing coefficients ξ and weights assigned to input variables $\Theta = \{\Theta_1, \Theta_2, \dots, \Theta_K\}$. As shown in Eqs. (17) and (19), the mixture similarity calculation strongly depends on the definition of Θ and ξ. Although some efforts have been made to determine the weights of input variables as the absolute values of linear regression coefficients, such feature weighting schemes are unsuitable for nonlinear problems. Moreover, as the local modeling technique for JIT learning is ignored, it is difficult to ensure high estimation accuracy based on regression coefficients weighting.

Therefore, in this work, we attempt to determine the mixing coefficients and input variable weights using a wrapper approach, where the selection of MWS parameters involves the prediction performance evaluation of JIT learners and retaining those parameters that lead to the best estimation accuracy. The process of selecting MWS parameters is essentially an optimization problem. Because classical optimization approaches have rigorous restrictions on the nature of the problem formulation such as continuity, convexity, and differentiability, a popular evolutional optimization approach, GA, is adopted here to optimize the MWS parameters. Note that other evolutional algorithms, such as particle swarm optimization and differential evolution, are also applicable.

To implement GA optimization, the MWS parameters should be represented as chromosomes. Assuming that the MWS similarity measure consists of K component simple WED similarity measures, the following decision variables should be optimized:

$$\Gamma = \left\{\{\theta_1^1, \theta_2^1, \dots, \theta_M^1\}, \{\theta_1^2, \theta_2^2, \dots, \theta_M^2\}, \dots, \{\theta_1^K, \theta_2^K, \dots, \theta_M^K\}, \{\xi_1, \xi_2, \dots, \xi_K\}\right\} \tag{20}$$

Because the MWS parameters are real numbers, real-valued coding is employed to generate the chromosomes for the optimization of weight θ assigned to input variables and the mixing coefficient ξ, where two input variables are assumed. In order to evaluate the validity of the individuals in the evolving population, an independent validation data set is used. Let $E_{val}(\Gamma)$ denote the estimated validation error corresponding to the individual Γ. It is clear that a smaller $E_{val}(\Gamma)$ results in a better Γ, so $fitness(\Gamma) = 1/E_{val}(\Gamma)$ is used as the fitness function. In addition, the upper and lower bounds of decision variables should be satisfied during the evolution process.

An advantage of the proposed MWS similarity measure lies in its strong capability of handling complex process characteristics due to the mixture of multiple simple WED similarity measures. By employing such combination, the nonlinearity and multiplicity of operation modes and/or phases can be characterized more effectively than by using the single-type WED similarity measure.

Another advantage of MWS lies in the efficiency and good applicability of the GA based optimization approach. On the one hand, because the local model technique is involved, it becomes easy to obtain the MWS parameters producing high model performance. On the other hand, the GA based optimization method applies to both linear and nonlinear model construction.

3.2 Input variable selection using GA

Input variable selection is essentially a binary optimization problem, where "0" and "1" denote the removal and involvement of input variables, respectively. Thus, in this study, we employ GA optimization for this purpose. The selection of input variables is encoded using a binary string, where each bit denotes whether the corresponding input will be selected. Thus, the length of the chromosome is equal to the number of input variables. Let $S = \{s_1, s_2, \dots, s_N\}$ be the binary states for input variable selection and $E_{val}(S)$ be the validation error corresponding to S. Then the fitness function is defined as $fitness(S) = 1/E_{val}(S)$. The GA algorithm evolves the chromosomes to find the optimal subset of input variables to maximize the fitness function.

3.3 Simultaneous optimization of input variable selection and weighting using MIGA

As introduced in the above sections, both MWS parameter determination and input variables selection can be solved within the framework of single-objective optimization. It is a common practice to address the two sub optimization problems through a sequential procedure, i.e., performing input viable selection then optimizing MWS parameters. However, some interactions exist between the two problems. That is to say, the optimal input variable set obtained from JIT learning using an ED similarity measure may be ill-suited for the optimization of MWS. When applying different WED similarity measures, the optimal input variable set may also change. Thus, it is more natural to optimize input variable selection and MWS parameters simultaneously.

Because the MWS parameters are coded in real-valued string while the selection of input variables is coded in binary string, we integrate the two sub optimization problems into a mixed integer optimization (MIO) problem formulated as follows:

$$\min\ f(S,\Theta,\xi)$$

which is subject to

$$\begin{cases} g_k(S,\Theta,\xi) \le b_k, k = 1,2,\dots,K \\ s_i^L \le s_i \le s_i^U, integer, i = 1,\dots,n_1 \\ \theta_j^L \le \theta_j \le \theta_j^U,\quad j = n_1+1,\dots,n_2 \end{cases} \tag{21}$$

where $f(S,\Theta,\xi)$ is the objection function, $g_k(S,\Theta,\xi)$ stands for the inequality constraint with respect to the kth component WED similarity measure; the superscripts L and U denote the lower and upper bounds of decision variables, respectively; and n_1 and n_2 are the number of input variables and MWS parameters, respectively.

As shown in Eq. (21), the decision variables $\{S,\Theta,\xi\}$ of the MIO problem include the binary values for input variable selection and the real-valued MWS parameters. Given an candidate set of $\{S,\Theta,\xi\}$, the objective function $f(S,\Theta,\xi)$ is estimated from an independent validation set:

$$f(S,\Theta,\xi) = \sqrt{\frac{1}{N_{\mathrm{val}}}\sum_{i=1}^{N_{\mathrm{val}}}\left(\hat{y}_{val,i} - y_{val,i}\right)^2} \tag{22}$$

where $\hat{y}_{val,i}$ and $y_{val,i}$ denote the predicted and actual values of the output variable, respectively; and N_{val} denotes the number of validation samples.

Further, inequality constraints associated with S and Θ are considered. It is clear that input variable selection has a key influence on the weight assignment of input variables. When one input variable is removed from the initial input set, the corresponding weight becomes meaningless and can be set to zero. Thus the following inequality constraint should be satisfied:

$$Zs_m \geq \theta_m^1 + \theta_m^2 + \cdots + \theta_m^K \tag{23}$$

where Z stands for an infinite positive number, s_m is a binary variable; and θ_m^k denotes the corresponding weight for the input variable x_m. If $s_m = 1$, the sum of $\theta_m^1,\ldots,\theta_m^K$ cannot exceed the infinite Z because s_m and θ_m^k are bounded. If $s_m = 0$, Eq. (23) holds only when $\theta_m^1,\ldots,\theta_m^K$ are zero.

Next, the MIGA algorithm is applied to solve the MIO problem in Eq. (21). The objective function in Eq. (22) is used to define the fitness function $fitness(S,\Theta,\xi) = 1/f(S,\Theta,\xi)$, and the upper and lower bounds constraints, together with the inequality constraint in Eq. (23), are considered during MIGA optimization. Details of the MIGA algorithm can be found in (Deep et al., 2009).

Figure 2 illustrates the flow diagram of MIGA optimization for the proposed JIT-VSW soft sensing framework. First, an initial population is generated by representing the decision variables $\{S,\Theta,\xi\}$ as chromosomes via the mixed coding scheme. Then, the population evolves through selection, crossover and mutation operations until the stopping condition is reached. Finally, the individual resulting in the highest fitness value is decoded to provide the best parameter settings for the proposed JIT-VSW soft sensor.

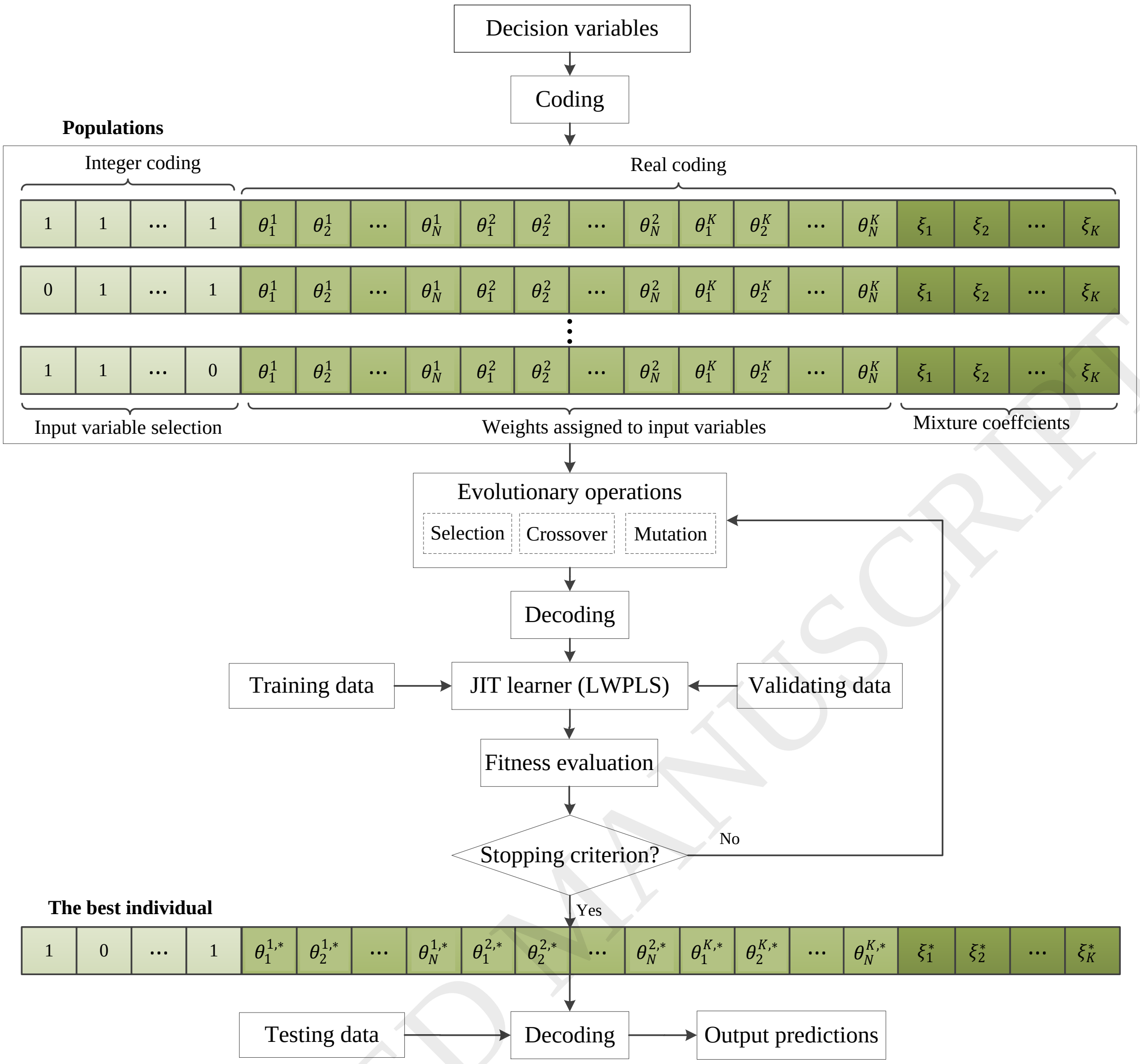


Fig. 2. Flow diagram of MIGA optimization for the JIT-VSW soft sensing framework.

In addition to the advantages resulting from the separate implementation of GA based MWS parameter optimization and input variable selection, the MIGA based optimization approach can achieve the simultaneous optimization of binary and real-valued decision variables. Moreover, interaction between the input variable selection and weighting can be effectively avoided within the MIGA optimization framework.

Though the proposed JIT-VSW method can benefit from the strong adaptability and flexibility of MIGA optimization, there still exists a high risk of overfitting. Generally speaking, evolutionary optimization approaches like GA may cause overfitting to the training samples if the decision variables are over-optimized. To deal with this issue, it is suggested to define the fitness function on an independent validation set or by cross-validation method instead of using training data directly. Moreover, the number of mixed WED similarity measures should not be too large to avoid overfitting.

3.4 Implementation procedure of JIT-VSW soft sensor

The JIT-VSW soft sensor is implemented in two stages: offline optimization and online prediction.

Offline optimization stage

(i) Collect the process input and output data for model construction and divide the modeling data into three subsets: a training set, validation set, and testing set.

(ii) Integrate the determination of MWS parameters and selection of input variables into a mixed integer optimization problem.

(iii) Apply the MIGA method to solve the mixed integer optimization problem formulated in Step (ii).

(iv) Reconstruct the database with the selected input variables.

Online prediction stage

(i) When a query data come, the mixed similarities between the query point and historical samples are calculated by using the weighted average of multiple component WED similarity measures.

(ii) Employ a relevant data set with high similarities to the query data to construct a local model, e.g., LWPLS.

(iii) Predict the output variable based on the constructed JIT model and discard it after estimation.

(iv) When new query data become available, return to Step (i).

4 Case studies

In this section, the proposed JIT-VSW soft sensing method is verified through three application examples: (a) predicting the penicillin concentration of the simulated fed-batch penicillin cultivation process; (b) predicting the butane content of the industrial debutanizer column process; and (c) predicting the Mooney viscosity of the industrial rubber mixing process. The methods used for comparison are as follows:

(i) PLS (partial least squares regression) (Wold et al., 2001): a global model.

(ii) LWPLS (locally weighted PLS) (Kim et al., 2013): a PLS model constructed by weighting the input data with a similarity matrix, where the input variable selection is not considered.

(iii) IVS-LWPLS (input variable selection based LWPLS): an LWPLS model using the input variables selected by GA optimization.

(iv) MWS-LWPLS (mixture weighted similarity measure based LWPLS): an LWPLS model using a combined weighted similarity measure MWS optimized by GA, where the input variable selection is not considered.

(v) JIT-VSW (JIT learning with variable selection and weighting): the proposed method. In this method, input variable selection and parameters of the MWS similarity measure are optimized simultaneously using MIGA. Without loss of generality, LWPLS is selected as the local modeling technique for JIT learning.

IVS-LWPLS, MWS-LWPLS and LWPLS can be viewed as degenerated variants derived from JIT-VSW. First, if only the input variable selection is considered, then the IVS-LWPLS algorithm is obtained. That is, IVS-LWPLS ignores the input variable weighting and only employs the single-type and unweighted similarity measure. Second, if only the mixture weighted similarity measure is employed, then the MWS-LWPLS algorithm is obtained. That is, MWS-LWPLS does not consider input variable selection. Third, if the input variable selection is not considered and the traditional single-type ED similarity measure is used, then the LWPLS algorithm is obtained. That is, input variable selection and weighting, as well as the mixture of similarity measures, are not considered in LWPLS modeling.

The root-mean-square error (RMSE) and coefficient of determination (R^2) are used to evaluate the prediction performance of soft sensor models:

$$\mathrm{RMSE} = \sqrt{\frac{1}{N_{\text{test}}}\sum_{i=1}^{N_{\text{test}}}(\hat{y}_i - y_i)^2} \tag{24}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N_{\text{test}}}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{N_{\text{test}}}(y_i - \bar{y})^2} \tag{25}$$

where N_{test} denotes the number of testing samples; $\hat{y}_i$ and y_i are the ith predicted and actual values, respectively; and $\bar{y}$ represents the mean value.

To construct high-performance soft sensor models, the optimal model parameters are determined by minimizing the prediction errors using independent validation data set. For the three case studies, the search ranges for different parameters are as follows:

(i) The number of latent variables, $R \in [1,2,\cdots,M]$ for PLS, LWPLS, IVS-LWPLS, MWS-LWPLS, and JIT-VSW, where M denotes the input dimension.

(ii) The localization parameter, $\varphi \in [0.1,0.5,1,1.5]$ for LWPLS, IVS-LWPLS, MWS-LWPLS, and JIT-VSW.

(iii) The number of component WED similarity measures in MWS similarity measure, $N_{\text{mix}} \in [1,2,\ldots,5]$ for MWS-LWPLS and JIT-VSW.

(iv) The population size and maximal generations in GA and MIGA optimization are set to 100 for IVS-LWPLS, MWS-LWPLS and JIT-VSW.

(v) The lower and upper bounds for optimization of MWS parameters in MWS-LWPLS and JIT-VSW are set to $lb = 0$ and $ub = 3$, where lb denotes the lower bound and ub denotes the upper bound. Moreover, the decision variables in terms of input variable selection $s_i(i = 1,2,\cdots,M)$ in IVS-LWPLS and JIT-VSW are chosen from $\{0,1\}$.

(vi) The local modeling size L for LWPLS, IVS-LWPLS, MWS-LWPLS, and JIT-VSW should be neither too small nor too large to guarantee the computational efficiency and numerical stability of JIT learning. Thus, $L = 100$ is employed for the penicillin fermentation case study, and $L = 50$ is employed for the debutanizer column and rubber mixing case studies.

4.1 Simulated fed-batch penicillin fermentation process

4.1.1 Process description

The fed-batch penicillin fermentation process has been widely used for exploring modeling, monitoring, and control of batch processes. A flow diagram of the fed-batch penicillin fermentation process is shown in Fig. 3. There are two typical operational phases during cultivation: a pre-culture phase lasting approximately 45 h and a fed-batch phase lasting approximately 355 h. During the initial pre-culture phase, a large amount of necessary cell mass is obtained. Then, nutrients such as glucose are fed continuously to promote the formation of penicillin during the fed-batch phase. However, the penicillin concentration have to be obtained through manual chemical analysis carried out off-line in laboratories with a large measurement delay of 4~8 h, which has been the main bottleneck for facilitating efficient monitoring and controlling of bioprocesses. Therefore, to achieve a high product formation rate, it is highly desirable to monitor the quality in real time during the cultivation process.

In this study, process data are generated by the PenSim simulator platform, which was developed by the Process Modeling, Monitoring, and Control Research Group of Illinois Institute of Technology (Birol et al., 2002; Zhao et al., 2009). A total of nine batches are collected under the default settings, including five batches for training, two for validation, and two for testing. Penicillin concentration, which is often obtained through offline analysis, is chosen as the target variable and 14 highly related process variables (Table 1) are used as input variables for soft sensor modeling.

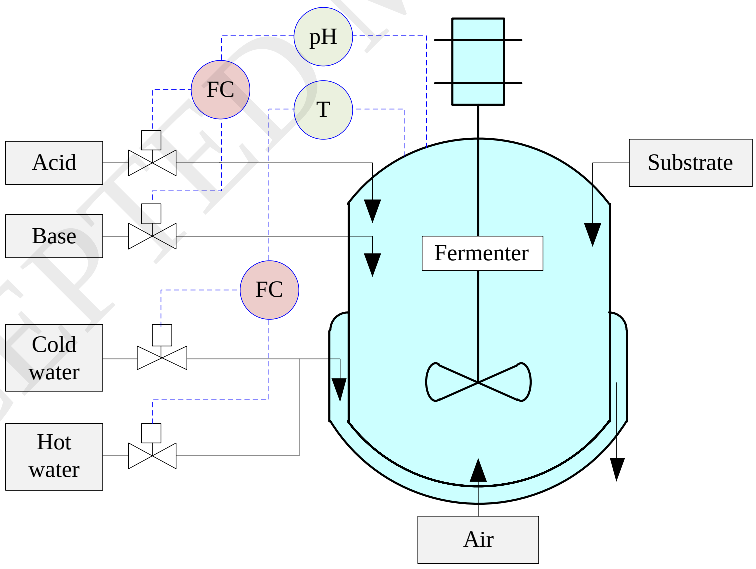


Fig. 3. Flow diagram of fed-batch penicillin fermentation process.

Table 1. Input variables for soft sensor development in the fed-batch penicillin fermentation process.

No.	Description(unit)

1	Culture time (h)
2	Aeration rate (L/h)
3	Agitator power (W)
4	Substrate feed rate (L/h)
5	Substrate feed temperature (K)
6	Dissolved oxygen concentration (g/L)
7	Culture volume (L)
8	Carbon dioxide concentration (g/L)
9	pH
10	Fermenter temperature (K)
11	Generated heat (kcal)
12	Acid flow rate (L/h)
13	Base flow rate (L/h)
14	Cooling water flow rate (L/h)

4.1.2 Online prediction results of penicillin concentration

The selected model parameters and resulting prediction results from different soft sensor methods are summarized in Table 2. It is readily observed that the PLS model provides the worst prediction result in terms of RMSE and R^2. This is because the prediction task is essentially a nonlinear regression problem, whereas PLS can only deal with linear modeling problems. Compared to PLS, LWPLS achieves significantly better prediction accuracy owing to the implementation of JIT learning. As opposed to the traditional LWPLS method, IVS-LWPLS and MWS-LWPLS achieve much better prediction performance as they consider the input variable selection and utilize the MWS similarity measure, respectively. In comparison, the proposed JIT-VSW approach, i.e., the combination of IVS-LWPLS and MWS-LWPLS, exhibits the best prediction performance, outperforming LWPLS by a large margin in terms of prediction RMSE value. This observation confirms that considering input variable selection and weighting is an effective way to improve JIT learning performance. It is also noteworthy that the number of component WED similarity measures in the MWS criterion required for the best-performing MWS-LWPLS and JIT-VSW is 2 and 4, respectively, implying the effectiveness of employing a mixture of multiple similarity measures for JIT learning. Also, the detailed prediction results in Fig. 4 show that JIT-VSW can obtain satisfactory predictions of penicillin concentration. Overall, the proposed JIT-VSW method is more desirable than other conventional JIT soft sensor methods as it provides accurate quality estimations for nonlinear processes.

Table 2 Prediction results of various soft sensor methods for the fed-batch penicillin fermentation process.

Method	L	R	φ	N_{mix}	RMSE	R^2
PLS	-	10	-	-	0.0276	0.9964
LWPLS	100	10	0.5	-	0.0228	0.9969
IVS-LWPLS	100	1	0.1	-	0.0159	0.9988
MWS-LWPLS	100	10	0.1	2	0.0192	0.9981
JIT-VSW	100	1	0.1	4	0.0138	0.9990

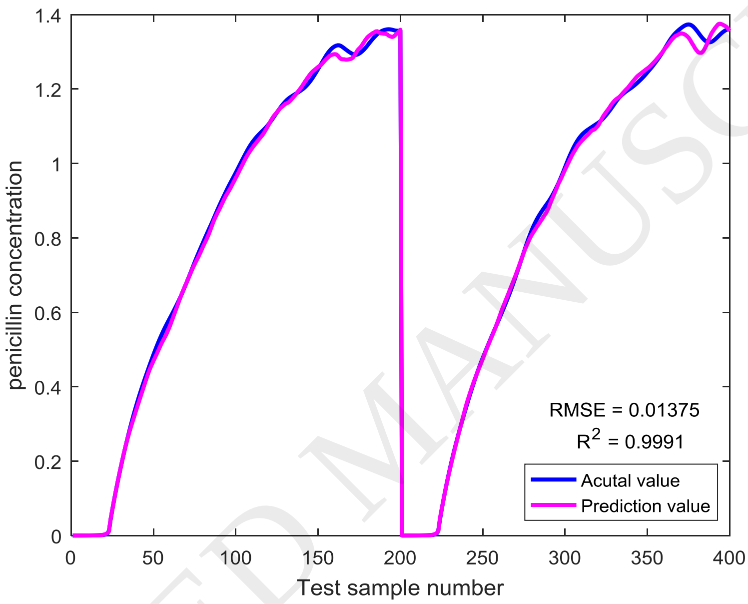


Fig. 4. Comparison of predicted and actual values obtained using the JIT-VSW soft sensor in the fed-batch penicillin fermentation process.

In addition, Fig. 5 presents the optimization results of input variable selection and MWS parameters. It is clear that only three input variables are selected as the informative features ("1" denotes the involvement of input variables and "0" denotes the rejection of input variables): substrate feed rate, culture volume, and generated heat. Moreover, the weights assigned to input variables for various WED similarity measures differ, and the differences between mixture coefficients indicate the different contributions of component WED similarity measures.

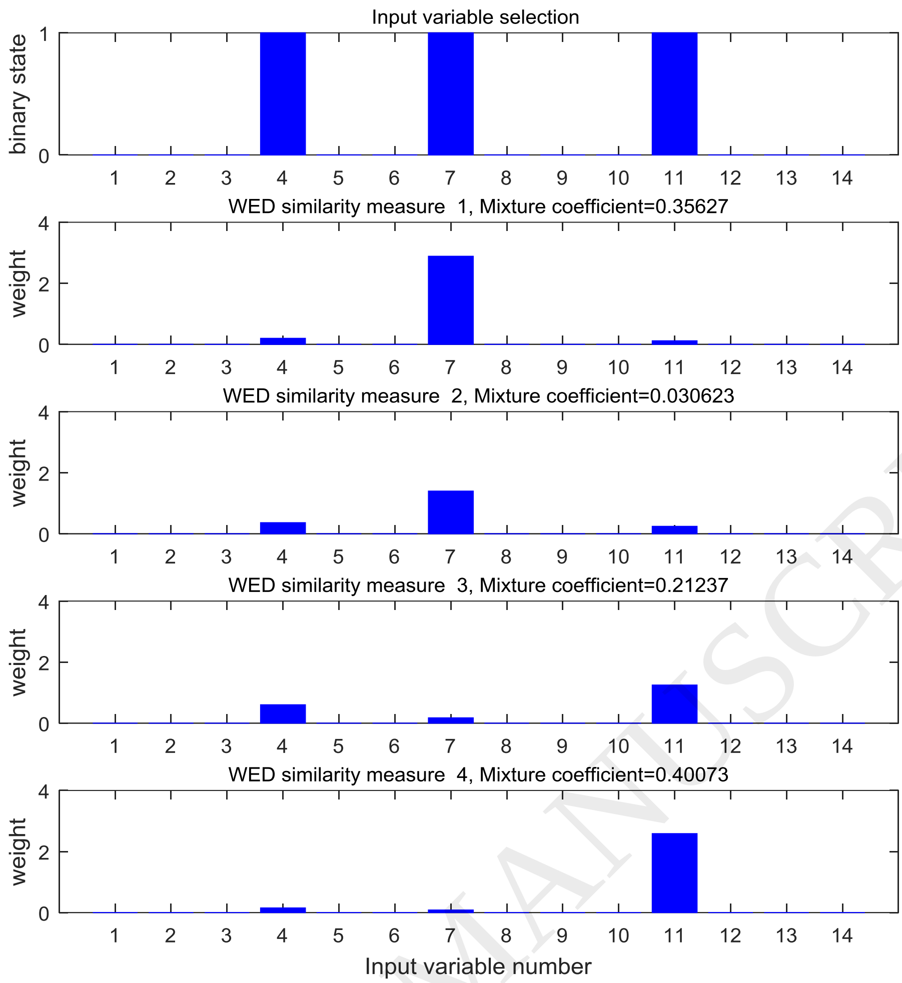


Fig. 5. MIGA optimization results in the fed-batch penicillin fermentation process.

4.2 Industrial debutanizer column process

4.2.1 Process description

The debutanizer column is used for desulfurization and naphtha cracking in the industrial refining process, which aims to remove propane and butane from the naphtha stream (Fortuna et al., 2007). In order to obtain high-quality naphtha products, the butane content in the bottom of the debutanizer column should be reduced as much as possible. However, the real-time measurements of butane concentration are still unavailable using hardware instruments. For example, the measurements from gas chromatograph can often cause a lag of 45 minutes. Alternatively, soft sensor methods are promising to predict the butane concentration online to improve product quality and process control efficiency. The flowchart of the debutanizer column process is illustrated in Fig. 6, where the grey circles represent hardware sensors installed in the plant to measure process variables. The input variables i.e., top temperature, top pressure, reflux flow, flow to next process, 6th tray temperature, bottom temperature A, bottom temperature B are used for soft sensor development. The process dataset can be downloaded from http://www.springer.com/1-84628-479-1.

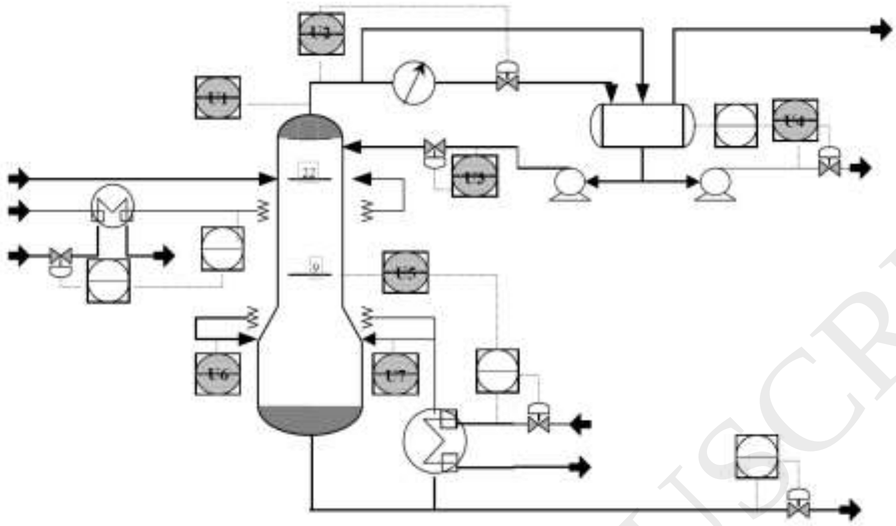


Fig. 6. Flowchart of the debutanizer column process.

4.2.2 Online prediction results of butane concentration

In this case study, both static and dynamic soft sensor models are designed to estimate the butane concentration. For the static scenario, 2394 samples are collected from the industrial process and the number of input variables is 7. Further, all samples are divided into three parts: 1197 samples for model training, 599 samples for parameter tuning and 598 samples for model testing. To cope with process dynamics, we employ the moving average (MA) model structure to build dynamic soft sensors, whose inputs can be formulated as $\mathbf{x}' = [\mathbf{x}(t) \ldots \mathbf{x}(t-d)]^{\mathrm{T}}$, where d denotes the maximal delayed time. In this study, $d = 6$. Accordingly, 2388 samples with 49 input variables are obtained for dynamic soft sensor model construction. In the same way, modeling samples are divided into three parts: 1194 samples for model training, 597 samples for parameter tuning and 597 samples for model testing. Moreover, all original input variables are used for building PLS, LWPLS, and MWS-LWPLS soft sensor models, while input variables are selected using GA and MIGA for IVS-LWPLS and JIT-VSW, respectively.

Tables 3 shows the selected model parameters and the corresponding prediction results from the static and dynamic soft sensor models. It is clear that PLS leads to considerably worse prediction performance with a significantly larger RMSE value than other methods; this is because PLS cannot effectively deal with the nonlinear process characteristics. In comparison, JIT learning methods, i.e., LWPLS, IVS-LWPLS, MWS-LWPLS, and JIT-VSW, achieve much better prediction results, exhibiting small RMSE values. Compared to LWPLS, IVS-LWPLS displays better prediction performance due to its consideration of input variable selection. It is worth noting that MWS-LWPLS performs better than LWPLS in the static test scenario, while providing less accurate predictions than LWPLS in the dynamic test scenario. This is mainly because of redundancy among the input variables for MWS-LWPLS. Thus, it is appealing to combine the advantages of IVS-LWPLS and MWS-LWPLS to enhance the JIT learning performance, which is confirmed by the outstanding estimation accuracy of the proposed JIT-VSW soft sensor method. These observations indicate that JIT-VSW performs best among the compared soft sensor methods. In addition, Table 3 reveals that dynamic soft sensor models clearly outperform static models because the dynamic predictive models can effectively handle the process dynamics.

Table 3 Prediction results of different static and dynamic soft sensors in the debutanizer column process.

Model type	Method	L	R	φ	N_{mix}	RMSE	R^2
Static	PLS	50	7	0.1	-	0.1418	0.2027
	LWPLS	50	1	0.1	-	0.0546	0.8817
	IVS-LWPLS	50	1	0.1	-	0.0454	0.9184
	MWS-LWPLS	50	2	0.1	1	0.0495	0.9029
	JIT-VSW	50	2	0.1	3	0.0452	0.9187
Dynamic	PLS	50	15	0.1	-	0.1165	0.4628
	LWPLS	50	2	0.1	-	0.0136	0.9927
	IVS-LWPLS	50	1	0.1	-	0.0132	0.9931
	MWS-LWPLS	50	1	0.1	1	0.0156	0.9887
	JIT-VSW	50	2	0.1	4	0.0121	0.9942

Furthermore, the prediction trends of butane concentration using the static and dynamic JIT-VSW models are illustrated in Fig. 7. The predictions from both static and dynamic JIT-VSW soft sensors correspond well to measured values of butane concentration. In comparison, the dynamic JIT-VSW model exhibits much smaller deviations between predicted and actual values than that of the static JIT-VSW model.

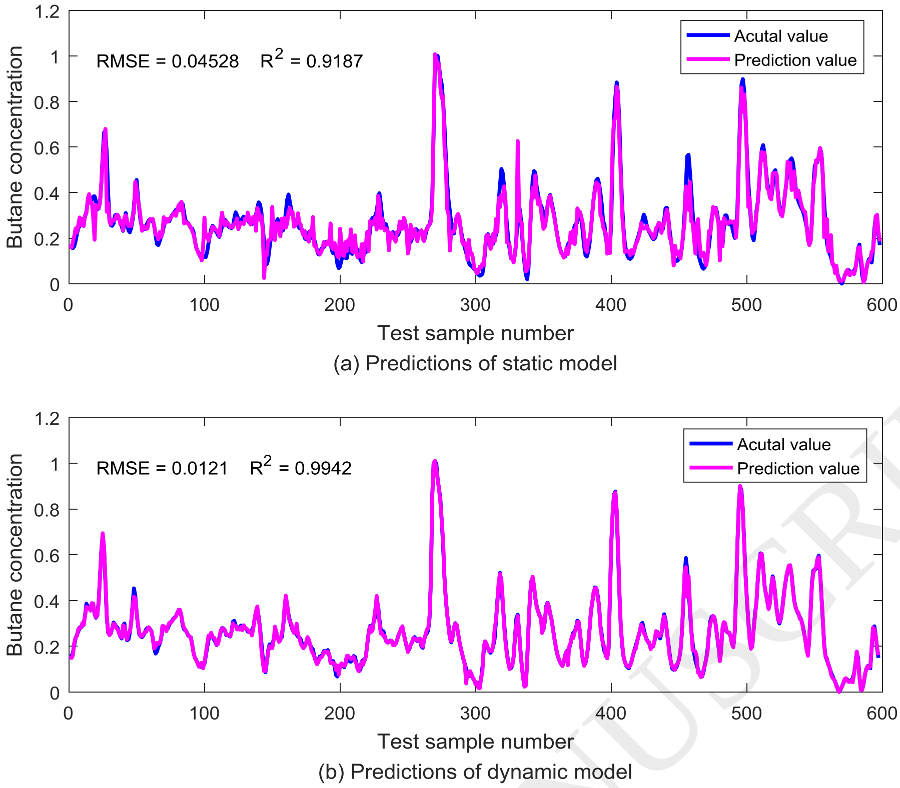


Fig. 7. Comparison of predicted and real values using the JIT-VSW soft sensor method in the debutanizer column process.

Finally, taking the dynamic soft sensor as an example, the detailed MIGA optimization results of input variable selection and MWS parameters are illustrated in Fig. 8. We observe that 34 informative input variables are selected and the weights of the corresponding input variables in component WED similarity measures are different. Meanwhile, the mixture coefficients reflect the contribution of the component similarity measures to the mixed similarity.

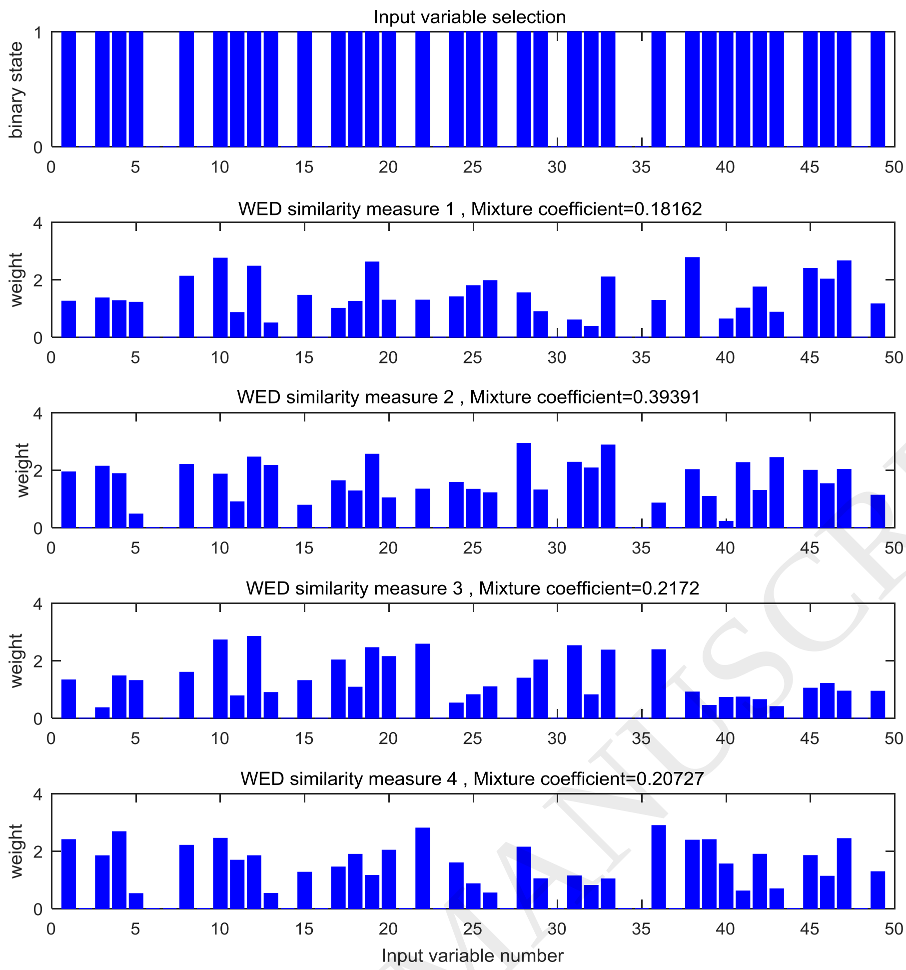
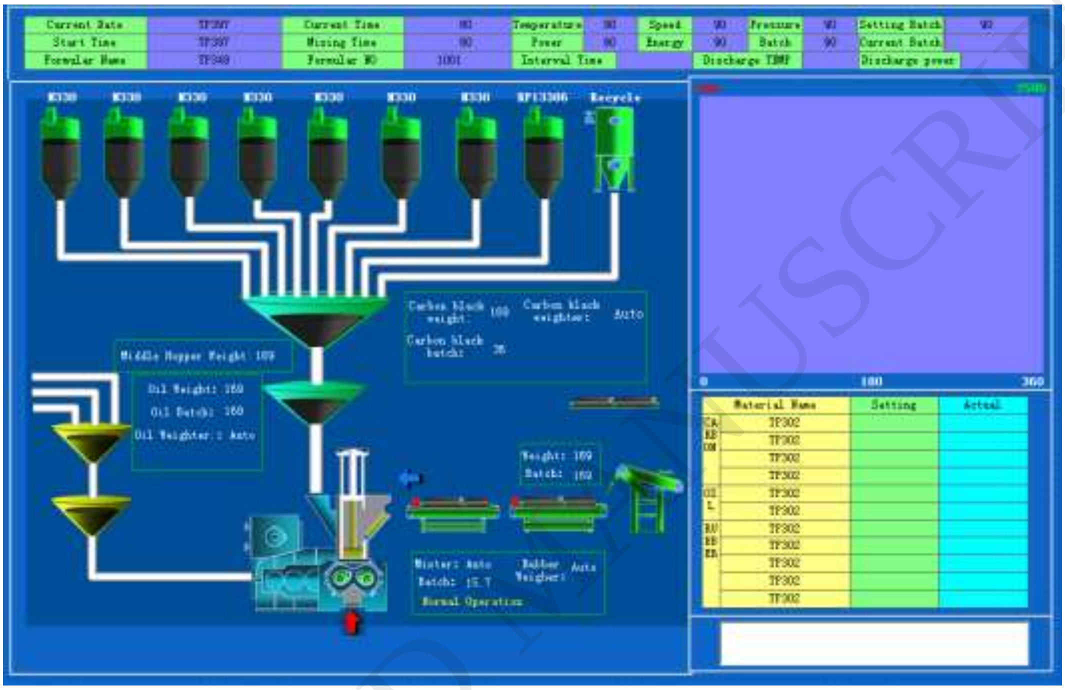


Fig. 8. MIGA optimization results for dynamic JIT-VSW in the debutanizer column process.

4.3 Industrial rubber mixing process

4.3.1 Process description

Rubber mixing is an important production process in the rubber and tire industry (Yang et al., 2016; Liu et al., 2016). The industrial rubber mixing process under study is practiced in a tire company in east China. The diagram of the rubber mixing process is shown in Fig. 9. According to the technical formula, various materials are fed into the raw rubber to produce synthetic rubber; during this process, numerous complex chemical reactions take place in an internal mixer. In order to obtain high-quality rubber products, it is necessary to measure the quality variables of mixed rubber over time to perform timely analysis and correction measures when an exception occurs. Generally, the Mooney viscosity is a crucial quality index when monitoring the rubber mixing process. However, in real rubber mixing processes, the Mooney viscosity is measured through a viscometer after one batch has discharged, which often takes about 4~6 h. As a result, the optimal and uniform quality of mixed rubber is difficult to obtain and plenty of raw materials may be wasted without full reaction. Therefore, it is desirable to construct a soft sensor model for online measurement of the Mooney viscosity in real-time. The process variables i.e., temperature in the mixer chamber, motor power, ram pressure, motor speed and energy are used as the input variables for soft sensor development.



(a)

(b)

Fig. 9. Industrial rubber mixing process: (a) monitoring interface and (b) industrial rubber mixers.

4.3.2 Online prediction results of Mooney viscosity

With a sampling interval of 2 s, a total of 1172 batches are collected from three internal mixers. First, all batch data are divided into three sets: 822 batches as the training set, 175 batches as the validation set and 175 batches as the testing set. Then, the delayed and undelayed variables with corresponding times 0 s, 14 s, 18 s, 22 s,…,118 s are used as the input variables of soft sensors. Consequently, a total of 140 variables are obtained as input variables and the Mooney viscosity is selected as the output variable.

Comparison results of the optimal online prediction and the best parameters for different soft sensor methods are tabulated in Table 4. Meanwhile, Fig. 13 shows the detailed prediction results of Mooney viscosity using JIT-VSW. From Table 4, we can see that, the proposed JIT-VSW soft sensor method obtains the lowest RMSE and maximum R^2. The superior prediction performance of JIT-VSW compared to other methods is mainly attributed to the combination of input variable selection and MWS similarity measure.

Table 4 Prediction results from different soft sensors in the rubber mixing process.

Method	L	R	φ	N_{mix}	RMSE	R^2
PLS	-	7	-	-	7.5584	0.7875
LWPLS	50	2	0.1	-	4.5682	0.9224
IVS-LWPLS	50	1	0.1	-	3.9012	0.9434
MWS-LWPLS	50	1	0.1	2	4.0077	0.9403
JIT-VSW	50	2	0.1	6	3.5472	0.9532

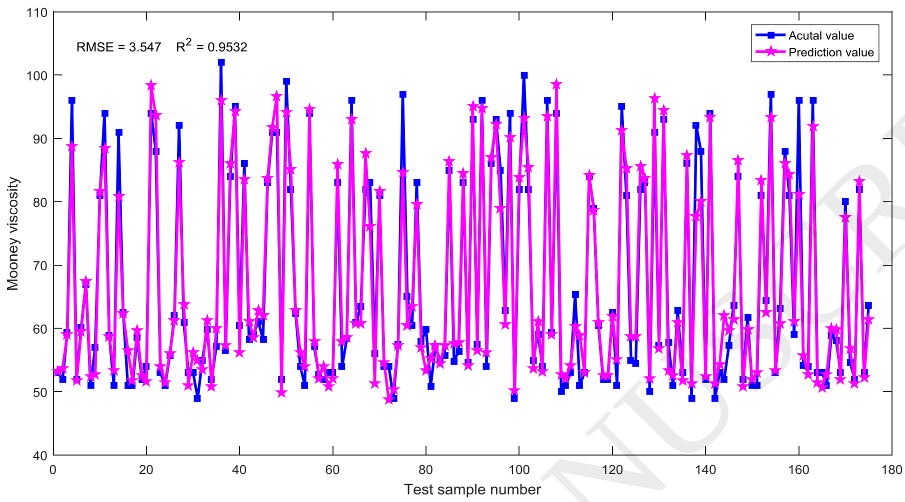


Fig. 10. Comparison of predicted and actual values using the JIT-VSW soft sensor method in the rubber mixing process.

Again, Fig. 11 shows the corresponding optimization results of input variable selection and MWS parameters. Almost 80 features are excluded from the initial input variables, which substantially decreases the computational load as well as increasing the prediction accuracy. Moreover, the weights of the input variables clearly differ from each other and the component WED similarity measures contribute to the mixed similarity by assigning different mixture coefficients.

The above experimental results confirm that the proposed JIT-VSW is more accurate and reliable than traditional JIT soft sensors for the prediction of Mooney viscosity in the rubber mixing process.

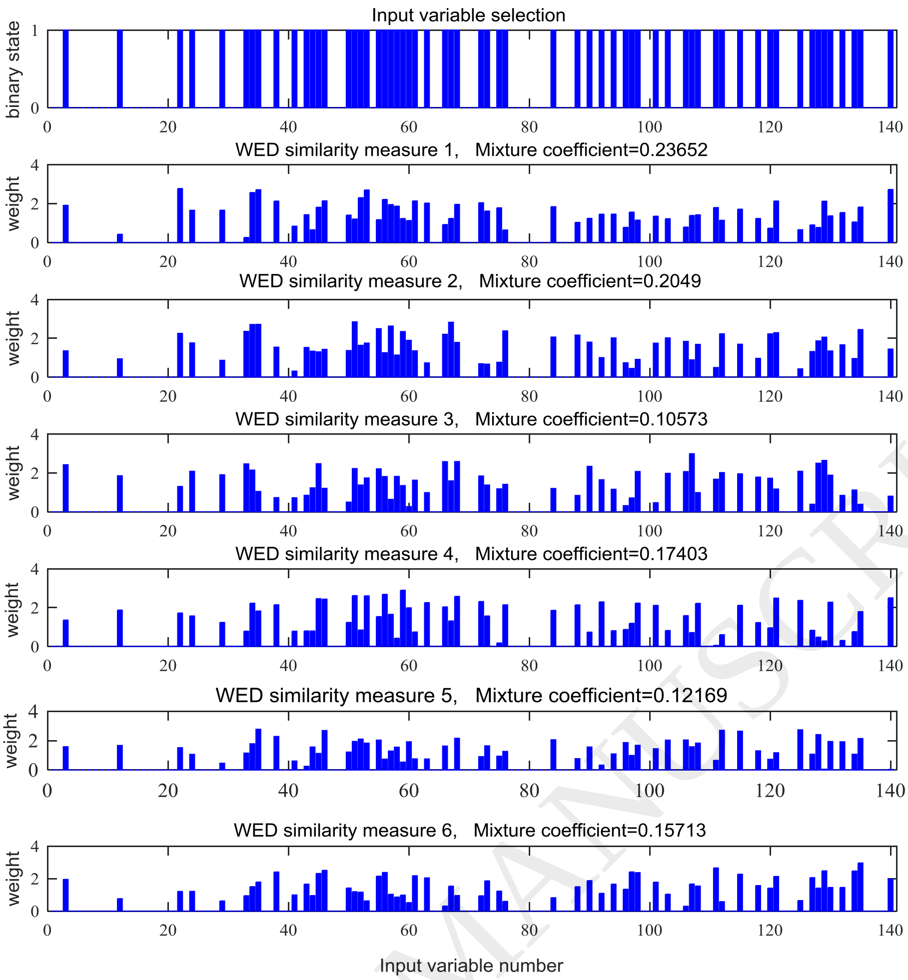


Fig. 11. MIGA optimization results in the rubber mixing process.

5 Conclusions

In this research, a new soft sensor approach, JIT-VSW, is proposed by exploiting input variable selection and weighting. To enhance the performance of relevant sample selection, a mixture-type similarity measure MWS is defined by combining multiple WED similarity measures through a weighted average strategy. Further, input variable selection and MWS parameters are optimized simultaneously using a mixed integer optimization approach, MIGA. By employing the MWS similarity measure and performing input variable selection, the proposed JIT-VSW approach enables the selection of relevant samples to query data more efficiently and handle process nonlinearity better than traditional JIT learning methods; this leads to significantly improved prediction performance. Experiments on three real-world application examples show that JIT-VSW can effectively exploit the input variable selection and weighting to improve the estimation accuracy of JIT learning soft sensors using evolution optimization approaches. Although this study employs LWPLS as the base learner in the case studies, the proposed JIT-VSW soft sensing framework can also utilize other base learners. In addition to soft sensor applications, our proposed method can be employed to address other modeling problems in nonlinear chemical and biological systems.

Notes

The authors declare no competing financial interest.

Acknowledgements

This work was supported by the National Science Foundation of China (61763020), the Scientific Research Foundation Project of Yunnan Education Department (2017ZZX149), and the Applied Basic Research Project of Yunnan Province (2018FD040).

References

Fortuna, L., Graziani, S., Rizzo, A., Xibilia, M.G., 2007. Soft Sensors for Monitoring and Control of Industrial Processes. Springer London.

Haimi, H., Mulas, M., Corona, F., Vahala, R., 2013. Data-derived soft-sensors for biological wastewater treatment plants: An overview. Environmental Modelling & Software 47, 88-107.

Kano, M., Nakagawa, Y., 2008. Data-based process monitoring, process control, and quality improvement: Recent developments and applications in steel industry. Computers & Chemical Engineering 32, 12-24.

Kadlec, P., Gabrys, B., Strandt, S., 2009. Data-driven Soft Sensors in the process industry. Computers & Chemical Engineering 33, 795-814.

Ge, Z., Song, Z., Gao, F., 2013. Review of Recent Research on Data-Based Process Monitoring. Industrial & Engineering Chemistry Research 52, 3543-3562.

Ge, Z., 2017. Review on data-driven modeling and monitoring for plant-wide industrial processes. Chemometrics & Intelligent Laboratory Systems 171.

Shao, W., Tian, X., Wang, P., Deng, X., Chen, S., 2015. Online soft sensor design using local partial least squares models with adaptive process state partition. Chemometrics & Intelligent Laboratory Systems 144, 108-121.

Ge, Z., Song, Z., 2014. Ensemble independent component regression models and soft sensing application. Chemometrics & Intelligent Laboratory Systems 130, 115-122.

Kaneko, H., Funatsu, K., 2014. Adaptive soft sensor based on online support vector regression and Bayesian ensemble learning for various states in chemical plants. Chemometrics & Intelligent Laboratory Systems 137, 57-66.

Jin, H., Chen, X., Yang, J., Zhang, H., Wang, L., Wu, L., 2015. Multi-model adaptive soft sensor modeling method using local learning and online support vector regression for nonlinear time-variant batch processes. Chemical Engineering Science 131, 282-303.

Gonzaga, J.C.B., Meleiro, L.A.C., Kiang, C., Filho, R.M., 2009. ANN-based soft-sensor for real-time process monitoring and control of an industrial polymerization process. Computers & Chemical Engineering 33, 43-49.

Xiong, W., Li, Y., Zhao, Y., Huang, B., 2017. Adaptive Soft Sensor Based on Time Difference Gaussian Process Regression with Local Time-delay Reconstruction. Chemical Engineering Research & Design 117.

Mei, C., Su, Y., Liu, G., Ding, Y., Liao, Z., 2017. Dynamic soft sensor development based on Gaussian mixture regression for fermentation processes. Chinese Journal of Chemical Engineering 25, 116-122.

Kadlec, P., Gabrys, B., 2011. Local learning‐based adaptive soft sensor for catalyst activation prediction. Aiche Journal 57, 1288-1301.

Jin, H., Chen, X., Yang, J., Wu, L., Chen, X., Yang, J., Wu, L., 2014. Adaptive soft sensor modeling framework based on just-in-time learning and kernel partial least squares regression for nonlinear multiphase batch processes. Computers & Chemical Engineering 71, 77-93.

Liu, Y., Zhang, Z., Chen, J., 2015. Ensemble local kernel learning for online prediction of distributed product outputs in chemical processes. Chemical Engineering Science 137, 140-151.

Wang, Z.X., He, Q.P., Wang, J., 2015. Comparison of variable selection methods for PLS-based soft sensor modeling. Journal of Process Control 26, 56-72.

Liu, Y., Gao, Z., Li, P., Wang, H., 2012. Just-in-Time Kernel Learning with Adaptive Parameter Selection for Soft Sensor Modeling of Batch Processes. Industrial & Engineering Chemistry Research 51, 4313-4327.

Liu, Y., Huang, D., Li, Y., 2012. Development of Interval Soft Sensors Using Enhanced Just-in-Time Learning and Inductive Confidence Predictor. Industrial & Engineering Chemistry Research 51, 3356–3367.

Kano, M., Fujiwara, K., 2013. Virtual Sensing Technology in Process Industries : Trends and Challenges Revealed by Recent Industrial Applications. Journal of Chemical Engineering of Japan 46, 1-17.

Kim, S., Kano, M., Hasebe, S., Takinami, A., Seki, T., 2013. Long-Term Industrial Applications of Inferential Control Based on Just-In-Time Soft-Sensors: Economical Impact and Challenges. Industrial & Engineering Chemistry Research 52, 12346-12356.

Jin, H., Chen, X., Yang, J., Wang, L., Wu, L., 2015. Online local learning based adaptive soft sensor and its application to an industrial fed-batch chlortetracycline fermentation process. Chemometrics & Intelligent Laboratory Systems 143, 58-78.

Yuan, X., Ge, Z., Huang, B., Song, Z., 2016. A Probabilistic Just-in-Time Learning Framework for Soft Sensor Development With Missing Data. IEEE Transactions on Control Systems Technology PP, 1-9.

Yuan, X., Ge, Z., Huang, B., Song, Z., Wang, Y., 2016. Semi-supervised JITL framework for nonlinear industrial soft sensing based on locally semi-supervised weighted PCR. IEEE Transactions on Industrial Informatics PP, 1-1.

Cheng, C., Chiu, M.S., 2004. A new data-based methodology for nonlinear process modeling. Chemical Engineering Science 59, 2801-2810.

Fujiwara, K., Kano, M., Hasebe, S., Takinami, A., 2009. Soft‐sensor development using correlation‐based just‐in‐time modeling. Aiche Journal 55, 1754–1765.

Fan, M., Ge, Z., Song, Z., 2014. Adaptive Gaussian Mixture Model-Based Relevant Sample Selection for JITL Soft Sensor Development. Industrial & Engineering Chemistry Research 53, 19979-19986.

Xie, L., Zeng, J., Gao, C., 2014. Novel Just-In-Time Learning-Based Soft Sensor Utilizing Non-Gaussian Information. IEEE Transactions on Control Systems Technology 22, 360-368.

Kim, S., Okajima, R., Kano, M., Hasebe, S., 2013. Development of soft-sensor using locally weighted PLS with adaptive similarity measure. Chemometrics & Intelligent Laboratory Systems 124, 43-49.

Shigemori, H., Kano, M., Hasebe, S., 2011. Optimum quality design system for steel products through locally weighted regression model. Journal of Process Control 21, 293-301.

Kim, S., Kano, M., Nakagawa, H., Hasebe, S., 2011. Estimation of active pharmaceutical ingredients content using locally weighted partial least squares and statistical wavelength selection. International Journal of Pharmaceutics 421, 269-274.

Marler, R.T., Arora, J.S., 2004. Survey of multi-objective optimization methods for engineering. Structural & Multidisciplinary Optimization 26, 369-395.

Jin, Y., Sendhoff, B., 2008. Pareto-Based Multiobjective Machine Learning: An Overview and Case Studies. IEEE Transactions on Systems Man & Cybernetics Part C 38, 397-415.

Rangaiah, G.P., Bonilla-Petriciolet, A., 2013. Multi-Objective Optimization in Chemical Engineering: Developments and Applications. Wiley.

George, E., 2000. The Variable Selection Problem. Publications of the American Statistical Association 95, 1304-1308.

Jian, W., Zhu, L., Xu, Z., Chen, X., 2017. A variable selection method for soft sensor development through mixed integer quadratic programming. Chemometrics & Intelligent Laboratory Systems 167.

Whitley, D., 1994. A genetic algorithm tutorial. Statistics and computing, 65-85.

Deep, K., Singh, K.P., Kansal, M.L., Mohan, C., 2009. A real coded genetic algorithm for solving integer and mixed integer optimization problems. Applied Mathematics & Computation 212, 505-518.

Wold, S., Sjöström, M., Eriksson, L., 2001. PLS-regression: a basic tool of chemometrics. Chemometrics & Intelligent Laboratory Systems 58, 109-130.

Birol, G., Ündey, C., Çinar, A., 2002. A modular simulation package for fed-batch fermentation: penicillin production. Computers & Chemical Engineering 26, 1553-1565.

Zhao, C., Wang, F., Gao, F., 2009. Improved calibration investigation using phase-wise local and cumulative quality interpretation and prediction. Chemometrics & Intelligent Laboratory Systems 95, 107-121.

Yang, K., Jin, H., Chen, X., Dai, J., Wang, L., Zhang, D., 2016. Soft sensor development for online quality prediction of industrial batch rubber mixing process using ensemble just-in-time Gaussian process regression models. Chemometrics & Intelligent Laboratory Systems 155, 170-182.

Liu, Y., Fan, Y., Zhou, L., Jin, F., Gao, Z., 2016. Ensemble Correntropy - Based Mooney Viscosity Prediction Model for an Industrial Rubber Mixing Process. Chemical Engineering & Technology 39, 1804-1812.