A New AdaBoost.IR Soft Sensor Method for Robust Operation Optimization of Ladle Furnace Refining

Hui-Xin TIAN,1,2* Yu-Dong LIU,1,2) Kun LI,3) Ran-Ran YANG1,2) and Bo MENG1,2)

1) School of Electrical Engineering & Automation, Tianjin Polytechnic University, Tianjin, 300387 China.
2) Key Laboratory of Advanced Electrical Engineering and Energy Technology, Tianjin Polytechnic University, Tianjin, 300387 China.
3) School of Management, Tianjin Polytechnic University, Tianjin, 300387 China.

(Received on June 22, 2016; accepted on February 2, 2017; J-STAGE Advance published date: April 4, 2017)

LF (Ladle Furnace) refining plays an important role during secondary metallurgic process. The traditional LF refining operation relies on the workers’ experience, it is disadvantageous to ensure the stable production, high-quality products and energy saving. A new robust operation optimization method of molten steel temperature based on AdaBoost.IR soft sensor is proposed in LF refining process. Firstly, an intelligent model based on BP (Back Propagation) neural network is established by analyzing the changes of energy during whole refining process of LF as sub intelligent model. Then an AdaBoost.IR is designed for the characters of industrial data, and is suitable for industrial soft sensor modeling. The ensemble soft sensor model is established for realizing the online real time measurement of molten steel temperature with better accuracy by using AdaBoost.IR. Secondly the robust operation optimization model is described by analyzing the process of LF refining based on above AdaBoost.IR soft sensor. And the HPOS-GA is used to solve the optimal operation solution of robust optimization model. The new robust operation optimization of temperature based on AdaBoost.IR soft sensor is proposed in LF refining process. Firstly, an intelligent model and solving optimal solution. Due to the LF refining process is complex and changed, the optimal operation parameters need to overcome the fluctuating of production. Therefore, a robust operation optimization model should be established for the robust optimal solution.

KEY WORDS: robust operation optimization; molten steel temperature; LF refining; soft sensor; AdaBoost.IR.

1. Introduction

The steel industry faces stiff competition in the global market. The steel companies have to focus on how to reduce cost and energy consumption while produce high quality products satisfying various customer demand.1) They must ensure the stable continuous production and efficient operation in essential. The process of secondary metallurgic is very important to improve the quality of products. Ladle furnace (LF) steel refining plays a substantial role in secondary metallurgic process, whose objective is to produce qualified steel grades.2) Another important purpose of LF is to ensure the temperature of molten steel for continuous casting. Therefore, it is necessary to optimize the LF refining process for stable continuous production and high quality products. In order to optimize the operations of molten steel temperature, the temperature should be measured in real time. But in real LF refining process, the temperature is measured no more than three times generally using expendable thermocouples. So, the development of a soft sensor model based on historical production data to predict the molten steel temperature in LF is particularly concerned. On the basis of real time predicted temperature, the best operation parameters of LF refining should be calculated by building an optimization model and solving optimal solution. Due to the LF refining process is complex and changed, the optimal operation parameters need to overcome the fluctuating of production. Therefore, a robust operation optimization model should be established for the robust optimal solution.

Generally, the soft sensor methods of predicting the temperature of molten steel in LF can be mainly divided into two kinds: mechanistic method and intelligent method. The mechanistic method is mainly based on thermodynamics and conservation of energy.2) In early studies, mechanistic models are developed in a variety of different ways.3–5) However, these models cannot be used for on-line prediction as the parameters were hard to obtain. It is attributed to the harsh operating environment of ladle metallurgy especially the high temperatures and corrosive slag associated with the process. With the fast development of artificial intelligence, more and more intelligence algorithms are used to establish the prediction models in the process of industrial production.6,7) Almost all the intelligent models are designed to obtain the information from the historical production data. So the intelligent method can overcome the limitations about parameters of mechanistic model. However, in practical applications of industrial production, the intelligent models based on single algorithm are unstable because the complex industrial process and the complexity of indus-

* Corresponding author: E-mail: icedewl@163.com
DOI: http://dx.doi.org/10.2355/isijinternational.ISIJINT-2016-371
trial production data. Ensemble technique that combined the predictors is an efficient strategy for achieving high performance of prediction, especially in fields where the development of a powerful single predictor system requires considerable efforts, and has received much attention by researchers. In recent decade years the ensemble technique has been developing to the field of regression problems. The regression ensemble strategy can overcome above limitations and improve generalization ability and prediction accuracy. Most of researchers of regression ensemble focus on the ensemble theory. The research about how to fit the complex industrial data and satisfy the needs of complex industrial process is scarcely.

Besides, the robust operation optimization is also the key to realize the stable molten steel temperature. The robust optimization model needs to be built using the prediction temperature by soft sensor model. In view of the robust optimization problem in the process of industrial production, most of researches are concentrated on petrochemical industry and chemical industry. Shimoyalna etc. proposed the linear additive expectation and variance of the original fitness function as the objective function of the robust optimization problem. Jin considered both the performance and robustness index in the robust optimization problems. The original fitness function regard serves as performance index and the robust index represented by the fitness function of the expected value or variance. M. Vallerioa et al. used sigma point method to resolve the robust multi-objective dynamic optimization problem of chemical processes. Qi Zhang et al. proposed an adjustable robust optimization approach for the problems in continuous industrial processes. But the robust optimization in LF refining is studied scarcely. It is necessary to research the robust optimization problem of molten steel temperature in LF considering the characteristics of the LF refining process and the requirements of end molten temperature.

Against above background, the process of LF refining is analyzed to find the main factors that affect the molten temperature. According to the main factors, the soft sensor model based on the BP network is established. For accurate and efficient prediction of molten steel temperature, we proposed a new AdaBoost strategy for industrial soft sensor (AdaBoost.IR), which will fit the industrial process data well. The AdaBoost.IR can aggregate several soft sensors based on single BP network to overcome the noise of industrial process data, and obtain useful information, which is helpful in optimizing operations during LF refining. Then a robust operation optimization model of molten steel temperature in LF refining process is established on the basis of the AdaBoost.IR ensemble soft sensor of temperature considering the fluctuation of production operation, which obtained by analyzing the refining process of LF. At last, in order to obtain a good robust optimal solution, a hybrid algorithm based on particle swarm optimization and genetic algorithm (HSPSO-GA) is used. The new robust operation optimization method of molten steel temperature can optimize and guide the operation of LF refining for controlling end temperature. At the same time the stable LF refining production will be ensured effectively. Then the LF refining quality will be improved, and it is helpful to saving energy and reducing consumption.

The article is organized as following. In Sect. 2, an intelligent temperature prediction model based on BP neural network is established by analyzing the refining process of LF. In Sect. 3, aiming at the characters of industrial process data, a new ensemble method AdaBoost.IR is proposed to build the ensemble soft sensor model of molten steel temperature for improving the performance of prediction. In Sect. 4, the robust operation optimization model of molten steel temperature in LF refining process is described in detail. And a solving algorithm HSPSO-GA is used to find the robust optimal solution. In Sect. 5, the proposed methods of AdaBoost.IR soft sensor and robust operation optimization are experimented by 300 t LF of Baosteel. Finally, the conclusions are given in Sect. 6.

2. Intelligent Temperature Prediction Model Based on BP Neural Network

At present, Ladle furnace is used extensively in the iron and steel industry. The main purpose of ladle furnace treatment is to produce qualified steel with desired temperature and chemical composition for specified steel grades, when the ladle is taken over at downstream secondary metallurgy units or at a continuous caster. The process of LF refining is showed in Fig. 1. The refining production process includes step 1 to step 7. In practical LF refining process, the molten steel temperature should be measured by expendable thermocouples two or three times in most cases. It is to say that there is short of online measurement instrument for obtaining the molten steel temperature. If the results of sample and temperature measurement does not meet the requirements of qualified steel grades in step 6 sometime, the step 5 and 6 will be operated again. In order to avoid the unnecessary operation and consumption, the molten steel temperature soft sensor model will be established to realize online in real time measurement in the following paragraphs.

2.1. Analysis of Main Factors

In order to establish the intelligent prediction model of temperature, the whole LF refining process is considered as an energy conservation system. Traditionally, the metallurgic process of LF is from the ladle entry to the ladle exit (Fig. 1). The whole LF refining process is an energy conservation system, whose input energy is equal to the output energy. However, in practical refining process, the temperature is not measured every time at the beginning of the process. The measurement may be done before or after the power turned on. The energy change from ladle entry to the first temperature measurement is neglected. Therefore, in order to ensure the balance of energy in whole system, we choose the time of the last temperature measurement before power supply as the start time of the energy conservation system, and this temperature is the initial temperature. Similarly, the ending time of the energy conservation system is the time of end temperature measurement. The energy conservation system of LF refining process is showed in Fig. 1. The changed energy in energy conservation system is considered to find the correlative factors that affect molten steel temperature. By analyzing the thermodynamics and conservation of the energy during the LF metallurgic process, the energy gain of LF is mainly due to the electric
The energy loss mainly includes the following three sections: the first section is the heat exchanges between ladle furnace and surroundings, which include the ladle refractory wall and the top surface. The energy loss in this section is relatively stable. It will increase with time. Therefore, this loss energy may be reflected by refining time. The second section is the changed energy for additions, which includes the sum of heat exchanges and chemical reaction heats. In this section, the changed energy by various metal alloys can be calculated by the parameters in Table 1 for 300 t ladle furnace. In addition, the slag adding also affects the changed energy. The third section is the energy loss by argon purging. To sum up, the main factors that affect the molten steel temperature are as followed: the refining power consumption, the initial temperature, the heat effects of metal alloy additions, the adding amount of slag, the volume of argon purging, the weight of molten steel and the refining time.

### 2.2. Soft Sensor Model of Molten Steel Temperature Based on BP Network

For the purpose of building intelligent sub soft sensor model of molten steel temperature, the BP (back propagation) neural network is applied. BP network is the most widely used model of the neural network paradigm and has been applied successfully in many application studies in a broad range of areas. BP networks are multi-layered feed-forward neural networks that are trained using the error BP procedure, a supervised mode of training. The architecture of BP network consists of an input layer, a hidden layer and an output layer. By above analysis, there are seven main factors that affect the temperature as potential inputs of neural network, while the molten steel temperature is the single output of soft sensor model. For better performance and accuracy of prediction model, several BP networks will be aggregated in next section.

### 3. Ensemble Temperature Prediction Model Based on AdaBoost.IR

Boosting is one of the most efficient ensemble strategies. It was used in classification problems at the earlier stage. Many researchers have proved that Boosting can solve the classification problems effectively. The boosting algorithms are used widely in the field of classification. However, the researches about boosting algorithm in regression problems are much less than the ones in classification problems. Schapire and Freund proposed the AdaBoost.R algorithm to generate regression models. Drucker developed the AdaBoost.R2 algorithm, which is an ad hoc modification of AdaBoost.R. He conducted some experiments firstly for regression problems and obtained good results. But in traditional AdaBoost.R2, the important loss function must be not more than 0.5, if not the AdaBoost.R2 will be end. So the application of AdaBoost.R2 in industrial prediction modeling field is highly limited. Furthermore, the training sample inevitably contains the data with big fluctuation in industry process. But the AdaBoost.R2 is sensitive to noise because the weight update is proportional to average loss function. It cannot fit the need of industrial production. Aiming at shortages of the loss function in AdaBoost.R2, Solomatine and Shrestha proposed the AdaBoost.RT algorithm to solve the regression problems more efficiently. In AdaBoost.RT, the so-called absolute relative error threshold is introduced to project training examples into two classes (poorly and well-predicted examples). And the harder examples will be given more chance to be trained. So the threshold is very important for the accuracy of prediction. How to design the

#### Table 1. Temperature drop coefficients of metal alloy in 300 t LF.

| Addition   | temperature effects parameter $\times 10^{-2}$ (°C/kg) | Addition   | temperature effects parameter $\times 10^{-2}$ (°C/kg) |
|------------|------------------------------------------------------|------------|------------------------------------------------------|
| HcFeCr     | $-0.95$                                              | LcFeCr     | $-0.65$                                              |
| HcFeMn     | $-0.9$                                               | LcFeMn     | $-0.75$                                              |
| FeMo       | $-0.75$                                              | FeSi       | $+0.9$                                               |
| Al         | $+5.0$                                               | C          | $-2.5$                                               |
| FeNb       | $-0.35$                                              | Ni         | $-0.5$                                               |
| CaSi       | $-1.05$                                              | FeTi       | $-0.4$                                               |
| HCZ1       | $-1.0$                                               | FeAl       | $+1.0$                                               |
| LCCR       | $-0.65$                                              | Al-D       | $-0.5$                                               |
value of threshold of AdaBoost.RT becomes a new problem which the users have to face. Tian Huixin et al. proposed an adaptive threshold method to improve the AdaBoost.RT. But the new parameters have to be chosen before using AdaBoost.RT. In a word, the suitable parameters are the important guarantee for good performance of AdaBoost.R. But both traditional AdaBoost.R and improved AdaBoost.R have to face the problems of setting parameters. It leads to the fact that the methods of AdaBoost.R is hard to use in practical industry production. Therefore, we must try to get rid of the blindness of setting parameters without mechanism analysis in industry production.

Aiming at above shortages of traditional AdaBoost.R algorithm, an AdaBoost.IR algorithm is proposed for industrial soft sensor regression problems with easier operation. A slack variable $\delta$ is introduced to distinguish between well samples and poor samples in new algorithm. The slack variable $\delta$ can be designed according to the allowable maximum absolute error (AE) of outputs in the application of industrial production (LF refining process). It is so easy for industry users by simple mechanism analysis. So it can overcome the blindness of setting parameters in other AdaBoost.R methods. The operation of AdaBoost.IR also becomes easier than the one of traditional AdaBoost.R for users in practical industries. In new AdaBoost.IR, the weights of data will be changed according to the value of absolute relative error (ARE). The weight of poor sample will be increased greatly. But if the poor sample is the one with bigger noise, the weight will be increased slowly. If the noise of samples is big enough (ARE > 1), the weight will be decreased. It is to say, the new AdaBoost.IR will pay more attention to some of the poor samples whose information is hard to be obtained by previous training. And the other poor samples with big noise will be ignored gradually in later training iterations. Therefore, the new strategy of AdaBoost.IR is more suitable for the need of industrial production. It provides a more simple operation for industry users to establish accurate prediction model by using the theory of AdaBoost.R.

The details of new AdaBoost.IR are described as follows:

**Input:**
1) Sequence of $m$ examples $S = \{(a_i, b_i), \ldots (a_m, b_m)\}$, where output be $\mathbb{R}$.
2) Design a super prediction learning machine.
3) Integer $T$ specifying number of iterations (machines).
4) The slack variable $\delta$.

**Initialize:**
1) $t=1$.
2) $D_t(i) = 1/m$, $\forall i = 1, 2, \ldots, m$. Here $D_t(i)$ is the weight of $(a_i, b_i)$.

**For:** while $t \leq T$
1) Choose $k$ data ($\text{train\_data}$) from $S$ according to $D_t(i)$, $k \leq m$.
2) Train the learning machine and obtain a prediction model: $g_t(a) \rightarrow b$.
3) Calculate the absolute error of every data $(a_i, b_i)$:
   \[ AE_t(i) = |g_t(a_i) - b_i| \]  \hspace{1cm} (1)
4) Calculate the absolute relative error of every data $(a_i, b_i)$:
   \[ ARE_t(i) = \frac{|g_t(a_i) - b_i|}{b_i} \]  \hspace{1cm} (2)
5) Calculate the root mean square error of $(a_i, b_i)$ according to the prediction model:
   \[ RMSE_t = \sqrt{\frac{\sum_{i=1}^{k} (g_t(a_i) - b_i)^2}{k}} \]  \hspace{1cm} (3)
6) Update the weight:
   \[ D_{t+1}(i) = D_t(i) \times \begin{cases} \frac{1}{\text{ARE}_t(i)} & \text{if } (a_i, b_i) \text{ is not chosen} \\ 1 & \text{if } (a_i, b_i) \text{ is chosen} \end{cases} \]  \hspace{1cm} (4)

Here the $Z_t$ is a normalization factor.
7) set $t = t + 1$.

**Output:** Output the final hypothesis:
\[ g_m(x) = \frac{\sum_{i=1}^{T} \frac{1}{RMSE_t} g_t(a_i)}{\sum_{i=1}^{T} \frac{1}{RMSE_t}} \]  \hspace{1cm} (5)

In the beginning, the $m$ examples $S = \{(a_1, b_1), \ldots (a_m, b_m)\}$ are supplied as data set. The molten steel soft sensor model based on the BP network in section 2 is used as super prediction learning machine. The parameters $T$ is selected which is the iterations of algorithm. The slack variable $\delta$ is selected according to the characters of output of prediction model in practical industrial process. $D_t(i) = 1/m(k)$ is distributed to each sample $(a_i, b_i)$ of $S$ initially, here $i = 1, 2, \ldots, m$.

In each iteration $t = 1, 2, \ldots, T$, the training data $\text{train\_data}_t = \{(a_1, b_1), (a_2, b_2), \ldots (a_m, b_m)\}$, $k \leq m$ are selected randomly from the dataset according to the $D_t(i)$, $i = 1, 2, \ldots, m$. The $\text{train\_data}_t$ is used to train the BP prediction learning machine, and a soft sensor model $g(x) = \hat{b}$ will be obtained.

Then the absolute error (Eq. (1)), absolute relative error (Eq. (2)) and RMSE (Eq. (3)) of every training data can be calculated. According to the $\text{RMSE}_t$, the weight $D_{t+1}(i)$ is updated as Eq. (4).

There are two relationships between slack variable $\delta$ and $k$ samples (which have been chosen to train): 1) When $AE_t(i) \leq \delta$ it means the sample is trained well, and we call it well-predicted examples. Here the $ARE_t(i)$ is in (0, 1). So the weight of sample $i$ will multiply $ARE_t(i)$ to decrease its weight. 2) When $AE_t(i) > \delta$, it means the sample is trained not well in the iteration, and we call it poorly samples. If $ARE_t(i) \leq 1$ and the $ARE_t(i)$ is bigger, $1/ARE_t(i)$ is smaller. Here the weight will be changed with a small increase. In another case, if $ARE_t(i) > 1$, it means the sample with big noise is an outlier. Here the weight of sample will be decrease for less training times. In a word, in next iteration the weights of poor sample (not including outliers) will be increase for more training. And the interference of outliers is avoided efficiently.

After $T$ hypotheses being generated for each $\text{train\_data}$
database, the RMSE is used to reflect the performance of sub training machine (BP network). So the final hypothesis \( g_{\text{fin}} \) is obtained by the weighted majority voting of all composite hypotheses as Eq. (5).

### 4. Robust Operation Optimization of Molten Steel Temperature in LF

In practical LF refining process, the operation is controlled by the operators’ experiences, which leading to low production efficiency and high cost. And the workers with long-term operating experience must have worked for many years. So it is urgent to establish an optimal operation model and solve it for obtaining the best operating parameters. Thanks to these parameters, the stable production will be ensured. In addition, owing to the constantly changing refining process and environment, and the restrictions of engineering, it is hard to design a suitable optimal operation model. It is to say we have to face the uncontrollable errors or noise during the LF refining process unavoidably. In this case, the small changes of design parameter will lead to a failed objective function. And the globally optimal solution is meaningless because it will become sensitive to small changes. Therefore, it is necessary to consider the robust optimization solution of LF refining process.

#### 4.1. Robust Optimization Model

Considering the stability of production, the optimization model is described based on above temperature soft sensor as follows:

\[
F(X) = \min |\hat{y} - y|
\]

subject to \( x_{i-\text{min}} \leq x_i \leq x_{i-\text{max}} i = 1,2,\cdots,n \) …… (6)

where, \( F(X) \) is the objective function to be minimized, \( \hat{y} = f(X), X = [x_1,x_2,\cdots,x_4] \). \( f(X) \) is the prediction molten steel temperature by AdaBoost.IR soft sensor, \( f(X) \) is the objective molten steel temperature. \( x_1,x_2,\cdots \) are the operation parameters, \( x_{i-\text{min}} \) and \( x_{i-\text{max}} \) are the lower and upper bounds of \( x_i \).

On the other hand, the lower energy consumption is also an important objective in the process of LF refining for steel companies. So the lowest refining power consumption objective is added to the optimization model of molten steel temperature. The optimization objectives of stability of production and lower energy consumption are mutual restrictive sometimes. For this kind of problems, an approximate Pareto front will be obtained by multi-objective optimization algorithm. There are many non-inferior solutions. The suitable solution must be chosen by operators with rich experience. It will lead to fluctuations of production when operators are changed. For overcoming above shortage, we describe the multi-objective optimization model as follow:

\[
F(X) = \min \{\alpha (\hat{y} - y) + (1 - \alpha) x_i\}
\]

subject to \( x_{i-\text{min}} \leq x_i \leq x_{i-\text{max}} i = 1,2,\cdots,7 \) …… (7)

where \( 0 < \alpha < 1 \).

Additionally, LF refining production process includes many physical and chemical reactions in ladle. The parameters of operation will move up or down during the refining. Here we further consider the fluctuations of controllable parameters that received according to the research in the second section. For a given solution (i.e. a set of control variables), 10 samples with fluctuations are generated randomly. Then the objective function values of the solutions are calculated according to each sample. And finally the average and variance of these 10 objective function values are obtained as the evaluation of this solution. The description of robust objective optimization model is:

\[
F(X) = \min \{\alpha (1 - \beta) f_{AE}(X + \sigma) + \beta f_{E}(X + \sigma)\} + (1 - \alpha) x_i \}
\]

subject to \( x_{i-\text{min}} \leq x_i \leq x_{i-\text{max}} i = 1,2,\cdots,7 \) …… (8)

Where, \( F(X) \) is the robust objective function to be minimized, \( f_{AE} \) and \( f_{E} \) respectively are the average error and variance of a parameters’ set \( X \), the fluctuation \( \sigma \) of each parameter is assumed to be within 5%. \( \beta \) is an adaptive coefficient. It will linearly change with the increase of iterations. Here the Monte Carlo method is used to evaluate the performance of a solution.

Through above analysis of main factors that affect the molten steel temperature in section 2, the factors can be classified into controllable variable and non-controllable variable. They are described in detail in Table 2. \( x_5, x_6, \) and \( x_7 \) are regarded as non-controllable variable. And \( x_1,\cdots,x_4 \) are controllable during the whole LF refining process. Their fluctuation scope is also showed in Table 2.

#### 4.2. HPSO-GA for Robust Optimal Solution

After building above robust multi-objective optimization model, the robust optimal solution of the model should be solved. Genetic Algorithm (GA) is one of the well-known evolutionary algorithms\(^\text{17}\) and used widely to solve the optimal solution in various fields. In GA, a population of potential solutions to some optimization problem can be maintained, and new solutions will be generated by using a variety of genetic operators including recombination, selection and mutation during each iteration. GA has many good characters, such as high parallelism, strong randomness and adaptive ability. But it also has some shortages. For example, the speed of convergence is influenced by initial values,

| Table 2. The list of operation parameters. |
|------------------------------------------|
| \( x_1 \) | operation parameter | controllable or non-controllable | fluctuation scope |
| \( x_2 \) | refining power consumption | controllable | 0–220 |
| \( x_3 \) | refining time | controllable | 35–900 |
| \( x_4 \) | adding amount of slag | controllable | −7 000–10 000 |
| \( x_5 \) | volume of argon purging | controllable | 0–33 000 |
| \( x_6 \) | heat effects of metal alloy additions | non-controllable | - |
| \( x_7 \) | initial temperature | non-controllable | - |
| \( x_8 \) | weight of molten steel | non-controllable | - |

\( 2017 \) ISIJ5
and the performance of global optimization is not good enough. So the global optimal solution may not be found sometimes especially for the multi-objective optimization problems. Moreover, PSO (Particle Swarm Optimization) has fast convergence speed, good global performance and good anti-interference performance. So it is very suitable that PSO is combined with GA to overcome the shortages of traditional GA. The core idea of PSO is that if a particle discovers a promising new solution, all the other particles will move closer to it. Then the region will be explored more thoroughly in the process. Each particle in the swarm is updated using Eqs. (9) and (10). Here the swarm consists of \( n \) particles, and \( r_1 \) and \( r_2 \) are in the range (0,1), then

\[
v_{i,d}^{k+1} = \omega v_{i,d}^{k} + c_1 r_1 \left( p_{i,d}^{k} - x_{i,d}^{k} \right) + c_2 r_2 \left( p_{g,d}^{k} - x_{i,d}^{k} \right) \quad (9)
\]

for all \( d = 1,2,...,n \), where, \( v_{i,d} \) is the velocity of the \( d \)-th dimension of the \( i \)-th particle, and \( c_1 \) and \( c_2 \) denote the acceleration coefficients. \( p_{i,d} \) represents the local best position value of the \( i \)-th particle, \( p_{g,d} \) represents the global best position value among all the swarm of particles achieved so far. The new position of a particle is calculated using

\[
x_{i,d}^{k+1} = x_{i,d}^{k} + v_{i,d}^{k+1} \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \lo
trend of performance with the increasing $T$ in AdaBoost.RT is similar to the one in AdaBoost.IR. But in AdaBoost.R2, the value of $T$ has a larger fluctuant range. It is hard to obtain a suitable $T$. According to the relationship between RMSE and iteration $T$, we can draw a conclusion that when the value of $T$ is from 15 to 23, the AdaBoost.IR has the best and stable performance.

Next the further tests of the performance of soft sensor model are done among the soft sensor model based on BP network, AdaBoost.R2, AdaBoost.RT and AdaBoost.IR. The structure and parameters of the BP network are the same as those of sub learning machine in AdaBoost.IR, AdaBoost.RT and AdaBoost.R2. The numbers of machines $T$ is 20. The results of prediction by different soft sensor models are showed in Figs. 3, 4, 5 and 6 separately. From the performance of BP models in Fig. 3 we can find the better accuracy of prediction can be obtained for the data with smaller noise. But the performance for the data with big noise is not good. In Fig. 4, the accuracy of prediction by AdaBoost.R2 soft sensor model is greatly improved compared with the one by BP soft sensor model. Some data with big noise can be used to predict with high accuracy, but the generalization ability is not satisfied. As show in Fig. 5, the prediction of molten steel temperature by AdaBoost.RT...
has the higher accuracy than the former methods. However, the experiment need much more time to adjust the parameters. According to the process of test, the soft sensor model cannot be built when the threshold is more than 0.1. Figure 6 shows the prediction of molten steel temperature by AdaBoost.IR. The experiment proves that the method of AdaBoost.IR can increase prediction the accuracy of soft sensor model and fit the noisy industrial data. Besides, the operation of algorithm is easy and suitable for industrial production. Figure 7 shows the comparison among above three models by absolute errors. The accuracy of prediction by AdaBoost.IR is the best among the three soft models. Especially when the industrial data is with high fluctuation, the AdaBoost.IR also presents the best performance.

For testing the performance of different soft sensor regression models, the five evaluation indicators are used: RMSE, MRE, MAXE, MINE and accuracy, and they are calculated in detail in Eqs. (14)–(18).

\[
RMSE = \frac{1}{k} \sqrt{\frac{1}{k} \sum_{i=1}^{k} (f(x_i) - y_i)^2}
\]  \hspace{1cm} (14)

\[
MRE = \max_{i=1 \ldots k} \left( \frac{f(x_i) - y_i}{y_i} \right)
\]  \hspace{1cm} (15)

\[
MAXE = \max_{i=1 \ldots k} \left( f(x_i) - y_i \right)
\]  \hspace{1cm} (16)

\[
MINE = \min_{i=1 \ldots k} \left( f(x_i) - y_i \right)
\]  \hspace{1cm} (17)

\[
\text{accuracy} = \frac{N_a}{N_w} \times 100\%
\]  \hspace{1cm} (18)

Where, \(k\) is the number of samples, \(N_x\) is the number of furnaces with absolute error < 5°C, \(N_w\) is the whole testing times, \(f(x_i)\) is the prediction of soft sensor, and \(y_i\) is the real temperature. Table 3 show the performance comparison among different soft sensor models based on BP, AdaBoost.R2, AdaBoost.RT and AdaBoost.IR by testing data. According to the comparison of five evaluation indicators, we find the error of AdaBoost.IR model is limited to the minimum range. It is to say, the new AdaBoost.IR soft sensor can overcome the shortage of sensitivity to the industrial data, and improve the accuracy of prediction. It also has good ability of generalization. And the prediction results can meet the demands of industrial production.

5.2. Solving Robust Optimization of Molten Steel Temperature by HPSO-GA

After the successful temperature prediction by ensemble soft sensor, the LF refining operation can be optimized using the hybrid of particle swarm optimization and genetic algorithm (HPSO-GA). In this section, the performance of HPSO-GA for robust optimization solution will be test firstly. And then the HPSO-GA will be used to find the robust optimization solution of LF refining production process operation parameters.

Firstly, the performance of HPSO-GA for robust optimization solution is test. We devoted to describing the benchmark test problems and parameter settings of HPSO-GA which used in the experiments. Then, the efficiency of the main components in HPSO-GA compared with other evolutionary algorithms based on the benchmark test problems. In the experiments, two benchmark robustness test problems (RTP1 and RTP2) are selected. The definitions of the problems are given in Eqs. (19) and (20).

\[
f(x) = \begin{cases} 
(x + 1)^2 + 1 & -2 \leq x < 0 \\
2.6^{|x|} - 1 & 0 \leq x < 2 
\end{cases}
\]  \hspace{1cm} (19)

\[
f(x) = \begin{cases} 
-0.5^*e^{-0.5(x-0.4)^2/0.05^2} & 0 \leq x < 0.4696 \\
-0.6^*e^{-0.5(x-0.5)^2/0.02^2} & 0.4696 \leq x < 0.5304 \\
-0.5^*e^{-0.5(x-0.6)^2/0.02^2} & 0.5304 \leq x \leq 1
\end{cases}
\]  \hspace{1cm} (20)

In HPSO-GA, the population size is 100, the ranges for each dimension used in the initial population generation method is set according to their scopes. The parameters used in the cross-over and mutation operators are set as their suggested values, according to Eqs. (11) and (12). In the experiment, a total of 100 independent runs are performed for each problem to collect the statistical performance of each algorithm. The results of optimization for RTP1 and the comparison between PSO (a, c) and HSPO-GA (b, d) are given in Fig. 8. The comparison of general optimization (a, c) and robust optimization (b, d) for RTP2 are given in Fig. 9. We can find that the HPSO-GA search the optimal
solution more quickly than PSO. Meanwhile the HPSO-GA could find the result more accurate. The difference between the general optimization and robust optimization is also described. The characteristic of robust optimization is the variables with disturbance in objective function. The experiments results demonstrate that the robust optimization function can help us to search the robust optimization solution which is more stable and more effective than general optimization solution.

Secondly, the proposed HPSO-GA is used to solve the robust optimal solution of the LF refining process operation. The robust operation optimization model has been designed in section 4.1 as Eq. (8). The process of robust operation optimization of molten steel temperature based on AdaBoost.IR soft sensor is shown in Fig. 10. The molten steel temperature and corresponding operation parameters are sampled 50 furnaces during the LF refining production. The end temperatures before optimization, after general optimization and after robust optimization are shown in Fig. 11. The absolute relative error of temperature before optimization, after general optimization and after robust optimization is shown in Fig. 12.
optimization and after robust optimization are compared, and the results of comparison are showed in Fig. 11. Figure 12 shows the detail of temperature absolute relative error (ARE) before optimization and after optimization. The absolute relative error after optimization is less than the one before optimization, and absolute relative error after robust optimization is less than 0.005. Obviously, the end temperature is more stable after using the robust operation parameters which have been optimized by above robust optimization method. That is to say the optimal solution of operation parameters can guide the LF refining efficiently. The stable temperature will ensure the quality of the LF products, and fit the demands of continuous casting. Furthermore, it is benefit for enterprises to improve production efficiency, save energy and reduce cost.

6. Conclusion

For more efficient LF refining production, a robust operation optimization method is proposed to obtain the optimal operation parameters. An AdaBoost.IR algorithm is proposed to establish the accurate soft sensor model of molten steel temperature. On this basis, the robust operation optimization model is described by considering the fluctuation during the process of LF refining and the demands of production. Then a HPSO-GA is used to solving the robust optimization model. The real production data is used to test the performance of above methods. The results of experiments show the new method can ensure the stable production, improve the quality of production and save energy. In the future research the operation optimization of alloy adding will be considered further more.

Acknowledgement

This research is supported by National Natural Science Foundation of China (Grant No. 61403277 and No. 71602143).

REFERENCES

1) I. Ahmada, M. Kanob, S. Hasebea, H. Kitada and N. Murata: J. Process Contr., 24 (2014), 375.
2) H. X. Tian, Z. Z. Mao and A. N. Wang: ISIJ Int., 49 (2009), 58.
3) U. Camdali, M. Tunc and F. Dikici: Appl. Therm. Eng., 21 (2001), 643.
4) U. Camdali and M. Tunc: Can. Metall. Q., 42 (2003), 439.
5) C. Unal and T. Murat: J. Iron Steel Res. Int., 13 (2006), 18.
6) H. X. Tian and Z. Z. Mao: IEEE T-ASE, 7 (2010), 73.
7) F. He, D. F. He, A. J. Xu, H. B. Wang and N. Y. Tian: J. Iron Steel Res. Int., 21 (2014), 181.
8) K. F. Vitae, D. HEab, A. Xuab and H. B. Wang: J. Iron Steel Res. Int., 22 (2015), 68.
9) K. Shimoyama, A. Oyama and K. Fujii: Proc. 1st Int. Symp. on Evolutionary Computation, Vol. 1, IEEE, Piscataway, NJ, (2005), 950.
10) Y. C. Jin and B. Sendhoff: Proc. 2nd Int. Symp. on Evolutionary Multi-Criterion Optimization, Springer, Berlin Heidelberg, (2003), 237.
11) M. Valleroia, D. Telena, L. Cabianca, F. Manenti, J. Van Impe and F. Logist: Chem. Eng. Sci., 140 (2016), 201.
12) Q. Zhang, M. F. Morarib, I. E. Grossmann, A. Sundaramoorthy and J. M. Pinto: Comput. Chem. Eng., 86 (2016), 106.
13) Y. Freund and R. Schapire: J. Comput. Syst. Sci., 55 (1997), 119.
14) H. Drucker: Proc. 14th Int. Symp. on Machine Learning, Springer, Berlin Heidelberg, (1997), 107.
15) D. P. Solomatine and D. L. Shrestha: Proc. 1st Int. Symp. on Neural Networks, Vol. 2, Springer, Berlin Heidelberg, (2004), 1163.
16) N. Kummer and H. Najjaran: Artif. Intell. Res., 3 (2014), 64.
17) Y. J. Lou, Y. S. Zhang, R. Huang, X. Chen and Z. X. Li: IEEE T-ASE, 11 (2014), 574.
18) M. Shokriani and K. A. High: Comput. Chem. Eng., 60 (2014), 57.