Operation Optimization of Slab Reheating Process Based on Differential Evolution

Qiong XIA,^{1,2)} Xianpeng WANG^{2)*} and Lixin TANG²⁾

1) State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang, 110819 P. R. China.

2) Institute of Industrial Engineering & Logistics Optimization, Northeastern University, Shenyang, 110819 P. R. China.

(Received on April 13, 2016; accepted on July 13, 2016)

In the slab reheating process, the temperature distribution of walking beam furnace is critical for the temperature of slab, as well as the subsequent hot rolling. Besides, this process is dynamic, nonlinear, and time varying due to coupling with complex physical and chemical reactions. In this paper, we focus on an operation optimization problem of tuning furnace temperature to optimize the slab reheating process, which considers the reheating quality and production cost. For this purpose, we develop a novel operation optimization method. Specifically, first, from the view of mechanism, a heat transmit model based on the heat-exchange of slab and furnace, an energy consumption function, and especially, an oxidation loss function are established. Therefore, incorporating the mechanism models, we build an optimization model to minimize temperature deviation of slab, energy consumption, and oxidation loss to describe our problem. Then, based on the features of the problem, we propose a modified differential evolution algorithm, which includes a space contraction scheme, a new self-adaptive parameter strategy, and a new mutation operator. Finally, to evaluate the performance of our method, extensive numerical experiments are implemented by comparing it with other well-known evolutionary algorithms based on practical data. The experimental results demonstrate the effectiveness of proposed operation optimization method on solving our problem.

KEY WORDS: differential evolution; heat-exchange mechanism; operation optimization; slab reheating process.

1. Introduction

In iron and steel production, slab reheating furnace is applied to reheat slabs from slab yard or continuous casting to make their temperature distributions conform to the hot rolling requirement.

Generally speaking, in the walking beam type furnace, slabs travel through the furnace hearth along the radial direction when they are reheated. The furnace is usually divided into several zones, and the four-zone-mode is often adopted:^{1,2)} the preheating zone, the heating zone 1, the heating zone 2, and the soaking zone. An illustration of the walking beam reheating furnace of a general steel enterprise is shown in Fig. 1.

Suitable temperature distribution of a slab plays a significant role in hot rolling process. The temperature and its uniformity of slabs concern the metallic character for hot rolling. In current hot rolling production in China, slabs are mostly overheated to make sure they can be successfully rolled, which in turn results in more consumption of fuel, more oxidation loss, and more discharge of waste gas.

The energy consumption of the slab reheating process

Fig. 1. Walking beam reheating furnace.

is quite huge, which takes a great part in production cost. Consequently one significant task of the operation optimization problem for the slab reheating process (OOPSRP) is to reduce the energy consumption on the premise of getting suitable temperature distribution for the slab. On the other hand, as a slab is reheated in the furnace, the oxidation loss occurs on its surface and oxide-scale layer is generated by the oxidation action on the surface with oxygen of high temperature in furnace. The scale decreases the reheating efficiency of the furnace and hot rolling quality. So the reduction of the oxidation loss is another important task of

* Corresponding author: E-mail: wangxianpeng@ise.neu.edu.cn

DOI: <http://dx.doi.org/10.2355/isijinternational.ISIJINT-2016-211>

the operation optimization of the reheating process.

To reduce the energy consumption and oxidation loss while guaranteeing the reheating quality, the OOPSRP is investigated.

The research of optimization for slab reheating process can be traced back to 1970s by Pike Jr and Citron.³⁾ As a breakthrough of this region, the authors studied the dynamic optimization problem to control final slab temperatures, and constructed a mathematical model with the objective of minimizing fuel consumption. Up to now, as the development of modern technique, especially the computer technology, improved approaches for optimizing the slab reheating process are developed, e.g., optimization for slab reheating with prediction⁴⁻⁷⁾ and pure optimization.⁸⁻¹¹⁾ However, among these researches, most of them just optimize the temperature of slab in their problems, without considering the energy and oxidation. Only Jang *et al.*⁴⁾ added the oxidation loss in their model, and Han *et al.*⁷⁾ took into account the energy consumption in reheating process, respectively.

Because of the high complexity, we employ differential evolution (DE), one of evolutionary algorithms (EAs), to solve the OOPSRP. Compared with other methods, DE has strong robustness and global optimizing performance with universality, and it is convenient to operate with vectors. Specific to the features of OOPSRP, we modify DE with a space contraction scheme, a new self-adaptive parameter strategy, and a new mutation operator.

Three main innovation points of this paper are summarized below.

- 1) The new heat transition model established in this paper is accurate enough and much easier to apply in practice than existing mechanism model;
- 2) In the optimization objectives, according to the practical production requirements, the slab reheating quality for hot rolling, the quantity of oxidation loss in the slab surface, and the energy consumption are considered simultaneously;
- 3) To effectively solve the OOPSRP with features of multivariable and nonlinear, we propose an improved DE with a space contraction (SC) mechanism, a new composite mutation strategy, and a self-adaptive parameter setting.

The rest of the paper is organized as follows. In Section 2, the optimization strategy and mathematical models of OOPSRP are given, and a modified DE algorithm based on the features of the OOPSRP is proposed in Section 3. In Section 4, extensive numerical experiments on real and simulation data are carried out to evaluate our model and algorithm. Finally, the conclusions and future works are discussed in Section 5.

2. Mathematical Models of OOPSRP

For OOPSRP, a mechanism model of heat transfer as the relationship between the decision variables and objectives to explicitly express the relationship between the furnace temperature and slab temperature is necessary.

Due to the distinctiveness and complexity of the slab reheating furnace, it is very hard to obtain its mathematical model. Most published approaches trying to establish the temperature model of the heat transfer in furnace are based

on the system analysis of heat exchange process in furnace. Thermal radiation and transient heat conduction based Finite Volume Method (FVM) are analyzed by Steinboeck *et al.*²⁾ for the model establishment, and this model reflects the thermal field in furnace. While a User Defined Function (UDF) program is linked to FLUENT to deal with the dynamic model establishment for unsteady condition calculation considering the movement of slabs by Han *et al.*¹²⁾ Mathematical models are obtained by Jang and Lee *et al.*⁴⁾ to predict the temperature distribution of furnace and slabs. Tan *et al.*⁵⁾ proposed a model constructed based on zones, in which not only the net thermal radiation exchange between the top and bottom of furnace is considered but also the enthalpy exchange between neighbor zones is calculated. Laurinen and Rönning⁶⁾ applied a neural network method to predict the temperature of slabs. Steinboeck *et al.*,²⁾ Han *et al.*,⁷⁾ and Martín⁸⁾ predicted the temperature distribution of slabs with the finite volume method. In the above published works, the heat transfer mechanism is adequately analyzed, and the temperature distribution in the furnace is controlled or predicted, which can be a reference for our work. However, the model establishment method above for slab reheating has not been executed for the real production process, i.e., the practicability of modelling should be further enhanced.

In this paper, the mathematical models including several mechanism models and an optimization model are established. For the multi-zone feature of walking beam furnace, a set-point method is applied. The mathematical model is established for every segments of the furnace, which are obtained by equally dividing the whole furnace along the moving direction of slabs, and then the set-point values of segments of furnace are taken as the decision variables to optimize the slab reheating process. Because of the diversities of furnaces and slabs to reheat in different iron and steel corporations, there is no one accurate and fixed mathematical model of the furnace with universality for real slab reheating process. i.e., Sohlberg¹³⁾ determined several parameters of the model with grey box modeling by the actual production condition, which expanded the application range of the model. This idea is also applied in this paper.

The model in this paper is established based on the characteristics of thermal transmission analyzed by Steinboeck *et al.*²⁾ The unknown parameters of models are calculated by series of data from a real reheating process in an iron and steel corporation. This method can be adopted by other different furnaces to reheat different slabs with plenty of data from locale.

2.1. Heat Exchange Models

In this paper, we consider the travel length of slabs being reheated in furnace as D segments and suppose that slabs move at a constant speed in the furnace. After the slabs are loaded into the furnace, they go through the segments in the furnace. Each segment is considered as an independent system, in which the mathematical models of temperature for furnace and slab are established. To optimize the reheating process of slabs in the furnace, the optimal values of temperature in the segments are determined. The segmented structure of a furnace is shown in Fig. 2.

As shown in Fig. 2, the whole length of the furnace, which is also the displacement of slabs, is equally divided

Fig. 2. Structure chart of the segments of the furnace.

into D segments along the length direction, and correspondingly the reheating time of slab is also equally divided into D segments of time when the slab moves at a constant speed. Moreover, the heating state of slab at time t_j corresponds to the furnace temperature of segment j , i.e., $T_a(t_j)$, $T_s(t_j)$, and $T_c(t_j)$ are the surface, central, and average temperature of a slab, respectively, while $W(t_j)$ is the temperature of segment j .

As the variety of slabs and furnace, especially the complicated environment inside the furnace, it is difficult to establish a specific and complete mechanism model of the slab reheating process in the furnace. To overcome this issue, there are several simplifications for the slab reheating process:

- 1) In this paper, we consider one slab to represent the total slabs reheated in the furnace;
- 2) The furnace temperature we propose in this paper is the value measured by the thermocouples in the hearth of furnace;
- 3) For the convenience to establish the mechanism models of heat transfer, we assume that the slab is reheated homogeneously at the top and the bottom of it. The heat changes between the slab and the fixed and walking beams are ignored in this paper;
- 4) To explicitly express the energy consumption affected by the temperature variables, the heat quantity absorbed by slab including the radiation and convection heat is calculated as the energy consumption index for the optimization problem.

Based on the simplifications mentioned above, we adopt the basic frame of mechanism model proposed by Steinboeck *et al.*²⁾ based on the dynamic thermal analysis with the slab reheating process. The models express the mathematical relationship between the slab temperature and furnace temperature, which are expressed as

$$\dot{T}_a(t_j) = k_a \left\{ [W(t_j)]^4 - [T_s(t_j)]^4 \right\} \dots\dots\dots (1)$$

$$T_s(t_j) = T_a(t_j) + k_s \left\{ [W(t_j)]^4 - [T_s(t_j)]^4 \right\} \dots\dots\dots (2)$$

$$T_c(t_j) = T_s(t_j) - k_c \left\{ [W(t_j)]^4 - [T_s(t_j)]^4 \right\} \dots\dots\dots (3)$$

where k_a , k_s , and k_c are coefficients for $T_a(t_j)$, $T_s(t_j)$, and $T_c(t_j)$, respectively, depending on slab thickness H , heat conductivity coefficient λ , total heat absorption rate Φ , and Stefan-Boltzmann constant σ . There may be different values of slab sizes and heat environment in different furnace, thus there may be different values of these coefficients. In this paper, for the convenience of applying the mechanism models, k_a , k_s , and k_c are designated to a certain value, respectively, which are derived from the real process data by a regression method.¹⁴⁾

In this paper, several groups of data of slab reheating pro-

cess are collected. The data are obtained from a real buried-thermocouple experiment in a walking beam type furnace.

As we can observe in the above models, there is a differential Eq. (1) which is nonlinear, multi-variable, and implicit. Generally speaking, EAs do not perform well when solving a problem with differential equation. So we have to convert the differential equation to algebraic type first. However, we do not know the specific expression of $T_a(t_j)$, $W(t_j)$, and $T_s(t_j)$, which makes it difficult to solve the differential equation with traditional integral method. To overcome this difficulty, a method of polynomial interpolation is adopted to make the mathematic model be solvable with an evolutionary algorithm.

Collocation method,¹⁵⁾ as a polynomial interpolation approach, is an important implicit Runge-Kutta method, which can be used to express differential equation with polynomial interpolation of algebraic type. In this paper, collocation method is used to solve the differential Eq. (1).

The main idea of collocation method is to divide the whole time interval into D grids: $\Delta t_1, \Delta t_2, \dots, \Delta t_D$, with K collocation points in each grid, in which the number of time interval is set to D to connect with the OOPSRP. With the collocation method, the time-vary variables of the equation are approximately expressed by polynomials.

The differential Eq. (1) is solved at some time points (collocation points or interpolation points). The state variable $T_a(t_j)$ in this paper is then expressed by a Lagrange polynomial of order $K+1$ in a single finite grid, $K=3$, as is shown in Fig. 3.

In Fig. 3, $\Delta t_j = t_j - t_{j-1}$ is the length of grid j (which is also the time for slab passing one segment of the furnace), and the value of Δt_j is normalized in this segment, i.e., $\tau_k \in [0, 1]$, which is set according to the values set by Biegler,¹⁶⁾ i.e., $\tau_0=0$, $\tau_1=0.155051$, $\tau_2=0.644949$, and $\tau_3=1$.

Lagrange interpolation method is used to get the expression of $T_a(t_{jk})$ about $K+1$ interpolation points, $k=0, 1, 2, 3$. Then the value of differential variable $T_a(t_j)$ at the collocation point k in grid j is represented as follows.

$$T_a(t_{jk}) = T_a(t_{j-1}) + \Delta t_j \sum_{r=1}^K \Omega_r(\tau_k) \dot{T}_a(t_{jr}) \dots\dots\dots (4)$$

Then we incorporate the differential Eqs. (1) into (4), and the interpolation expression of differential variable $T_a(t_j)$ is rewritten as below.

$$T_a(t_{jk}) = T_a(t_{j-1}) + \Delta t_j \sum_{r=1}^3 \Omega_r(\tau_k) k_a \left\{ [W(t_{jr})]^4 - [T_s(t_{jr})]^4 \right\} \dots\dots\dots (5)$$

Where $\Omega_r(\tau_k)$ is a cubic polynomial which is determined by (6).

$$\Omega_r(\tau_k) = \int_0^1 l_r(\tau) d\tau \dots\dots\dots (6)$$

Where $l_r(\tau)$ is a Lagrange polynomial expressed by (7).

$$l_r(\tau) = \prod_{k=0, k \neq r}^K (\tau - \tau_k) / (\tau_r - \tau_k), r = 0, 1, \dots, K. \dots\dots\dots (7)$$

Here, the differential Eq. (1) is converted to an algebraic form, and the other two mechanical models, i.e., (2) and (3), can be directly applied for optimization, so the mechanism models are rewritten as (8).

Fig. 3. Polynomials interpolation for state variable in a single grid of time.

$$\begin{cases} T_a(t_{jk}) = T_a(t_{j-1}) + \Delta t_j \sum_{r=1}^3 \Omega_r(\tau_k) \cdot k_a \left\{ [W(t_{jr})]^4 - [T_s(t_{jr})]^4 \right\} \\ T_s(t_j) = T_a(t_j) + k_s \left\{ [W(t_j)]^4 - [T_s(t_j)]^4 \right\} \\ T_c(t_j) = T_s(t_j) - k_c \left\{ [W(t_j)]^4 - [T_s(t_j)]^4 \right\} \end{cases} \quad (8)$$

According to the models in (8), the mechanism models of slab reheating process in the furnace are expressed as an algebraic type which can be solved by an evolutionary algorithm.

2.2. Energy Consumption Model

Respect to the huge energy consumption in the slab reheating process, some developed methods to compute the energy consumption are obtained based on the heat balance principle in each zone of the furnace.^{17,18)} The energy efficiency is improved by Lee and Jou¹⁹⁾ through reducing the ratio of excess air and increasing the preheating temperature of the air. The furnace damper angle was adjusted by Lee and Jou²⁰⁾ to lower the pressure in the furnace hearth, retarding the rising velocity of hot gas in it, and then the heat taken by waste gas is decreased. However, it is not easy and convenient to do this. In this paper, the energy consumption is reduced through optimizing the temperature distribution of furnace.

In slab reheating process, the radiation heat and convection heat obtained by slab are the most important factors. Concretely speaking, when there is no change in the furnace construction, the energy saving in the reheating process is actually to economize the radiation and convection heat for slabs.

Respect to the slab reheated in segment j of the furnace, the radiation heat quantity absorbed by slab²¹⁾ is represented as

$$Q_j^r = C_{gwm} A_m \left[(W(t_j)/100)^4 - (T_a(t_j)/100)^4 \right], \dots (9)$$

where A_m is the reheated surface area of slab, and C_{gwm} is the radiation coefficient, which is calculated by

$$C_{gwm} = \frac{5.67 \varepsilon_g \varepsilon_m \cdot [1 + \varphi(1 - \varepsilon_g)]}{\varepsilon_g + \varphi(1 - \varepsilon_g) [\varepsilon_m + \varepsilon_g(1 - \varepsilon_m)]}, \dots (10)$$

with the gas blackness $\varepsilon_g = 0.238$ determined by the fuel of the furnace. The metal blackness $\varepsilon_m = 0.8$ is obtained from

the oxidation condition of the surface. The angle coefficient of slab, $\varphi = 0.4$, reflecting the rate of the heating area of reheated slab to the wall area of the furnace.

The convection heat for slab is obtained by

$$Q_j^c = \lambda A_m [W(t_j) - T^a(t_j)], \dots (11)$$

with the heat transfer coefficient λ , and we set $\lambda = 24(W/(m \cdot ^\circ C))$ in this paper.

Following the Eqs. (9) and (11) above, the heat absorbed by a reheated slab at each sampling point can be expressed as

$$Q = \sum_{j=1}^D (Q_j^r + Q_j^c). \dots (12)$$

The heat we calculated above is not the total energy consumption in the slab reheating process, however, this value can be a quantitative index for the energy economic through optimizing temperature of furnace and slab.

2.3. Oxidation Loss of Slabs

As oxidation loss may bring harm to the slab reheating process and the slab itself, there are several published methods to restrain the generating of scale layer for metal, many of which are to spread a coating layer on the surface of slab or other metal.²²⁻²⁴⁾ The coating layer isolates the metal from hot air to prevent the generation of scale layer indeed. The disadvantage of this method is time-consuming, high material cost, and serious reduction of the heat transfer efficiency. Additionally, it is not easy to break away the coating layer from the steel,^{22,24)} which might decrease the rolling quality and spend extra time to clean up the slab surface. Sun *et al.*²⁵⁾ proposed a nitrogen protection to reduce the growth of the oxide-scale layer on the surface of slabs, and it works very well and is easy to clean away. However, it increases much more cost for taking this measure.

An approach for operation optimization is needed to decrease the oxidation effect to slabs in the furnace without the disadvantages above. Commonly, oxide-scale layer forms and grows as the slab surface temperature rises.^{4,22,25,26)} In this paper, we reduce the oxidation loss of slabs by optimizing the reheating curve of slab through appropriately setting segment temperature of the furnace.

The quantity of oxide-scale increases as the surface temperature rises, and the process is nonlinear and dynamic. The weight growth rate of the oxide-scale in a unit area (m^2) of slab²⁷⁾ is calculated by (13).

$$\frac{dh(t_j)}{dt_j} = \frac{a_1}{h(t_j)[T_s(t_j) + 273]} \cdot \exp \left[\frac{-a_2}{T_s(t_j) + 273} \right] \dots (13)$$

Where a_1 and a_2 are oxidation dynamic constants obtained through the actual parameters of the furnace production, such as the material, the reheating requirement, the atmosphere in the furnace. In this paper, we set $a_1 = 1.315 \times 10^7 (kg/m^2)^2 \cdot K/min$ and $a_2 = 19987 K$,²⁷⁾ respectively.

When the slab arrives at the exit of the furnace, we define the reheating time for slab $t = t_j$, then the surface temperature $T_s(t_j)$ has a known value. With the basic method for solving the ordinary differential equation, the weight of oxide-scale on unit area in segment j can be calculated by the following equation.

$$h(t_j) = \sqrt{\frac{2a_1 \cdot \Delta t_j}{T_s(t_j) + 273}} \cdot \exp \frac{-a_2}{T_s(t_j) + 273} \dots\dots\dots (14)$$

Since then, the total weight of the oxidation loss of a slab in the reheating process can be calculated as below.

$$L = A_m \cdot \sum_{j=1}^D h(t_j) \dots\dots\dots (15)$$

Up to now, the mechanism models for slab reheating process are established along the longitudinal direction of furnace.

2.4. Optimization Model for OOPSRP

In this paper, the optimization objectives are the reheating quality of slab for hot rolling, the energy consumption and the oxidation burning loss in the process.

It is emphasized that, when we calculate the temperature difference of surface and center of slab, as well as the energy consumption of the slab reheating process, only the values of series discrete position points (or time points) can be obtained, which cannot represent the real objective values of OOPSRP, so we adopt the mean values of them instead of the sum of them. From another perspective, this expression for the objective variables can be used for optimization problem with different number of sampling points.

The objective function of OOPSRP is expressed in (16).

$$\min Z = \frac{\omega_1 \sum_{j=1}^D |T_s(t_j) - T_c(t_j)|}{D} \dots\dots\dots (16)$$

$$+ \omega_2 |T_s(t_f) - T_s^*(t_f)| + \omega_3 \frac{Q}{D} + \omega_4 L$$

The first item of objective function proposed above is the mean temperature difference between the surface and the central of a slab in the reheating process. The second one denotes the difference between the actual and expected surface temperatures of slab at the extracting time, in which $T_s(t_f)$ and $T_s^*(t_f)$ are respectively the optimized and expected values of surface temperature of slab. The third item of the objective function is the mean energy consumption of slab reheating process in furnace. The last item denotes the final weight of the oxide-scale layer generated in the reheating process. In addition, ω_1 , ω_2 , ω_3 , and ω_4 are the corresponding weight coefficients for the four objective function items, respectively.

Additionally, there are several constraints about temperature to ensure the security of slab and furnace in slab reheating process, which are listed as follows.

$$T_{sL} \leq T_s(t_j) \leq T_{sU} \dots\dots\dots (17)$$

$$T_{cL} \leq T_c(t_j) \leq T_{cU} \dots\dots\dots (18)$$

$$T_{aL} \leq T_a(t_j) \leq T_{aU} \dots\dots\dots (19)$$

$$W_{\min} \leq W(t_j) \leq W_{\max} \dots\dots\dots (20)$$

In the constraints above, for all $j \in [1, D]$, constraint (17) limits the surface temperature of a slab in a secure range during the reheating process. Constraints (18) and (19)

denote the central and the average temperature ranges of a reheated slab, respectively. Constraint (20) provides the temperature range of segments in furnace hearth. In general, the first three constraints are applied to protect the reheated slabs from over-heated and the fourth one is particularly important for the furnace to keep a secure condition. The upper and the lower bounds in all constraints are given in a real case.

3. An Improved DE for OOPSRP

Due to the fact that the OOPSRP problem is multimodal and nonlinear, we present a novel DE variant to solve the OOPSRP. We define the DE algorithm as O-DE, which means the DE for OOPSRP.

DE, as one of EAs, is a powerful population-based stochastic searching algorithm, which is proposed by Storn and Price in 1995²⁸⁾ to solve optimization problems.²⁹⁾ Generally speaking, DE is real number encoding, and in the OOPSRP the decision variables are also real values of temperature which are accordance with the encoding way of DE. On the other hand, the decision variables are the temperature of segments of the furnace, respectively, which are a series of parallel values, and the operation objects in DE is vectors, in which the elements are also parallel values. Additionally, the excellent capability for global searching and fast convergence speed of DE enhance the accuracy and rapidity of the solving process, so we can get better solution in a relatively shorter time with DE. Thus we get the conclusion that, DE is a suitable algorithm for solving the OOPSRP in this paper.

In this paper, there are several improvement strategies such as modified initialization, mutation strategy, parameters setting and population structure in the improved DE based on the analysis of the features of OOPSRP.

3.1. Initialization

For the purpose of convenient calculation, we directly use the element $x_{i,j}$ ($j=1, \dots, D$) of the individual $\mathbf{x}_i$ to indicate the temperature of j th segment of the furnace.

For the OOPSRP, the temperature of each segment in furnace is limited in a relatively small range to satisfy the requirement of the slab reheating process. Therefore, in O-DE, we use a space contraction (SC) mechanism to set each element $x_{i,j}$ of individual $\mathbf{x}_i$ in a corresponding given range $[L_j, U_j]$, instead of the common range $[W_{\min}, W_{\max}]$. The values of L_j and U_j are set according to the approximate experience range of actual requirement for a real reheating process. Moreover, to achieve the expected slab temperature, the furnace temperature of segments should be non-descending along the path from the entrance to the exit of the furnace, as expressed in Fig. 4. This method can shrink

Fig. 4. Setting range of segment temperature in the furnace.

the feasible space and improve the searching effectiveness of DE.

From the analysis above, we consider an individual $\mathbf{x}_i$ as a feasible solution in which each element $x_{i,j}$ is generated from the corresponding value range $[L_j, U_j]$, then the constraint expression (20) can be replaced with $L_j \leq W(t_j) \leq U_j$, and $x_{i,j} \leq x_{i,j+1}$ ($j=1, \dots, D$). Therefore, the initial population is generated by (21).

$$x_{i,j} = L_j + \text{rand}_{ij}(0,1) \cdot (U_j - L_j), \quad \dots (21)$$

$$i = 1, 2, \dots, NP, j = 1, 2, \dots, D.$$

3.2. Mutation

Mutation is the key operation of DE to generate new mutant individual to guide the searching direction. Employing better individual (with smaller fitness value) in mutation can speed up the rate of convergence, which enhances exploitation searching ability. Adopting random individual in mutation is conducive to increasing diversity of the algorithm, which improves exploration searching ability.

To balance the exploration and exploitation ability of DE, both better and random individuals should be combined in the mutation operation. Therefore, we present a new composite mutation strategy DE/current-to-rand/1(rand-to-better) (CRRB) in which the base vector is composed by the current individual and a random individual, and the difference vector is constructed by a random individual and a corresponding better individual. The strategy can be expressed as

$$\mathbf{v}_{i,g} = \mathbf{x}_{i,g} + F \cdot (\mathbf{x}_{r_1,g} - \mathbf{x}_{i,g}) + F \cdot (\mathbf{x}_{br_2,g} - \mathbf{x}_{r_2,g}), \quad \dots (22)$$

$$i \neq r_1 \neq r_2 \neq br_2,$$

where r_1 and r_2 are two randomly indexes different from i , and br_2 is the index of a random individual with better fitness value than $\mathbf{x}_{r_2,g}$, i.e., $f(\mathbf{x}_{br_2,g}) < f(\mathbf{x}_{r_2,g})$. In the new mutation strategy, the composite base vector is $\mathbf{x}_{i,g} + F \cdot (\mathbf{x}_{r_1,g} - \mathbf{x}_{i,g})$ with which we can maintain the diversity of the population by employing current and random individuals. The difference vector $\mathbf{x}_{br_2,g} - \mathbf{x}_{r_2,g}$ is applied to guide the searching direction of the individuals by adopting a random individual and a correspondingly better individual.

3.3. Crossover

After mutation operation, a new mutation individual $\mathbf{v}_{i,g}$ and its corresponding target individual $\mathbf{x}_{i,g}$ are recombined via classic crossover operation to generate trial individual $\mathbf{u}_{i,g}$. Each element of $\mathbf{u}_{i,g}$ is derived via (23).

$$u_{i,j,g} = \begin{cases} v_{i,j,g}, & \text{if } (\text{rand}_{ij}(0,1) \leq CR \text{ or } j = j_{\text{rand}}) \\ x_{i,j,g}, & \text{otherwise,} \end{cases} \quad \dots (23)$$

The evolution operation (involving mutation and crossover) of DE can be denoted as DE/x/y/z.²⁹⁾ The symbols x , y , and z indicate the base vector, difference vector, and crossover operator, respectively.

3.4. Parameters Setting

As statement in many literatures, the performance of DE is very sensitive to the setting of its control parameters. There are three parameters in DE, i.e., mutation factor F , crossover factor CR , and population size NP . F is a scaling factor that controls the step size of searching by adjusting

the size of difference vector. CR is a probability factor that controls the proportion of mutant elements combining into the trial individual. In O-DE, based on the difference between mutation and crossover, and features of the issue, we develop a self-adaptive parameter setting (SAPS) method which includes a position-based self-adaptive strategy and a fitness-based self-adaptive strategy for tuning F and CR values, respectively.

3.4.1. Position-based Self-adaptive Strategy (PBSA)

In the individual presentation, each element indicates the temperature of each segment of furnace. During the searching process, each element $x_{i,j}$ of individual $\mathbf{x}_i$ gradually tends to be close to corresponding value of the best individual $\mathbf{x}_{\text{best}}$ of current population. However, the change trends (i.e., speed and direction of move) of elements are mutually different. To take a simple instance as expressed in Fig. 5, $\mathbf{P}$ is a small population composed of four individuals $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$, and $\mathbf{x}_4$, where $\mathbf{x}_1$ is also the best solution found so far, i.e., $\mathbf{x}_{\text{best}}$. Generally speaking, in the solution space, other individuals should move along the direction towards to $\mathbf{x}_{\text{best}}$ to obtain better fitness value. For the first element 420 of the individual $\mathbf{x}_2$, the best step size is 30, however for the third element 552, to achieve 550 only needs two steps for $\mathbf{x}_2$. It can be seen from this kind of general situation that the needed step size for different elements are mutually different. In other words, promising values of mutant factor F for each element of the individual should be different with each other. Therefore, the single F_i value of classical DE is replaced by a vector $\mathbf{F}_i = \{F_{i,1}, \dots, F_{i,D}\}$.

The value of $F_{i,j}$ ($j=1, \dots, D$) in $\mathbf{F}_i$ for the j th element of $\mathbf{x}_i$ can be obtained by

$$F_{i,j} = \frac{|x_{i,j} - x_{\text{best},j}| + 1}{x_{\text{max},j} - x_{\text{min},j} + 1}, \quad i = 1, \dots, NP, j = 1, \dots, D, \dots (24)$$

where $x_{\text{max},j}$ and $x_{\text{min},j}$ are maximum and minimum values of the j th element of the corresponding individual within the population, respectively. For the first ($j=1$) element in this instance, $F_{2,1}$ should be approximately set to 0.5 ($\approx 31/61$) because corresponding values of elements $x_{i,j}$, $x_{\text{best},j}$, $x_{\text{max},j}$, and $x_{\text{min},j}$ are 420, 450, 450, and 390, respectively, among the population $\mathbf{P}$. Similarly, $F_{2,2}$ and $F_{2,3}$ are approximately set to 0.25 and 0.2, respectively. For the best individual $\mathbf{x}_{\text{best}}$ of the current generation, its mutation factor vector $\mathbf{F}_{\text{best}}$ is derived randomly from the range (0, 1).

According to the strategy, instead of a single F , a mutant factor vector $\mathbf{F}$ is generated for each individual. Moreover, to achieve the promising solution, the closer to the corresponding element of the best individual, the smaller F value is assigned, vice versa.

		j	1	2	3
$\mathbf{P}$	$\mathbf{x}_1$ ($\mathbf{x}_{\text{best}}$)		450	500	550
	$\mathbf{x}_2$		420	500	552
	$\mathbf{x}_3$		430	520	553
	$\mathbf{x}_4$		390	510	580

Fig. 5. A simple population of DE algorithm.

3.4.2. Fitness-based Self-adaptive Strategy (FBSA)

The crossover operation combines target individual $\mathbf{x}_i$ and mutant individual $\mathbf{v}_i$ to generate a trial (offspring) individual $\mathbf{u}_i$. In other words, $\mathbf{u}_i$ inherits some genes from $\mathbf{x}_i$, and obtains some new elements from $\mathbf{v}_i$ as well. For evolving the whole population, when $\mathbf{x}_i$ is a superior solution in the parent population, the offspring should acquire more elements from the target individual with a relatively smaller CR value. On the contrary, if $\mathbf{x}_i$ is an inferior one, $\mathbf{u}_i$ is supposed to get more new components from the mutant individual with a relatively larger CR value.

Base on the analysis above, a fitness-based self-adaptive strategy is adopted to set CR_i for each target individual $\mathbf{x}_i$ according to its fitness value f_i as follows.

$$CR_i = \frac{f_i - f_L + f_{L2} - f_L}{f_U - f_L + f_{L2} - f_L} \dots\dots\dots (25)$$

In formula (25), f_L , f_{L2} , and f_U are the minimum, the second minimum, and the maximum fitness values, respectively, which are obtained till the current generation. The difference between f_{L2} and f_L is applied to prevent the value of parameter being 0.

With the proposed parameter setting strategies, F and CR can be self-adaptively adjusted by PBSA and FBSA, respectively. To further balance the exploration and exploitation searching abilities, the Cauchy distribution is introduced to enhance the diversity of the parameters. Specifically, we assign the parameters as expressed as

$$F_i = randc(x_F, \gamma_F), \dots\dots\dots (26)$$

$$CR_i = randc(x_{CR}, \gamma_{CR}), \dots\dots\dots (27)$$

where the function $randc()$ returns the random values that following the Cauchy distribution. In the Cauchy distribution, central values x_F and x_{CR} are calculated by (24) and (25), respectively, and both of scaling parameters γ_F and γ_{CR} are set to 0.3. Comparing with Normal distribution, Cauchy distribution has a flatter and wider curve, *i.e.*, the latter can bring us more diverse parameter.

4. Experiment Results

To evaluate the accuracy of the mathematic models and the performance of O-DE algorithm for OOPSRP, respectively, extensive computational experiments are carried out in this section. We present the experimental results and corresponding analyses in following seven subsections. First, in Subsection 4.1, we test the accuracy of mechanism models of heat exchange proposed in Subsection 2.1 through several groups of actual data collected from the BaoSteel. In Subsection 4.2, we compare the performance of O-DE against six different DE variants and one none-DE algorithm on solving the OOPSRP, as well as the corresponding objective function values of actual furnace temperature. Finally, in Subsection 4.3, we give out some insight analyses based on the obtained experimental results.

All the computational experiments are implemented by Visual Studio C++ 2013 and SPSS 21.0 on a personal computer with the Intel Core i7-4770 3.40 GHz CPU, 4 GB memory, and Windows 8.1 64-bit operating system.

4.1. Accuracy of Heat Exchange Model

To evaluate the accuracy of our heat exchange model proposed in Subsection 2.1, we firstly collected three groups of actual data, which include the temperature values of furnace and slab, from three different reheating furnaces of the BaoSteel Corporation. For the distinctions between the three furnaces, they are not for the same hot rolling production line. The corresponding condition values of operating environment are described as follows:

4.1.1. Structure Parameters of Furnace

The efficient length of the furnace is 45 m. In addition, the reheating time of slab is 180 minutes, 194 minutes and 221 minutes in the furnace 1#, furnace 2# and furnace 3#, respectively.

4.1.2. Size of Slab

Because of the mathematical model depends on real condition parameters, we obtain different models through different reheated slabs, and a certain type of slab to reheat is enough for these experiments. Therefore, the testing slab sizes we applied are 9 000 mm (length) $\times$ 1 200 mm (width) $\times$ 210 mm (thickness).

4.1.3. Temperature Requirements of the Reheating Process

The expected extracting temperature of slab is set to be 1 200°C, and the slabs initiatory temperature is set to 20°C, while the difference the final temperature with the idea value is limited to 30°C. The iron type of slabs is HPHC, which is a type of European standard steel mainly used in large steel truss structure.

Under the above conditions, the temperature of furnace (W), slab center (T_c), and slab surface (T_s) are sampled and recorded once every 20 seconds through a “black box” with temperature recorder. To briefly express the temperature change processes of furnace and slabs, and match the segments variables of our operation optimization method, we uniformly extract 12 values for average temperature during the whole reheating process. Therefore, the real temperature curves of three furnaces and corresponding testing slabs can be illustrated in **Fig. 6**, where T_a and T_d indicate the average temperature of testing slab and the temperature difference between surface and center ($=T_s - T_c$), respectively.

Putting these values into the mechanism model formula (8), we can compute the temperature coefficients k_a , k_s , and k_c of each furnace, and list them in **Table 1**.

To clearly analyze and compare the performance indicators of each group, we calculate temperature difference index (TD1) between surface and center, final temperature difference index (TD2), energy consumption (EC), and oxidation loss (OL) of objective function (16), respectively, and list them in **Table 2**.

Through the actual index values in Table 2, we can observe that, there exist great differences between the orders of magnitude of the four index values. To fully reflect the contribution made by each index value to the objective function, we normalize them by setting the weight coefficients as expressed in **Table 3**.

Therefore, combining with the given coefficients, we can obtain the objective function values (FV) of three groups of actual temperature data.

(a)

(b)

(c)

Fig. 6. The real temperature curves of the testing slab in furnace 1# (a), 2# (b), and 3# (c).

Table 1. The temperature coefficients in the model.

Furnace Index	k_a	k_s	k_c
1#	4.63E-10	1.13E-11	3.40E-11
2#	8.53E-10	6.30E-12	6.49E-11
3#	-2.07E-08	5.07E-11	1.11E-11

In Table 2, based on the heat transfer mechanism model (8), the slab temperature under actual furnace temperature data can be calculated. Three additional groups of real temperature data for furnaces 1#, 2#, and 3# are collected to verify our model. The slab reheating curves of corresponding temperature difference (T_d') between actual average temperature (T_a) and calculated value (T_a') by the model are illustrated in Fig. 7.

From the curves, it can be seen that the difference between the actual and the calculated temperature values

Table 2. The performance indicators of actual data.

Furnace Index	TD1 (°C)	TD2 (°C)	EC (kJ)	OL (g)	FV
1#	2.94E+01	4.89E+01	6.94E+05	6.81E+01	1.54E+01
2#	3.60E+01	7.71E+01	4.68E+05	5.59E+01	1.65E+01
3#	2.08E+01	3.13E+01	4.41E+05	9.25E+01	1.05E+01

Table 3. The setting of weight coefficients.

Weight Coefficient	Value
ω_1	1.0E-01
ω_2	1.0E-01
ω_3	1.0E-05
ω_4	1.0E-02

(a)

(b)

(c)

Fig. 7. The curves of average temperature obtained by the model and temperature difference from the actual values of the testing slab on furnace 1# (a), 2# (b), and 3# (c).

are limited within a small range (-15°C , 20°C) during the whole reheating process. This range satisfies the given accuracy requirement in practical slab reheating process. In other words, the proposed heat transfer mechanism models are accurate enough to apply for the optimization model to evaluate the reasonability of the furnace temperature.

4.2. Comparing with Other Algorithms

In this subsection, to demonstrate the outstanding performance of O-DE to solve the OOPSRP problem, eight different methods are selected as the competitors, which include five well known classic DE variants, one improved DE variant, one none-DE algorithm, and manual method as follows.

- 1) DE/rand/1/bin with $F=0.5$, $CR=0.5$;
- 2) DE/best/1/bin with $F=0.5$, $CR=0.5$;
- 3) DE/current-to-best/1/bin with $F=0.5$, $CR=0.5$;
- 4) DE/rand/2/bin with $F=0.5$, $CR=0.5$;
- 5) DE/best/2/bin with $F=0.5$, $CR=0.5$;
- 6) IDE with the suggested parameters setting;³⁰⁾
- 7) CLPSO with the best parameters used in;³¹⁾
- 8) Manual method, *i.e.*, the actual temperature data.

The testing slab and three furnaces used for collecting temperature data in previous subsection are still employed to construct the test environment, and the corresponding condition parameters are set as before. To protect the furnace stack, the temperature of the furnace is limited in the range $[300^{\circ}\text{C}$, $1300^{\circ}\text{C}]$ which also includes the searching range of each dimensional element of individuals in algorithm. Besides, to prevent slabs being over burnt or under burnt, the range $[35^{\circ}\text{C}$, $1250^{\circ}\text{C}]$ is taken for the average, surface, and central temperature boundary of slab, simultaneously. In addition, the values of temperature coefficients and weight coefficients of model are set as in Tables 1 and 3, respectively.

For all the testing algorithms, the population size $NP=50$, and 30 independent runs are conducted with $g_{\max}=200$, *i.e.*, the number of objective function evaluations (FES) equals 10 000 ($NP \times g_{\max}$). We directly employ the objective function value $f(\mathbf{x}_{\text{best}})$ to evaluate the performance of each algorithm, where $\mathbf{x}_{\text{best}}$ is the best solution found so far until the searching stop. In addition, the mean value (denoted by Mean) of the 30 runs and the corresponding standard deviation (denoted by St.d.) are calculated and listed in the experimental results **Table 4**.

Furthermore, to get a statistic conclusion of the numerical results, we apply the two-sided Wilcoxon rank sum test³²⁾ with 0.05 significant level to analyze the significance of the performance difference between O-DE and each competitor algorithm. In the results Table, we also mark the three kinds of statistics significant cases (denoted by Stat.) with “+”, “=”, and “-” to indicate that O-DE performs significant better than, similar to, and worse than the competitor, respectively.

From the results, we can conclude that O-DE performs significantly better than other competitors for all test instances.

4.3. Insight Analyses

In this subsection, based on the experimental results above, a series of insight analyses are given on OOPSRP and O-DE.

Table 4. Numerical results of comparison between O-DE and other algorithms.

	Furnace1#			Furnace2#			Furnace3#		
	Mean	St.d.	Stat.	Mean	St.d.	Stat.	Mean	St.d.	Stat.
O-DE	9.17	0.15	/	10.09	0.17	/	6.19	0.27	/
DE/rand/1	10.67	0.42	+	11.24	0.43	+	6.83	0.32	+
DE/best/1	11.13	1.08	+	15.32	0.36	+	10.15	0.16	+
DE/c-to-b/1	9.76	0.35	+	11.11	0.91	+	6.61	0.95	+
DE/rand/2	10.93	0.47	+	11.75	0.30	+	7.02	0.42	+
DE/best/2	11.44	0.25	+	11.97	0.18	+	7.25	0.08	+
IDE	9.72	0.27	+	11.05	0.17	+	6.49	0.18	+
CLPSO	10.34	0.74	+	11.43	0.64	+	7.08	0.59	+
MANUAL	15.44	/	/	16.54	/	/	10.53	/	/

Fig. 8. The optimized temperature curves of the testing slab in furnace 1# (a), 2# (b), and 3# (c).

4.3.1. On OOPSRP

The reheating curves of testing slab within practical furnace temperature have been depicted in Subsection 4.1, here we draw the temperature curves of the slab in Fig. 8, according to the optimized furnace temperature obtained by O-DE.

In Fig. 8, comparing with the un-optimized heating curves in Fig. 6, we can observe from the optimized curves that the temperature of three furnaces rise smoothly during the first half of heating process (including preheating zone and reheating zone 1), and keeps in a relatively stable range at the latter half of heating process (including reheating zone 2 and soaking zone). Besides, under this kind of temperature distribution, the difference between slab center and surface temperatures becomes smaller.

Therefore, we can obtain a conclusion that the optimized furnace temperature is more suitable for the heating process of slabs.

4.3.2. On O-DE

According to the No Free Lunch Theorem,³³⁾ there is no algorithm that can perfectly solve all kinds of optimization problems, *i.e.*, the algorithm is problem dependent. In other words, for a specific issue, the improved strategies derived from the both problem feature and algorithm mechanism have to be combined to construct an effective solver method.

O-DE is developed to solve the OOPSRP based on the features of the problem. Firstly, according to the different temperature range of different furnace segments, SC mechanism shrinks the feasible region and thus speeds up the convergence rate. Secondly, based on the above strategy, adaptive parameter setting assigns control parameters for furnace temperature of each segment, which can improve the searching effectiveness. Finally, the CRRB mutation operator enhances the diversity of algorithm population and guides the searching direction by introducing random and better individuals, respectively.

In conclusion, the problem-dependent improvement strategies make our method perform better than the other algorithms on solving the OOPSRP.

5. Conclusions

In this paper, we focus on the OOPSRP problem, *i.e.*, to optimize the extracting temperature of slab by tuning the furnace temperature. For this purpose, first, we modified and presented a new heat transfer model between furnace and slab. Then, an optimization model of OOPSRP is established. Subsequently, to solve the OOPSRP problem efficiently, an improved DE based operation optimization method is proposed, which contains three new strategies derived according to the features of the problems. Finally, based on actual data, extensive computational experiments are carried out to test the proposed model and approach. The experimental results show that our mechanism model can express the heat transition relationship between the temperatures of furnace and slabs.

In terms of a continuation of the research in this paper, we are investigating the data driven method based general

method to establish the heating transform model. We are also developing an application software platform based on our model and algorithm and identifying opportunities to solve practical issues by collaborating with the enterprise.

Acknowledgements

This research is partly supported by the National 863 High-Tech Research and Development Program of China (2013AA040704), the Innovative Research Groups of the National Natural Science Foundation of China (71321001), and the National Natural Science Foundation of China (61374203, 61573086).

REFERENCES

- 1) M. Cui, X. W. Gao and H. G. Chen: *Int. J. Therm. Sci.*, **50** (2011), 898.
- 2) A. Steinboeck, D. Wild, T. Kiefer and A. Kugi: *Int. J. Heat Mass Transf.*, **53** (2010), 5933.
- 3) H. E. Pike, Jr. and S. J. Citron: *Automatic*, **6** (1970), No. 1, 41.
- 4) J. H. Jang, D. E. Lee, M. Y. Kim and H. G. Kim: *Int. J. Heat Mass Transf.*, **53** (2010), 4326.
- 5) C. K. Tan, J. Jenkins, J. Ward, J. Broughton and A. Heeley: *Appl. Therm. Eng.*, **50** (2013), 1111.
- 6) P. Laurinen and J. Rönning: *J. Mater. Process. Technol.*, **168** (2005), 423.
- 7) S. H. Han, S. W. Baek and M. Y. Kim: *Int. J. Heat Mass Transf.*, **52** (2009), 1005.
- 8) E. Martin, M. Meis, C. Mourenza, D. Rivas and F. Varas: *Appl. Therm. Eng.*, **47** (2012), 41.
- 9) J. D. Li, Y. Li, D. D. Zhao, T. L. Fu, Z. D. Wang and G. D. Wang: *J. Iron. Steel Res. Int.*, **19** (2012), No. 12, 1.
- 10) A. Steinboeck, K. Graichen and A. Kugi: *IEEE Trans. Control Syst. Tech.*, **19** (2011), No. 6, 1444.
- 11) A. Steinboeck, K. Graichen, D. Wild, T. Kiefer and A. Kugi: *J. Process. Control*, **21** (2011), 279.
- 12) S. H. Han, D. Chang and C. Y. Kim: *Int. J. Heat Mass Transf.*, **53** (2010), 3855.
- 13) B. Sohlberg: *J. Process Control*, **13** (2003), 225.
- 14) C. F. Lee, J. C. Lee and A. C. Lee: *Statistics for Business and Financial Economics*, World Scientific, Hong Kong, (1999), 652.
- 15) L. T. Biegler: *Chem. Eng. Process*, **46** (2007), 1043.
- 16) L. T. Biegler: *Nonlinear Programming*, MOS-SIAM Press, Philadelphia, (2010), 287.
- 17) S. H. Han, D. Chang and C. Huh: *Energy*, **36** (2011), 1265.
- 18) H. J. Karimi and M. H. Saidi: *J. Iron. Steel Res. Int.*, **17** (2010), No. 4, 12.
- 19) C. L. Lee and C. J. G. Jou: *Fuel Process. Technol.*, **92** (2011), 2335.
- 20) C. L. Lee and C. J. G. Jou: *Int. J. Hydrogen. Energy*, **38** (2013), 2504.
- 21) H. Baehr and K. Stephan: *Heat and Mass Transfer*, 2nd ed. Springer, Heidelberg, (2006), 10.
- 22) X. Zhou, S. F. Ye, H. W. Xu, P. Liu, X. J. Wang and L. Q. Wei: *Surf. Coat. Tech.*, **206** (2012), 3619.
- 23) P. Liu, L. Q. Wei, S. F. Ye, H. W. Xu and Y. F. Chen: *Surf. Coat. Technol.*, **205** (2011), 3582.
- 24) X. J. Wang, L. Q. Wei, X. Zhou, X. M. Zhang, S. F. Ye and Y. F. Chen: *Appl. Surf. Sci.*, **258** (2012), 4977.
- 25) W. H. Sun, A. K. Tieu, Z. Y. Jiang and C. Lu: *J. Mater. Process. Technol.*, **155–156** (2004), 1307.
- 26) D. B. Lee and J. W. Choi: *Oxid. Met.*, **64** (2005), 319.
- 27) B. A. Markovsky and I. I. Lavrentivchuk: *Furnace Control Algorithm*, 1st ed., Beijing, MIP Press, China, (1985), 1.
- 28) R. Storn and K. V. Price: *Tech. Rep. TR-95-012*, Int. Computer Science Institute, Berkeley, CA, (1995).
- 29) K. V. Price, R. Storn and J. Lampinen: *Differential Evolution- A Practical Approach to Global Optimization*, Springer, Berlin, (2005), 37.
- 30) L. X. Tang, Y. Dong and J. Y. Liu: *IEEE Trans. Evol. Comput.*, **19** (2015), No. 4, 560.
- 31) J. J. Liang, A. K. Qin, P. N. Suganthan and S. Baskar: *IEEE Trans. Evol. Comput.*, **10** (2006), No. 3, 281.
- 32) H. B. Mann and D. R. Whitney: *Ann. Math. Stat.*, **18** (1947), No. 1, 50.
- 33) D. H. Wolpert and W. G. Macready: *IEEE Trans. Evol. Comput.*, **1** (1997), No. 1, 67.