IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 21, 2020, accepted April 18, 2020, date of publication April 28, 2020, date of current version May 13, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2990904

Multi-Objective Optimization of Rolling Schedule for Five-Stand Tandem Cold Mill

YUNLONG WANG[1], JINKUAN WANG[1], (Member, IEEE), CHUNHUI YIN[2], AND QIANG ZHAO[3]

[1]College of Information Science and Engineering, Northeastern University, Shenyang 110819, China
[2]School of Computer and Communication Engineering, Northeastern University at Qinhuangdao, Qinhuangdao 066004, China
[3]School of Control Engineering, Northeastern University at Qinhuangdao, Qinhuangdao 066004, China

Corresponding author: Jinkuan Wang (wjk@neuq.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFB0304100, in part by the National Natural Science Foundation of China under Grant U1908213, in part by the Fundamental Research Funds for the Central Universities under Grant N182303037, and in part by the Foundation of Northeastern University at Qinhuangdao under Grant XNB201803.

ABSTRACT The optimization of rolling schedule is the main content of tandem cold rolling which will affect the quality of products directly. A rolling schedule with the objectives of minimum energy consumption, relative power margin and slippage preventing is established. First, in order to make the rolling schedule more accurate in the calculation process, a mathematical model combines with deep neural network is proposed to calculate the rolling force. Second, a multi-objective particle swarm optimizer with dynamic opposition-based learning is proposed to optimize the rolling schedule. It has a new particle learning strategy to update the moving position of particles. Moreover, opposition-based learning is proposed to make the particles jump out of local optima. Finally, the experiments are carried out based on the field data. Simulation results demonstrate that the accuracy of the rolling force is greatly improved. The proposed algorithm has a promising performance on both diversity and convergence. At the same time, the optimized rolling schedule can well balance the rolling power and prevent slipping between five stands comparing with the original rolling schedule.

INDEX TERMS Multi-objective optimization, tandem cold rolling, deep neural network, rolling force, rolling schedule.

I. INTRODUCTION

With the development of economy, the demand for cold-rolled strip in the steel market is growing rapidly. The requirement of consumers for the quality of products has increased. Steel strip manufacturing is a complex and large industrial process. It covers multiple levels of control and multiple procedures [1]. The cold rolling process equipment, measuring instruments and their installation positions of a five-stand UCM tandem cold rolling mill in a factory are shown in Figure 1. The setting of rolling schedule affects the quality of products and energy consumption. Rolling schedule is an important aspect of the cold rolling process. A reasonable setting of rolling schedule can fully utilize the equipment capabilities, which not only improves the production and quality of steel but also reduces the consumption of rolling energy. Therefore, the optimization of rolling schedule is a typical multi-objective optimization problem.

The associate editor coordinating the review of this manuscript and approving it for publication was Olga Fink.

There are a lot of references that transforms the rolling schedule into a single-objective optimization problem by weighted method. Qi *et al.* integrated relatively equal load and good strip crown and flatness into one objective function in [2]. Chen *et al.* transformed the multi-objective optimization problem into a single-objective optimization problem in [3]. The optimization of rolling schedule needs to meet the constraints such as rolling force, rolling torque and rolling power. By introducing a penalty term into the objective function, the rolling schedule was transformed into an unconstrained problem. The function was solved by the Nelder-Mead simplex method. Compared with the rolling schedule optimized by empirical formula, the proposed method can well balance the rolling force and rolling power simultaneously. Poursina *et al.* integrated power distribution cost, tension cost function, perfect shape condition into one cost function in [4]. They used the genetic algorithm to optimize the cost function and achieved a good result. The case-based reasoning-Tabu search hybrid algorithm was proposed to optimize the rolling schedule in tandem cold

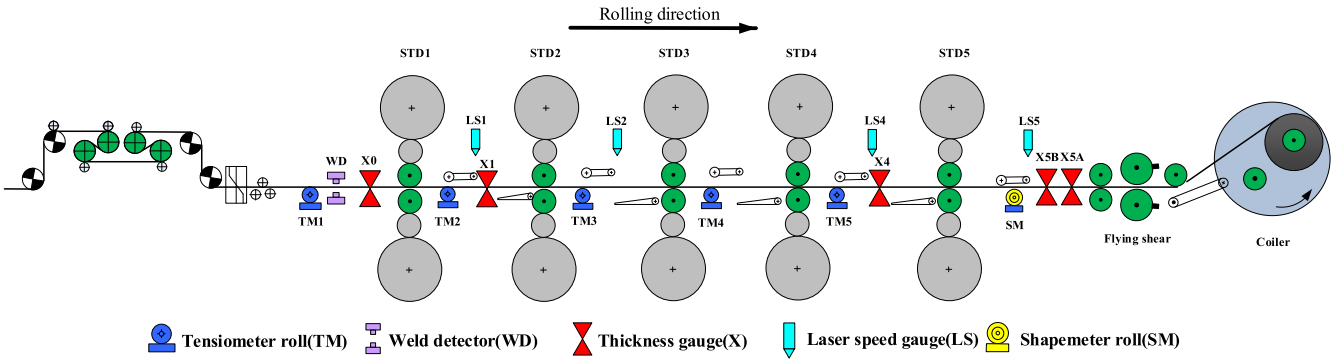


FIGURE 1. Equipment layout of a five-stand tandem cold mill.

rolling in [5]. The improved iterative method reduced the number of iterations by about half compared with the ordinary tabu search algorithm. However, in practice, because the objective functions of multi-objective optimization contradict each other and the number of optimization targets is large, it is difficult to determine the weight coefficients. In recent years, the multi-objective optimization algorithm provides a solution for solving the problem of rolling schedule. The multi-objective fuzzy theory and method was proposed to optimize the rolling schedule in [6]. Li *et al.* proposed a MODE/D algorithm to optimize the rolling schedule in [7]. The results showed that it can obtain a good Pareto-optimal front which made a good trade-off among the three objective functions. A differential evolution algorithm based on the evolutionary direction was proposed to solve the multi-objective optimization model of the rolling schedule. The results showed that the profile was improved and rolling energy consumption was reduced compared with the actual rolling schedule [8]. Although there have been many multi-objective optimization algorithms proposed to calculate the rolling schedule, the diversity and convergence of the solution set is not good enough. As a result, there will be problems such as the quality of products decline and high energy consumption.

The rest of this paper is organized as follows. In section 2, a hybrid rolling force model of deep neural network model combines with a mathematical model is introduced. In section 3, the mathematical models of the rolling schedule are introduced. A multi-objective particle swarm optimizer with dynamic opposition-based learning which is named MOPSO-DOL is proposed to optimize the rolling schedule. In section 4, the proposed model is used to predict the rolling force and the comparative experiment is carried out. The proposed MOPSO-DOL is tested on benchmark problems with other multi-objective optimization algorithms and it is used to optimize the rolling schedule. The optimal solutions are chosen by the weighted-sum approach as the set values of the rolling schedule. Finally, section 5 presents the conclusion.

II. THE ROLLING FORCE MODEL

The rolling force is a very important parameter in the cold rolling process. The accuracy of the rolling force model directly affects the accuracy of the output thickness. The presetting of the rolling force controls the setting of the rolling schedule.

The traditional mathematical model is a model based on the internal mechanism of objects and production processes. A new approach was proposed to predict the rolling force of cold rolled sheet based on the plastic mechanics in [15]. Most of the errors fallen within the range of $\pm 10\%$. However, due to the complexity of the working environment and the coupling relationship between variables, the traditional mathematical model requires many assumptions in practice. Thus, it can't reflect the influence of disturbance factors on the rolling force.

In order to predict the rolling force more accurately, intelligence-based models are widely introduced into the field of rolling. AziGuLi *et al.* used the extreme learning machine and self-learning model to predict the rolling force in [16]. The artificial neural network model was used to predict the rolling force and acquired a high precision in [18]. Mahmoodkhani *et al.* developed an online rolling force prediction tool by combining the finite element model with the artificial neural network in [19]. The error of rolling force was reduced to 10%. But these models are all shallow machine learning models whose express ability to the complex function is restricted.

The past several years have seen increasingly rapid advances in the field of deep learning, deep neural network has a good nonlinear fitting ability, which can reflect the influence of disturbance factors on rolling force accurately. Wei *et al.* tried to apply deep learning to the prediction of rolling force in [20]. An MLP rolling force prediction model based on deep learning method was proposed. Simulation results showed that the model could reduce the relative error between the predicted results and the measured data to less than 3%. Many models have been established to improve the prediction accuracy of the rolling force. These models either consider only data-driven models or only mathematical models. A method combining artificial neural network and physical model was proposed to calculate the rolling force in [17], where deformation resistance and friction coefficient were calculated by physical models. The Bland-Ford-Ellis rolling force model was used to calculate the rolling force.

In this paper, we propose a method combining the mathematical model and the deep neural network to calculate the rolling force.

A. TRADITIONAL MATHEMATICAL MODEL

In this paper, the SIMS mathematical model is applied to calculate the rolling force. The SIMS model is expressed as:

$$F = Bl'_c Q_p K K_T \tag{1}$$

where F is the rolling force; B is the strip width, mm; l'_c is the roll contact length, mm; K is the deformation resistance of metals, $K = 1.15\sigma$; K_T is the influence coefficient of front and back tensile stress on the rolling force.

The calculation formula of the contact arc length after flattening is as follows.

$$l'_c = \sqrt{R'(h_0 - h_1)} \tag{2}$$

where R' is the flattened roll radius, h_0 and h_1 are the entry thickness and exit thickness, mm.

The flattened roll radius calculated by the Hitchcock formula is:

$$R' = R\left(1 + 2.11 \times 10^{-5} \frac{F}{B(h_0 - h_1)}\right) \tag{3}$$

where R is roll radius, mm. R' can be calculated by an iterative method. Then Hill's simplified external friction stress state coefficient formula is:

$$Q_p = 1.08 + 1.79\mu\varepsilon\sqrt{1-\varepsilon}\sqrt{R'/h_1} - 1.02\varepsilon \tag{4}$$

where ε is the relative reduction rate, μ is the coefficient of friction.

$$K_T = 1 - \frac{(\alpha T_b - (1-\alpha)T_f)}{K} \tag{5}$$

where T_b and T_f are the back tension and forward tension, MPa.

For a cold-rolled strip, the magnitude of the deformation resistance depends on the chemical composition and cumulative deformation of metal materials. The deformation rate and deformation temperature have less influence on the deformation resistance. Therefore, the selected deformation resistance model is shown in (6).

$$\sigma = \frac{2}{\sqrt{3}}(A + B \cdot \varepsilon_\Sigma)(1 - C \cdot e^{-D \cdot \varepsilon_\Sigma}) \tag{6}$$

where ε_Σ is cumulative deformation resistance, $\varepsilon_\Sigma = 2/\sqrt{3} \times \ln(H_0 - h)$; H_0 is the thickness of raw material, mm; h is the target thickness, mm; A, B, C, D are the coefficients of deformation resistance model associated with steel grades.

The traditional SIMS is the most commonly used rolling force model in the field. Using the SIMS model only will cause a large deviation. The coefficients in the model are constant and the model makes many assumptions that it doesn't consider the impact of disturbance factors on the rolling force. The rolling force and the flattened roll radius need to be calculated by the iterative method, which also causes the deviation of the rolling force. Therefore, it is not appropriate to use the SIMS model only to calculate the rolling force. It needs to be corrected to adapt to the field application.

B. AGGREGATION MODEL OF DNN AND SIMS

In this paper, the method of training deep neural network in [20] is used to predict the rolling force. Roller radius, entrance thickness, initial thickness, objective thickness, forward tension, back tension, deformation resistance are chosen as the inputs of the deep neural network according to the SIMS model.

Three kinds of SIMS model combines with deep neural networks are proposed. The first combination method is marked as DNN combines with SIMS I. The mathematical model is used to calculate the main value of the rolling force, and the deep neural network is used to predict the deviation of the rolling force. The rolling force deviation is the actual rolling force minus the rolling force calculated by the mathematical model. The second combination method is marked as DNN combines with SIMS II. The deep neural network is used to predict the deviation of the rolling force. The rolling force deviation is the ratio of rolling force calculated by the mathematical model divided by the actual rolling force. The final rolling force is obtained by multiplying the calculated value of the mathematical model and the predicted ratio. The third combination method is marked as DNN combines with SIMS III. The rolling force calculated by the mathematical model is regarded as one of the inputs of the deep neural network. The third combination method is shown in Figure 2.

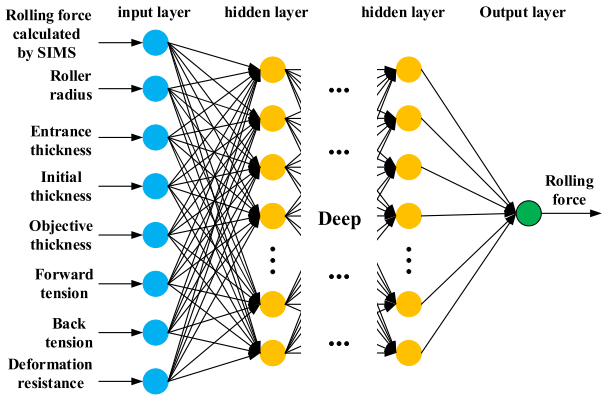


FIGURE 2. DNN combines with the SIMS III rolling force model.

III. THE ROLLING SCHEDULE MODEL

In this part, the objective functions, constraints and decision variables are established firstly. Then a multi-objective particle swarm optimizer with dynamic opposition-based learning is proposed to optimize the rolling schedule.

A. THE OBJECTIVE FUNCTIONS OF THE ROLLING SCHEDULE

The rolling schedule for five-stand tandem cold mill was established in [22]. The rolling power model is expressed as:

$$P = M\omega \tag{7}$$

where P is the rolling power, KW; M is the rolling torque, K$N \cdot m$; ω is the rated speed of motor, rad/s; The model of rolling torque is as follow:

$$M = 2Fl'_c\psi \tag{8}$$

where ψ is the coefficient of the rolling force arm, in this paper, $\psi = 0.5$.

For minimizing the energy consumption generated during the rolling process, the minimum energy consumption objective function is established. The objective function can be defined as in (9).

$$f_1 = \frac{\sum_{i=1}^{5} \frac{P_i}{P_{ri}}}{5} \tag{9}$$

where P_i is the actual power of stand i; P_{ri} is the rated power of stand i.

In order to increase productivity and fully exploit equipment capabilities, the relative power margin objective function is established which is expressed as follows:

$$f_2 = \sum_{i=1}^{5}\sum_{j>i}^{5}\left|\frac{P_{ri} - P_i}{P_{ri}} - \frac{P_{rj} - P_j}{P_{rj}}\right| \tag{10}$$

For the purpose of preventing the occurrence of slipping during the process of rolling, a function to prevent slipping is established. Slipping refers to the phenomenon of relative movement between rolls and strips which easily causes scratches on the surface of strips. The objective function for preventing slipping is shown in (11).

$$f_3 = \beta\sqrt{\sum_{i=1}^{5}\left(\Psi_i - \frac{1}{5}\sum_{i=1}^{5}\Psi_i\right)^2} + (1-\beta)\frac{1}{5}\sum_{i=1}^{5}\Psi_i \tag{11}$$

where β is the weighted coefficient, and generally β is 0.4. Ψ_i is the slip factor of stand i. And Ψ_i is shown in (12).

$$\Psi_i = \frac{1}{4\mu_i}\left|\sqrt{\frac{h_{0i} - h_{1i}}{R'_i}} + \frac{T_{bi} - T_{fi}}{F_i}\right| \tag{12}$$

The rolling mill can effectively prevent slipping when $0 < \Psi_i < 0.5$. Obviously, the smaller the slip factor, the smaller the probability of slippage occurs.

B. THE CONSTRAINTS OF THE OBJECTIVE FUNCTIONS

When optimizing the rolling schedule, the selection of the rolling schedule must meet the constraints. According to the reasons for the constraints, it can be divided into equipment constraints and process constraints.

The equipment constraints are as shown below.

$$\begin{aligned} 0 &\le F_i \le F_{i\max} \\ 0 &\le P_i \le P_{i\max} \\ 0 &\le M_i \le M_{i\max} \end{aligned} \tag{13}$$

where F_i, P_i, M_i are the rolling force, rolling power, and rolling torque of ith stand respectively; $F_{i\max}$, $P_{i\max}$, $M_{i\max}$ are the maximum rolling force, rated power of the motor and maximum rolling torque of the ith stand respectively.

The process constraints are as follows.

$$\begin{aligned} \varepsilon_{i\min} &\le \varepsilon_i \le \varepsilon_{i\max} \\ T_{i\min} &\le T_i \le T_{i\max} \end{aligned} \tag{14}$$

where ε_i and T_i are the reduction rate and tension of each stand; $\varepsilon_{i\min}, \varepsilon_{i\max}$ and $T_{i\min}, T_{i\max}$ are the minimum and maximum of the reduction rate and tension of ith stand.

C. THE SELECTION OF DECISION VARIABLES

The rolling schedule for tandem cold mill involves multiple rolling parameters, and the reduction rate and tension of each stand are two important parameters. After the reduction rate and tension are determined, the rolling force, rolling torque, rolling power can be calculated according to the relevant models.

For a five-stand tandem cold mill, the entry thickness of the first stand and the exit thickness of the fifth stand are known; The entry tension of the first stand and the exit tension of the fifth stand are given according to the process conditions. Therefore, a total of eight variables that are thickness and tension between each stand are selected [3].

$$X = (h_1, h_2, h_3, h_4, T_1, T_2, T_3, T_4)^T \tag{15}$$

where X is the optimization vector, h_1, h_2, h_3, h_4 are thickness between each stand; T_1, T_2, T_3, T_4 are tension between each stand.

D. MULTI-OBJECTIVE OPTIMIZATION MODEL

The diversity and convergence of solution set of traditional multi-objective optimization algorithms have poor performance when optimizing the rolling schedule. The competitive mechanism based multi-objective particle swarm optimizer (CMOPSO) [9] has fast calculation speed and high convergence accuracy, but the diversity of the solution set is poor, and it is easy to fall into local optima. To address the above problems, a new algorithm called MOPSO-DOL is proposed. The algorithm has three main components: the random particle learning strategy, the dynamic opposition-based learning strategy, and the environmental selection.

The random particle learning strategy is composed of elite particle selection and random particle learning. The elite particles are selected from the non-dominated sorting and crowding distance based sorting. In CMOPSO, γ elite particles are selected in order from the ranking results as the elite solution set L. After the elite solution set L is determined, the algorithm performs a pairwise competition strategy.

Suppose the particles a and b are two randomly selected particles from the elite solution, the particle p is the current particle to be updated, and θ_1 and θ_2 are the angles between the particle p and the elite particles a and b. The angles θ_1 and θ_2 are calculated separately. Particle with a smaller angle with the particle p will be used as the winning particle. The particle a is marked as a winning particle if $\theta_1 < \theta_2$ for guiding the update of the particle p.

When the winning particle is selected, the particle p updates its position and velocity by learning from the winning particle. Assume that the position of the particle p is P_i, the velocity is V_i, and the position of the winning particle is P_w. Then the velocity update formula and position update formula of particle p are:

$$V_i' = R_1 V_1 + R_2(P_w - P_i) \tag{16}$$

$$P_i' = P_i + V_i' \tag{17}$$

where R_1 and R_2 are randomly generated vectors in the interval [0,1]. After that, all particles perform polynomial variations to enhance the ability to search for optimal regions.

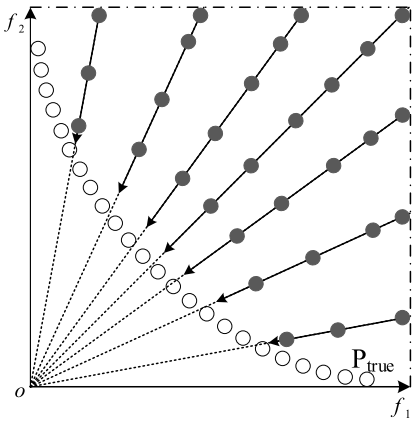


FIGURE 3. The flying direction of particles in an ideal situation based on competition mechanism.

It's obvious that the competition-based method can make the particle flying faster to its convergence direction. When the ideal situation occurs, the flying direction of each particle is shown in Figure 3. It can be seen from the picture that all of the particles are flying towards their nearest convergence direction. It is assumed that the feasible particle solutions are evenly distributed in space. Thus, the particles gradually decrease from the middle to the sides. The number of particles that eventually converge on the real Pareto front will gradually decrease from the middle to the sides, which will undoubtedly reduce the distribution and convergence of the final solutions.

In order to make the particles converge to the real Pareto front more quickly, a new learning mechanism is proposed. The way of learning from the first γ elite particles is still used. The difference is that the competition based learning will not be used. A particle from the first γ elite particles is randomly picked and then let the rest of the particles learn from it by formula 16 and 17. The P_w in formula 16 is changed to the position of the selected particle. Thus, the particles distribute more evenly on the real Pareto front and the convergence speed is improved. At the same time, the quality of the solution is higher when decision-makers select the rolling schedule setting parameters on the final Pareto optimal solution set.

When some of the first ten particles sorted by Pareto level and crowding distance fall into local optima, the other particles learning from them will also fall into the local optima. To solve this problem, the opposition-based learning (OBL) is proposed to make the particles jump out of the local optima.

The OBL was first proposed as a scheme in the field of machine intelligence in [10]. It was widely spread in the field of evolutionary computation and proved to be efficient. But its theoretical studies are still immature in multi-objective optimization [11]. Gao *et al.* proposed a velocity-free multi-objective particle swarm optimizer with centroid in [14]. The OBL was applied to generate the initial swarm and the better particle was chosen from the solution and its opposite solution. The OBL can increase the diversity of particles and accelerate the convergence rate of the population. Supposing x is a real number belonging to [a,b], its opposite solution $\hat{x}$ is defined as follows:

$$\hat{x} = a + b - x \tag{18}$$

Let $P(x_1, x_2, \ldots, x_D)$ be a point in D-dimensional space, $x_i \in [a_i, b_i], i = 1, 2, \ldots, D$. The opposite solution $\hat{P}$ is defined by $\hat{P}(\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_D)$ where

$$\hat{x}_1 = a_i + b_i - x_i \tag{19}$$

When there are particles in the first ten particles fall into local optima, the OBL method is applied to make them jump out of the local optima after every certain number of generations. In this paper, the OBL method is performed every *MaxGen*/10 generations, where *MaxGen* is the maximum number of generations. After that, the non-dominated sorting and crowding distance sorting are re-performed. Then the random particle learning strategy is implemented and update the next generation. The environmental selection is still adopted from the SPEA2 [25]. The pseudo-code of MOPSO-DOL is shown in **Algorithm 1**.

To verify the effectiveness of the proposed algorithm, the computational complexity is calculated. It has the same computational complexity whether considering using the OBL to jump out of local optima or not. The worst computational complexity of non-dominated sorting is $O(MN^2)$ where M is the number of objects and N is the population size. The computational complexity of the crowding distance based sorting is $O(MN \log N)$. Because the proposed algorithm does not use external archive to store the best particles, the worst computational complexity of environment selection is $O(N^3)$. Therefore, the worst computational complexity of MOPSO-DOL is $O(N^3)$. In practice, the setting of rolling schedule varies with different incoming strips which shows

TABLE 1. The parameters of the rolling mills.

Parameters	1# stand	2# stand	3# stand	4# stand	5# stand
Work roll diameter/mm	424.705	424.810	425.000	424.900	425.050
Work roll width/mm	1420	1420	1420	1420	1420
Back up diameter/mm	1300	1300	1300	1300	1300
Back up width/mm	1420	1420	1420	1420	1420
Maximun rolling force/kN	20000	20000	20000	20000	20000
Rated power of motor/kW	4200	4200	4200	4200	4200
Motor speed/rpm	300	400	400	400	400
Spacing between racks/m	5.5028	5.5028	5.5028	5.5028	5.5028

Algorithm 1 Pseudo-Code of MOPSO-DOL

Input: P(initialized population), $MaxGen$ (maximum number of generations), $Gen = 1$, γ(number of elite particles), N (population size)
Output: P(final population)

```
while Gen ≤ MaxGen do
    Perform the non-dominated sorting and crowding distance sorting to P;
    Select γ particles from P by ranking;
    if mod(Gen, MaxGen/10) == 0
      Update the positions of γ by formula 19;
      Perform the non-dominated sorting and crowding distance sorting to P;
      Select γ particles from P by ranking;
    end if
    for each particle p_i ∈ P do
      Randomly choose one elite particle from γ;
      Update the position and velocity of p_i by formula 16 and 17;
      P' ← P' + p_i;
    end for
    P' ← PolynomialVariations (P');
    P ← EnvironmentalSelection(P, P');
end while
return final population P
```

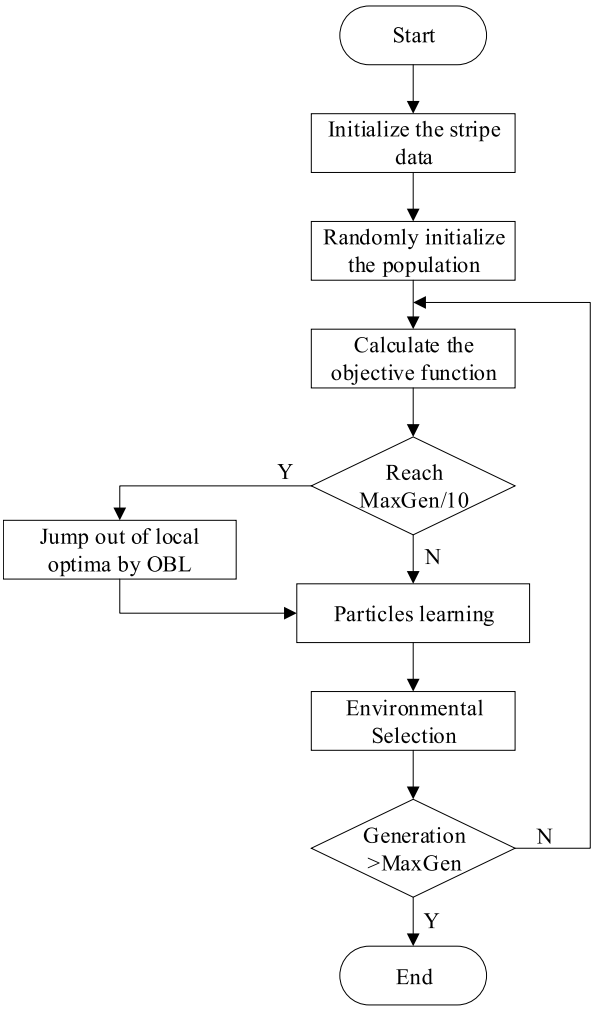


FIGURE 4. The flowchart of the rolling schedule optimization.

that the proposed algorithm can meet the demands of production. Finally, the flowchart of the optimization of rolling schedule is shown in Figure 4.

IV. SIMULATION RESULTS AND ANALYSIS

The following simulation results consist of three parts. The first part is to verify the effectiveness of the proposed mathematical model combines with deep neural network. The second part is to illustrate the validity of the proposed MOPSO-DOL on benchmark problems. The third part is to verify the application of the MOPSO-DOL on the optimization of rolling schedule.

A. THE SIMULATION OF THE ROLLING FORCE MODEL

The field data are used to verify the accuracy of the proposed model. The material of the steel is MRT4 and the width is 910 mm. The parameters of the rolling mills are shown in Table 1. After many experiments, the number of hidden layers of deep neural network is selected as five. The batch size is taken as 100. The activation function of the hidden layer is the ReLU and the output layer uses the sigmoid activation function. The number of training data is 40000, and 1000 data are used for testing.

In order to further explore the prediction accuracy of the proposed model, the deep neural network and GA-BP [20], [21] are chosen for comparison. Mean absolute percentage error (MAPE) and root mean square error (RMSE)

TABLE 2. Prediction accuracy of rolling force by different prediction models.

Number of stand	GA-BP		DNN		DNN combines with SIMS I		DNN combines with SIMS II		DNN combines with SIMS III	
	MAPE(%)	RMSE	MAPE(%)	RMSE	MAPE(%)	RMSE	MAPE(%)	RMSE	MAPE(%)	RMSE
1	1.008	92.075	0.967	89.194	0.963	88.793	**0.961**	**88.723**	0.968	89.234
2	2.358	249.402	**2.220**	237.481	2.226	238.530	2.237	239.213	**2.220**	**237.395**
3	2.211	249.384	1.823	213.342	1.831	213.466	1.830	213.322	**1.820**	**212.957**
4	2.953	308.316	2.224	246.166	2.215	**244.796**	2.233	245.599	**2.212**	246.229
5	1.330	126.249	0.998	97.266	1.008	98.030	1.011	98.387	**0.991**	**96.897**

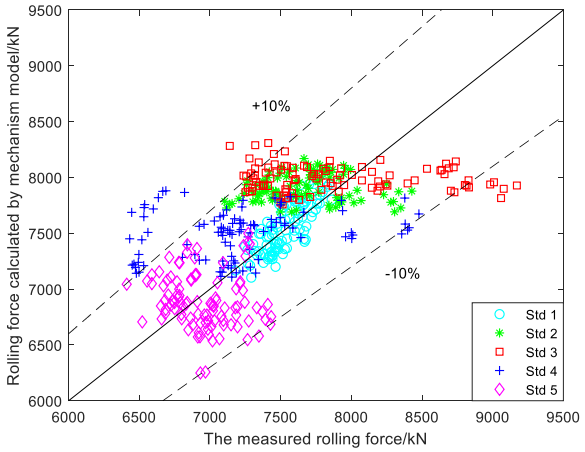


FIGURE 5. The rolling force calculated by the SIMS model.

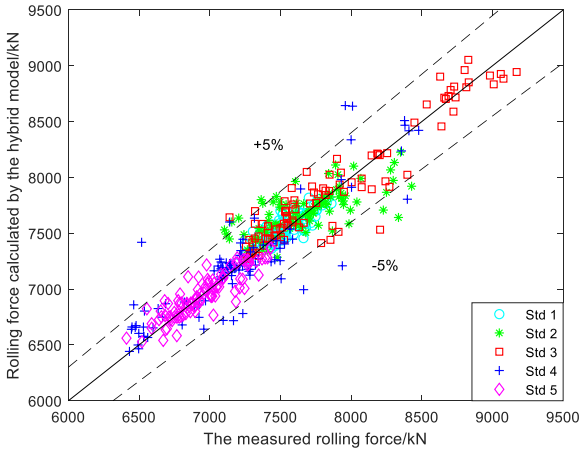


FIGURE 6. The rolling force calculated by the DNN combines with the SIMS III model.

is used as the comparison standard of accuracy. All of the test accuracies are the average value of 30 times of training in Table 2. And the minimum value is shown in bold. It can be seen that most of the calculation accuracy of the rolling force model of DNN combines with the SIMS III is higher than that of the other rolling force models. When the SIMS rolling force model is added to the output of the DNN, the noise will be produced. Taking the calculated value of the SIMS rolling force as one of the inputs of the DNN is equivalent to increasing the input characteristics of the neural network. Therefore, the prediction accuracy of the neural network will be improved.

The rolling force calculated by the SIMS model and DNN combines with the SIMS III model are shown in Figure 5 and Figure 6. As can be seen from the figures, the rolling force error calculated by the traditional method is within 10%. But the rolling force error is within 5% by the proposed model. And the proposed rolling force model is applied to the optimization of rolling schedule.

B. SIMULATION RESULTS ON BENCHMARK PROBLEMS

The proposed MOPSO-DOL is tested on the benchmark problems to confirm its performance. Since the problem of rolling schedule to be solved in this paper is a three-objective problem, this paper only tests on three-objective benchmark problems. The three-objective test functions DTLZ1-DTLZ7 and WFG1-WFG9 are selected for testing. The number of decision variables for DTLZ2-DTLZ6 and all WFG test problems is set to 12, DTLZ1 is set to 7, and DTLZ7 is set to 22. The inverted generational distance (IGD) metric is used to evaluate the performance of the algorithm. The smaller the IGD value, the better the performance of the algorithm. The proposed algorithm is compared with several state-of-the-art multi-objective particle swarm optimization algorithm, MMOPSO [12], MPSOD [13], CMOPSO [9], and three famous multi-objective evolutionary algorithms (MOEAs), NSGA-II [24], MOEA/D [23], and IBEA [26]. All parameters are set as suggested in the original references. The population size is set to 100, and the maximum evolutionary generation of the population is used as the termination condition of the algorithm. Except that the maximum evolution generation of DTLZ3 is set to 1000, the rest of the test problems are set to 300. Each test problem runs independently 30 times. The test functions and the source code of the compared algorithms are provided by PlatEMO [27].

Table 3 and Table 4 show the mean and standard deviation of IGD metric of each algorithm on standard test problem. The Wilcoxon rank sum test at the significance level of 5% is adopted, and the symbol '+', '−' and '≈' represent the compared algorithm is significantly better than, worse than and similar to the proposed MOPSO-DOL. The best average of each test function is marked with a gray background. It can be seen from the tables that comparing with the other algorithms, the proposed MOPSO-DOL performs well on most benchmark test problems which proves the effectiveness of the algorithm. The proposed MOPSO-DOL is applied to the optimization of rolling schedule.

TABLE 3. IGD values of the proposed MOPSO-DOL and three multi-objective PSO algorithms on DTLZ and WFG.

Problem	Obj.	MMOPSO	MPSOD	CMOPSO	MOPSO-DOL
DTLZ1	3	4.6668e-1 (3.90e-1) +	1.2605e+0 (4.46e-1) +	7.4265e+0 (5.44e+0) ≈	8.2121e+0 (3.89e+0)
DTLZ2	3	7.1756e-2 (2.48e-3) -	5.4953e-2 (1.55e-4) +	5.7587e-2 (8.39e-4) ≈	5.7061e-2 (8.69e-4)
DTLZ3	3	2.0536e-1 (3.26e-1) +	1.2951e+1 (4.03e+0) +	3.6186e+1 (2.39e+1) ≈	4.7138e+1 (3.43e+1)
DTLZ4	3	7.2148e-2 (2.79e-3) -	5.4964e-2 (1.75e-4) +	1.7885e-1 (3.06e-1) -	6.0469e-2 (1.70e-3)
DTLZ5	3	6.2363e-3 (4.42e-4) -	3.6550e-2 (2.03e-3) -	6.5979e-3 (6.44e-4) -	5.1822e-3 (5.01e-4)
DTLZ6	3	6.7449e-3 (6.98e-4) -	3.1514e-2 (1.03e-3) -	4.2141e-3 (4.67e-5) ≈	4.2239e-3 (4.53e-5)
DTLZ7	3	2.1890e-1 (2.26e-1) -	1.3249e-1 (6.32e-3) -	1.1881e-1 (1.55e-1) -	7.3791e-2 (5.18e-2)
WFG1	3	7.6605e-1 (8.22e-2) +	1.6197e+0 (5.69e-2) -	1.5094e+0 (1.72e-2) ≈	1.5122e+0 (2.41e-2)
WFG2	3	2.2788e-1 (9.92e-3) -	2.0821e-1 (9.18e-3) -	1.8069e-1 (3.79e-3) ≈	1.8116e-1 (4.79e-3)
WFG3	3	9.8000e-2 (1.70e-2) +	2.7085e-1 (1.31e-2) -	1.6177e-1 (1.30e-2) -	1.3648e-1 (1.05e-2)
WFG4	3	3.0405e-1 (1.05e-2) -	2.7022e-1 (5.26e-3) -	2.6247e-1 (5.74e-3) ≈	2.5976e-1 (4.84e-3)
WFG5	3	2.8978e-1 (9.48e-3) -	2.5556e-1 (2.14e-3) -	2.4802e-1 (6.29e-3) -	2.4354e-1 (5.01e-3)
WFG6	3	3.2062e-1 (4.25e-2) -	2.7358e-1 (7.67e-3) -	2.3816e-1 (4.73e-3) -	2.3543e-1 (7.59e-3)
WFG7	3	2.8690e-1 (9.06e-3) -	2.6354e-1 (5.96e-3) -	2.3253e-1 (6.03e-3) -	2.2443e-1 (2.63e-3)
WFG8	3	3.7181e-1 (1.30e-2) -	3.4463e-1 (7.84e-3) -	3.3964e-1 (6.47e-3) -	3.3493e-1 (5.34e-3)
WFG9	3	2.9221e-1 (2.56e-2) -	2.4905e-1 (3.38e-3) -	2.2173e-1 (4.12e-3) -	2.1914e-1 (3.38e-3)
+/-/≈		4/12/0	4/12/0	0/9/7	

TABLE 4. IGD values of the proposed MOPSO-DOL and three MOEAs on DTLZ and WFG.

Problem	Obj.	IBEA	NSGA-II	MOEA/D	MOPSO-DOL
DTLZ1	3	1.6954e-1 (2.85e-2) +	2.7706e-2 (1.29e-3) +	2.1153e-2 (5.87e-4) +	8.2121e+0 (3.89e+0)
DTLZ2	3	8.1548e-2 (3.21e-3) -	6.9318e-2 (2.21e-3) -	5.4467e-2 (9.58e-7) +	5.7061e-2 (8.69e-4)
DTLZ3	3	4.7725e-1 (6.62e-3) +	7.1145e-2 (3.26e-3) +	5.4676e-2 (1.47e-4) +	4.7138e+1 (3.43e+1)
DTLZ4	3	7.9313e-2 (2.52e-3) -	6.7635e-2 (2.19e-3) -	3.4945e-1 (3.47e-1) ≈	6.0469e-2 (1.70e-3)
DTLZ5	3	1.5925e-2 (1.53e-3) -	5.7246e-3 (2.27e-4) -	3.3783e-2 (6.59e-5) -	5.1822e-3 (5.01e-4)
DTLZ6	3	2.5206e-2 (3.67e-3) -	6.0127e-3 (2.91e-4) -	3.3872e-2 (4.63e-5) -	4.2239e-3 (4.53e-5)
DTLZ7	3	9.4801e-2 (7.63e-2) -	9.5927e-2 (6.92e-2) -	2.4154e-1 (2.23e-1) -	7.3791e-2 (5.18e-2)
WFG1	3	1.9143e-1 (8.47e-3) +	3.5224e-1 (3.92e-2) +	5.0790e-1 (9.33e-2) +	1.5122e+0 (2.41e-2)
WFG2	3	2.9558e-1 (5.72e-3) -	2.1936e-1 (1.08e-2) -	2.5617e-1 (9.08e-3) -	1.8116e-1 (4.79e-3)
WFG3	3	3.8787e-2 (1.39e-3) +	1.0090e-1 (1.88e-2) +	1.8824e-1 (3.13e-2) -	1.3648e-1 (1.05e-2)
WFG4	3	3.1591e-1 (1.13e-2) -	2.7171e-1 (9.19e-3) -	2.6461e-1 (6.44e-3) -	2.5976e-1 (4.84e-3)
WFG5	3	3.2418e-1 (1.13e-2) -	2.7815e-1 (1.01e-2) -	2.5285e-1 (4.06e-3) -	2.4354e-1 (5.01e-3)
WFG6	3	3.3251e-1 (1.54e-2) -	3.0616e-1 (1.28e-2) -	2.9170e-1 (1.80e-2) -	2.3543e-1 (7.59e-3)
WFG7	3	3.2199e-1 (1.12e-2) -	2.7868e-1 (1.02e-2) -	3.3660e-1 (3.81e-2) -	2.2443e-1 (2.63e-3)
WFG8	3	3.3671e-1 (9.61e-3) ≈	3.6467e-1 (1.20e-2) -	3.2341e-1 (8.45e-3) +	3.3493e-1 (5.34e-3)
WFG9	3	2.9340e-1 (1.09e-2) -	2.7563e-1 (9.38e-3) -	2.9553e-1 (3.10e-2) -	2.1914e-1 (3.38e-3)
+/-/≈		4/11/1	4/12/0	5/10/1	

C. APPLICATION ON THE OPTIMIZATION OF ROLLING SCHEDULE

In this section, the field data are used to verify the optimization results of the proposed MOPSO-DOL. The specific parameters of the cold rolling mills are shown in Table 1. The population evolution number Maxgen of MOPSO-DOL is set to 500 and the population size is set to 100. The number of elite particles is adopted as 10. The PlatEMO [27] platform is used for verification experiments.

After the Pareto optimal solution set is obtained, decision-makers need to select a set of solutions as the final rolling schedule settings. The weighted-sum approach in [28] is used to select the trade-off solution which is shown as follows:

$$\min f = w_1 \cdot \frac{f_1 - f_{1\min}}{f_{1\max} - f_{1\min}} + w_2 \cdot \frac{f_2 - f_{2\min}}{f_{2\max} - f_{2\min}} + w_3 \cdot \frac{f_3 - f_{3\min}}{f_{3\max} - f_{3\min}} \quad (20)$$

where w_1, w_2 and w_3 are the weight coefficient of the three objectives and $w_1 + w_2 + w_3 = 1$; $f_{i\max}$ and $f_{i\min} (i = 1, 2, 3)$ are the maximum and minimum value on the Pareto optimal solution set of the ith objective.

Decision-makers can determine the rolling schedule by setting the values of w_1, w_2 and w_3. For rolling power balance, set $w_1 = 0.2, w_2 = 0.5, w_3 = 0.3$; to prevent slippage, set $w_1 = 0.2, w_2 = 0.3, w_3 = 0.5$.

In order to verify the results of the multi-objective optimization of rolling schedule, the rolling schedule calculated by the multi-objective optimization algorithm is compared with the original rolling schedule. The optimized rolling schedules are marked rolling schedule No.1 and rolling schedule No.2, respectively. The comparison results are shown in Table 5. In order to make a clearer comparison between the original rolling schedule and the optimized rolling schedule, Figure 7 and Figure 8 show the distribution of the slip factors and the rolling power of five stands.

TABLE 5. The original schedule and the optimized rolling schedule.

NO. of Stand	Rolling schedule	Exit thickness/mm	Reduction /%	Tension /kN	Rolling force/kN	Power/kW	Slip factor
	Original schedule	1.269	34.7	160.521	7838.863	2946.2	0.2821
1	Rolling schedule No.1	1.188	38.9	141.500	7966.614	3168.6	0.2855
	Rolling schedule No.2	1.236	36.4	141.500	7084.946	2726.9	0.2800
	Original schedule	0.762	40.0	113.385	8455.166	3675.3	0.2302
2	Rolling schedule No.1	0.765	35.6	82.150	8455.126	3358.7	0.2002
	Rolling schedule No.2	0.766	38.0	82.150	7918.361	3314.7	0.2104
	Original schedule	0.460	39.6	77.969	7781.567	2611.2	0.1236
3	Rolling schedule No.1	0.515	32.7	70.641	9262.123	2826.3	0.1233
	Rolling schedule No.2	0.495	35.4	76.487	8387.024	2663.6	0.1307
	Original schedule	0.300	34.8	55.589	7500.996	1831.9	0.1300
4	Rolling schedule No.1	0.333	35.3	37.973	8844.859	2304.2	0.1360
	Rolling schedule No.2	0.333	32.7	38.727	8481.833	2082.6	0.1231
	Original schedule	0.220	26.7	12.240	7722.711	1325.5	0.0686
5	Rolling schedule No.1	0.220	33.9	12.860	6984.858	1425.0	0.0966
	Rolling schedule No.2	0.220	33.9	12.860	7134.466	1460.0	0.0968

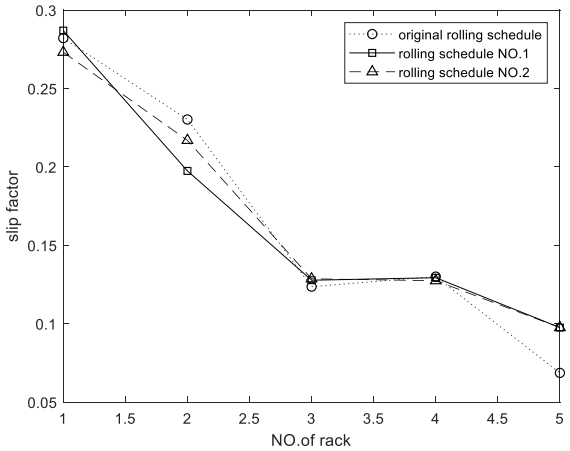


FIGURE 7. Comparison of the distribution of slip factors under different rolling schedules.

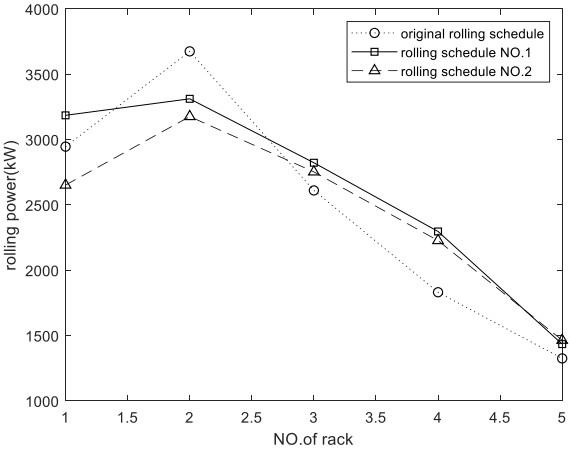


FIGURE 8. Comparison of rolling power distribution under different rolling schedules.

It can obviously see that the optimized rolling schedule can well balance the slip factor and rolling power between the five stands. Because the proposed algorithm can improve the diversity and convergence of particles in the evolution process. And OBL can make particles jump out of local optima.

V. CONCLUSION

In this study, to solve the problem that it is difficult to choose the coefficients when optimizing the rolling schedule by single-objective optimization, a multi-objective particle swarm optimizer with dynamic opposition-based learning is proposed. A new particle learning strategy is proposed to update the moving position of particles. Opposition-based learning makes particles jump out of local optima. It not only improves the quality of cold-rolled products, but also reduces the rolling energy consumption. The method based on deep neural network and mathematical model is used to calculate the rolling force. The simulation results show that the proposed MOPSO-DOL performs better than other multi-objective optimization algorithms on benchmark problems. It can improve the diversity and convergence of particles. The accuracy of the rolling force calculated by the proposed method is higher than that of the other models. Most errors of the rolling force fall within the range of $\pm 5\%$. Compared with the original rolling schedule, the optimized rolling schedule performs better on the rolling power and slippage preventing between five stands.

REFERENCES

[1] J. Sun, W. Peng, J. Ding, X. Li, and D. Zhang, "Key intelligent technology of steel strip production through process," *Metals*, vol. 8, no. 8, p. 597, Jul. 2018.

[2] X.-D. Qi, T. Wang, and H. Xiao, "Optimization of pass schedule in hot strip rolling," *J. Iron Steel Res. Int.*, vol. 19, no. 8, pp. 25–28, Aug. 2012.

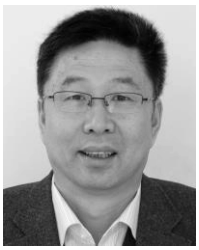
[3] S.-Z. Chen, X. Zhang, L.-G. Peng, D.-H. Zhang, J. Sun, and Y.-Z. Liu, "Multi-objective optimization of rolling schedule based on cost function for tandem cold mill," *J. Central South Univ.*, vol. 21, no. 5, pp. 1733–1740, May 2014.

[4] M. Poursina, N. T. Dehkordi, A. Fattahi, and H. Mirmohammadi, "Application of genetic algorithms to optimization of rolling schedules based on damage mechanics," *Simul. Model. Pract. Theory*, vol. 22, pp. 61–73, Mar. 2012.

[5] H.-N. Bu, Z.-W. Yan, and D.-H. Zhang, "Application of case-based reasoning-tabu search hybrid algorithm for rolling schedule optimization in tandem cold rolling," *Eng. Computations*, vol. 35, no. 1, pp. 187–201, Mar. 2018.

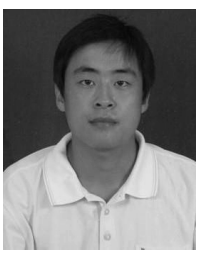
[6] J.-M. Yang, Q. Zhang, H.-J. Che, and X.-Y. Han, "Multi-objective optimization for tandem cold rolling schedule," *J. Iron Steel Res. Int.*, vol. 17, no. 11, pp. 34–39, Nov. 2010.
[7] W.-G. Li, X.-H. Liu, and Z.-H. Guo, "Multi-objective optimization for draft scheduling of hot strip mill," *J. Central South Univ.*, vol. 19, no. 11, pp. 3069–3078, Nov. 2012.
[8] Y. Li and L. Fang, "Robust multi-objective optimization of rolling schedule for tandem cold rolling based on evolutionary direction differential evolution algorithm," *J. Iron Steel Res. Int.*, vol. 24, no. 8, pp. 795–802, Aug. 2017.
[9] X. Zhang, X. Zheng, R. Cheng, J. Qiu, and Y. Jin, "A competitive mechanism based multi-objective particle swarm optimizer with fast convergence," *Inf. Sci.*, vol. 427, pp. 63–76, Feb. 2018.
[10] H. R. Tizhoosh, "Opposition-based learning: A new scheme for machine intelligence," in *Proc. Int. Conf. Comput. Intell. Modeling, Control Autom. Int. Conf. Intell. Agents, Web Technol. Internet Commerce (CIMCA-IAWTIC)*, vol. 1, 2006, pp. 695–701.
[11] S. Mahdavi, S. Rahnamayan, and K. Deb, "Opposition based learning: A literature review," *Swarm Evol. Comput.*, vol. 39, pp. 1–23, Apr. 2018.
[12] Q. Lin, J. Li, Z. Du, J. Chen, and Z. Ming, "A novel multi-objective particle swarm optimization with multiple search strategies," *Eur. J. Oper. Res.*, vol. 247, no. 3, pp. 732–744, Dec. 2015.
[13] C. Dai, Y. Wang, and M. Ye, "A new multi-objective particle swarm optimization algorithm based on decomposition," *Inf. Sci.*, vol. 325, no. 12, pp. 541–557, Dec. 2015.
[14] Y. Gao, L. Peng, F. Li, and X. Hu, "Velocity-free multi-objective particle swarm optimizer with centroid for wireless sensor network optimization," in *Proc. Int. Conf. Artif. Intell. Comput. Intell.*, 2012, pp. 682–689.
[15] W. Peng, J. G. Ding, D. H. Zhang, and D. W. Zhao, "A novel approach for the rolling force calculation of cold rolled sheet," *J. Brazilian Soc. Mech. Sci. Eng.*, vol. 39, no. 12, pp. 5057–5067, Dec. 2017.
[16] A. GuLi, C. Cui, Y. H. Xie, S. Ha and X. Wang, "Prediction of rolling force based on a fusion of extreme learning machine and self learning model of rolling force," in *Proc. Adv. Intell. Syst. Comput.*, vol. 686, Aug. 2017, pp. 3–11.
[17] J. Larkiola, P. Myllykoski, J. Nylander, and A. S. Korhonen, "Prediction of rolling force in cold rolling by using physical models and neural computing," *J. Mater. Process. Technol.*, vol. 60, nos. 1–4, pp. 381–386, Jun. 1996.
[18] S. Rath, A. P. Singh, U. Bhaskar, B. Krishna, B. K. Santra, D. Rai, and N. Neogi, "Artificial neural network modeling for prediction of roll force during plate rolling process," *Mater. Manuf. Processes*, vol. 25, nos. 1–3, pp. 149–153, Mar. 2010.
[19] Y. Mahmoodkhani, M. A. Wells, and G. Song, "Prediction of roll force in skin pass rolling using numerical and artificial neural network methods," *Ironmaking Steelmaking*, vol. 44, no. 4, pp. 281–286, Apr. 2017.
[20] L. X. Wei, X. Y. Wei, H. Sun, and H. Wang, "Prediction of aluminum hot rolling force based on deep network," *Chin. J. Nonferrous Met.*, vol. 28, no. 10, pp. 2070–2076, Oct. 2018.
[21] X. Qu, W. Jian Cai, and X. Lin Liu, "Cold rolling force prediction for copper strip based on applications of GA and BPNN," in *Proc. 2nd Int. Conf. Mechanic Autom. Control Eng.*, Jul. 2011, pp. 4818–4821.
[22] Z.-Y. Hu, J.-M. Yang, Z.-W. Zhao, H. Sun, and H.-J. Che, "Multi-objective optimization of rolling schedules on aluminum hot tandem rolling," *Int. J. Adv. Manuf. Technol.*, vol. 85, nos. 1–4, pp. 85–97, Jul. 2016.
[23] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.
[24] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
[25] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm," in *Proc. IEEE Conf. Evol. Methods Design, Optim. Control Appl. Ind. Problems*, May 2011, pp. 95–100.
[26] E. Zitzler and S. Kunzli, "Indicator-based selection in multiobjective search," in *Proc. Int. Conf. Parallel Problem Solving from Nature*, Sep. 2004, pp. 832–842.
[27] Y. Tian, R. Cheng, X. Zhang, and Y. Jin, "PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [Educational Forum]," *IEEE Comput. Intell. Mag.*, vol. 12, no. 4, pp. 73–87, Nov. 2017.
[28] S.-J. Jia, W.-G. Li, X.-H. Liu, and B. Du, "Multi-objective load distribution optimization for hot strip mills," *J. Iron Steel Res. Int.*, vol. 20, no. 2, pp. 27–32, Feb. 2013.

YUNLONG WANG received the B.S. degree from the School of Electrical Engineering, Hebei University of Science and Technology. He is currently pursuing the master's degree with the College of Information Science and Engineering, Northeastern University, China. His research interests include industrial big data analysis and intelligent control.



JINKUAN WANG (Member, IEEE) received the M.S. degree from Northeastern University, Shenyang, China, in 1985, and the Ph.D. degree from the University of Electro-Communications, Chofu, Japan, in 1993. In 1990, he joined the Institute of Space Astronautical Science, Japan, as a Special Member. In 1994, he was an Engineer with the Research Department, COSEL, Japan. Since 1998, he has been a Professor with the College of Information Science and Engineering, Northeastern University. His research interests include intelligent control, adaptive array, wireless sensor networks, and optimal operation of smart grid.

CHUNHUI YIN received the B.S. degree from the School of Electronic Information Engineering, South-Central University for Nationalities. She is currently pursuing the master's degree with the School of Computer and Communication Engineering, Northeastern University, China. Her research interests include industrial big data processing and data mining.



QIANG ZHAO received the Ph.D. degree in navigation guidance and control from the College of Information Science and Engineering, Northeastern University, Shenyang, China, in 2017. He is currently a Lecturer of measurement and control technology and instrumentation program with the School of Control Engineering, Northeastern University at Qinhuangdao. His research interests include electricity theft detection, fault diagnosis of renewable energy generation systems, and optimal operation of smart grid.