

This item was submitted to [Loughborough's Research Repository](#) by the author.
Items in Figshare are protected by copyright, with all rights reserved, unless otherwise indicated.

An estimation of distribution algorithm with resampling and local improvement for an operation optimization problem in steelmaking process

PLEASE CITE THE PUBLISHED VERSION

<https://doi.org/10.1109/TSMC.2019.2962880>

PUBLISHER

Institute of Electrical and Electronics Engineers

VERSION

AM (Accepted Manuscript)

PUBLISHER STATEMENT

© 2023 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

LICENCE

All Rights Reserved

REPOSITORY RECORD

Tang, Lixin, Chang Liu, Jiyin Liu, and Xianpeng Wang. 2023. "An Estimation of Distribution Algorithm with Resampling and Local Improvement for an Operation Optimization Problem in Steelmaking Process". Loughborough University. <https://hdl.handle.net/2134/11513571.v1>.

An Estimation of Distribution Algorithm with Resampling and Local Improvement for an Operation Optimization Problem in Steelmaking Process

Lixin Tang, *Senior Member, IEEE*, Chang Liu, Jiyin Liu, Xianpeng Wang, *Member, IEEE*

Abstract—This paper studies an operation optimization problem in a steelmaking process. Shortly before the tapping of molten steel from the converter furnace, end-point control measures are applied to achieve the required final molten steel quality. While it is difficult to build an exact mathematical model for this process, the control inputs and the corresponding outputs are available by collecting production data. We build a data-driven model for the process. To optimize the control parameters, an improved estimation of distribution algorithm (EDA) is developed using a probabilistic model comprising different distributions. A resampling mechanism is incorporated into the EDA to guide the new population to a broader and more promising area when the search becomes ineffective. To further enhance the solution quality, we add a local improvement to update the current best individual through simplified gravitational search and information learning. Experiments are conducted using real data from a converter steelmaking process. The results show that the algorithm can help to achieve the specified molten steel quality. To evaluate the proposed algorithm as a general optimization algorithm, we test it on some complex benchmark functions. The results illustrate that it outperforms other state-of-the-art algorithms across a wide range of problems.

Index Terms—Steelmaking process, data-driven model, estimation of distribution algorithm, resampling, local improvement.

I. INTRODUCTION

Many complex industrial processes are difficult to model precisely. However, most such processes can be observed, and there are often large amounts of historical data from the system operations. These data make it possible to optimize or improve the systems using data-driven models [1–5]. This paper studies one of such processes, the steelmaking process in converter furnaces. Steelmaking is one

of the most important stages in steel industry because the chemical contents of the steel are finalized at the end of steelmaking, while all the later stages can only change the physical shape and mechanical properties. A major step of steelmaking is performed in converter furnaces. A common type of converter furnace is a basic oxygen furnace (BOF). Shortly before the production of a charge of molten steel completes, the temperature and the carbon content are detected. End-point control measures are then applied to achieve the required final temperature and molten steel compositions. This control operation can be considered as a black-box model and the control parameters can then be optimized.

There has been previous research on modeling the operation [6–9]. However, most of these models have focused on predicting the end-point temperature and carbon content using information obtained just before the end-point control and the given control parameters. Tian and Mao [6] developed an ensemble extreme learning machine (ELM) based on a modified AdaBoost.RT algorithm for the end-point temperature prediction problem in a ladle furnace (LF). Xu *et al.* [7] utilized the spectrum distribution of the flame with support vector machine (SVM) to predict the end-point carbon content in BOF steelmaking. Liu *et al.* [8] predicted the end-point temperature and molten steel quality in BOF steelmaking through computer vision and general regression neural network. Shao *et al.* [9] used flame spectral analysis and a multi-class classification algorithm to predict the end-point carbon content in BOF steelmaking. To solve the end-point phosphorus content prediction problem, He and Zhang [10] presented a data-driven model based on principal component analysis and back propagation neural network. Some physical models [11] and data-driven control strategies [12] have been proposed for the process control and end-point control problems in the steelmaking process. Liu *et al.* [12] proposed an improved estimation of distribution algorithm (EDA) for solving end-point control problems in BOF steelmaking. The temperature and carbon content are considered in a black-box optimization model, but other quality indicators are not applied.

For industrial productions [13–15], the main purpose of black-box optimization is to optimize control parameters in the model. The main basics of a black-box optimization model are control variables, objectives, and constraints (such as performance constraints of production equipment, capacity constraints of production equipment, and a variety of complex function relationships among control variables). The objectives

This work was supported in part by the Major International Joint Research Project of the National Natural Science Foundation of China under Grant 71520107004, the Major Program of National Natural Science Foundation of China under Grant 71790614, the Fund for Innovative Research Groups of the National Natural Science Foundation of China under Grant 71621061, the 111 Project under Grant B16009, the National Natural Science Foundation of China under Grant 61702077, and the National Natural Science Foundation of China under Grant 61573086. (*Corresponding author: Lixin Tang.*)

Lixin Tang is with Key Laboratory of Data Analytics and Optimization for Smart Industry (Northeastern University), Ministry of Education (e-mail: lixintang@mail.neu.edu.cn).

Chang Liu is with Liaoning Engineering Laboratory of Data Analytics and Optimization for Smart Industry, Northeastern University, Shenyang, 110819, China (e-mail: lc1987328@126.com).

Jiyin Liu is with School of Business and Economics, Loughborough University, Leicestershire LE11 3TU, UK (e-mail: J.Y.Liu@lboro.ac.uk).

Xianpeng Wang is with Liaoning Key Laboratory of Manufacturing System and Logistics, Northeastern University, Shenyang, China (e-mail: wangxianpeng@ise.neu.edu.cn).

may be to minimize cost, minimize energy consumption, or maximize benefit.

In black-box optimization problems, the black-box model [16] is the basis of optimization. It is difficult to build exact mathematical models, but some approximate models [17] frequently work well enough. Some data analytics methods [18–20] have been used to describe the internal structure of black-box models, and some global optimization algorithms based on statistical models have been used to solve these expensive problems under uncertainty [21]. As shown in Fig. 1, the structure of a black-box model is based on an input–output mapping relationship. If the desired output is known, then the black-box optimization may be considered as an adaptive feedback process.

To achieve ideal outputs, many researchers have utilized evolutionary algorithms (EAs) to solve black-box optimization problems with single or multiple objectives [22–29]. Among the various intelligent optimization algorithms, EDAs have the advantages on the specific modeling and sampling mechanisms. The sampling mechanism generates new solutions using probabilistic models that are based on known information from previous samples. The specific mechanism reinforces the candidature of EDAs as promising black-box optimization tools [30]. Thus, EDAs have increasingly been applied and improved to solve multiobjective optimization problems by capturing the relationship between variables and objectives [31]. For the multimodal problems, clustering-based niching tactics [32] and maintenance of multiple sub-models [33] are skillfully applied to EDAs to explore promising areas. In dealing with high-dimension problems, restricted Boltzmann machine with clustering strategies [34] and model complexity control [35] have been added to EDAs to reduce their computational complexity. Kaban *et al.* [36, 37] conducted various studies on the performance of EDAs for high-dimension black-box optimization problems. Based on the aforementioned work on EDAs, there has been growing interest in the model structure and the linkages among variables.

With a single probabilistic model (such as Gaussian distribution, Cauchy distribution, or others) [38, 39], most of existing continuous EDAs have the disadvantages of inefficient local search ability or low convergence speed. Thus, alternatives with other single probabilistic models or with multiple probabilistic models [32, 40] have been employed to solve continuous optimization problems. A few of these have been very effective in solving some multimodal problems. A few hybrid optimization algorithms have also been incorporated into EDAs [41–43]. However, the approaches have still not taken full advantage of the varying search area in each generation and historic population information.

In this paper, we build a data-driven operation optimization model in BOF steelmaking, based on input and output data, which is considered as a black-box optimization problem. An EDA with resampling and local improvement (EDA-RL) is developed to optimize the end-point control parameters. The contributions of this paper to the literature in this field are as follows:

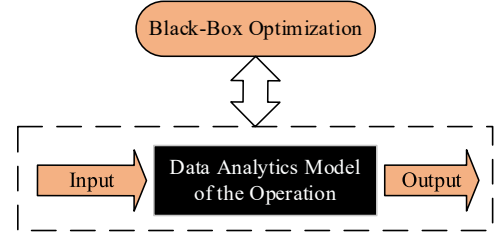


Fig. 1. Description of black-box optimization problem.

1) For the end-point control in BOF steelmaking, the temperature and quality indicators of molten steel are both considered as output objectives, and a data-driven operation optimization model is established using least-squares SVM (LSSVM) with a hybrid kernel function.

2) To solve the black-box optimization problem, an EDA framework with different probabilistic models is considered. Sampling from different models increases the diversity of the population, while the resampling mechanism enlarges the sampled region and helps guide the search to more promising areas. The new framework enhances the exploration ability of EDA.

3) An idea of local improvement is proposed to improve the exploitation ability of EDA. The current best individual is evolved using information from some superior individuals in the population through simplified gravitational search and information learning.

The remainder of this paper is organized as follows. Section II briefly describes the BOF steelmaking process and a data-driven optimization model. Section III presents our proposed algorithm in detail. In Section IV, the proposed algorithm is applied to a practical problem and is tested as a general optimization algorithm on some complex benchmark functions. Finally, conclusions are drawn in Section V, and future research is discussed.

II. THE BOF STEELMAKING PROCESS AND A DATA-DRIVEN OPERATION OPTIMIZATION MODEL

Because of the high temperature with fierce physical and chemical reactions in the BOF steelmaking process, it is difficult to establish a mechanism model and to control the temperature and the composition contents of molten steel. Consequently, the steelmaking process can be considered as a black-box optimization problem. In this paper, we build a data-driven optimization model for the end-point control problem in BOF steelmaking.

A. End-Point Control in BOF Steelmaking

Fig. 2 illustrates the structure of converter furnace. BOF steelmaking converts carbon-rich molten pig iron and steel scraps to molten steel through blowing oxygen. During the process, redundant carbon is burnt and released as gas. Other impurities are oxidized into slags on the surface, which can be separated. Auxiliary materials are also added to achieve the required content. After a series of physical and chemical reactions, the smelting process reaches the “end point” at which the following requirements need to be satisfied: 1) the

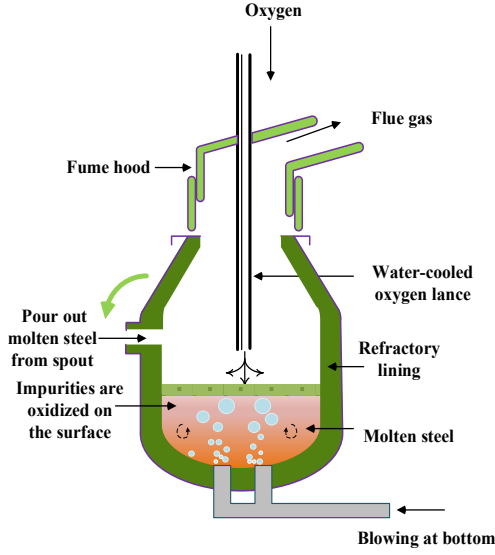


Fig. 2. Structure of converter furnace.

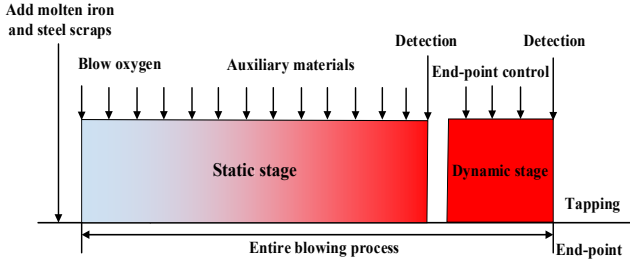


Fig. 3. Divided stages in the BOF steelmaking process.

carbon content is within the end-point control range, 2) the phosphorus content and sulfur content in molten steel do not exceed their specified upper limits, and 3) the tapping temperature is appropriate for steel refining and casting.

The process can be seen as consisting of two main stages, a static stage and a dynamic stage, as shown in Fig. 3. Most of the smelting time is spent in the static stage, which continues until the first sub-lance detection, shortly before the end-point at which the second detection is conducted. Between the two detections is the dynamic stage, within which end-point control is applied, based on the quality status obtained in the first detection and the final quality requirements.

The end-point control of BOF is very important in the late stage of steel smelting. Because of the complexity of desulphuration and dephosphorization, the removal of sulfur and phosphorus should be completed as soon as possible before the steelmaking process. To some extent, the carbon content and temperature are crucial to end-point control in BOF steelmaking. End-point control in the dynamic stage is conducted to adjust the carbon content and temperature by adding coolants (lime stone, etc.) or further blowing oxygen.

B. Data-Driven Operation Optimization Model

The input end-point control parameters include the rate of blowing oxygen, the amounts of auxiliary raw materials, and the rate of blowing inert gas at the bottom. The main output performance indicators are the temperature, the carbon content,

and the contents of other elements. However, because of the complex physical and chemical reactions that occur in the furnace at high temperature, it is difficult to build exact mechanism models expressing the relationship between the inputs and outputs. Operation state data can be sampled using multi-source sensors, with the temperature being observed using a flame analyzer, some composition contents in the molten steel being detected using throwing probe and sub-lance sampling, and gas quantity being detected using a gas analyzer. In practice, operators usually control the levels of input parameters to attain the desired outputs, based on experience and knowledge collected from historical data.

In this paper, a data-driven model is built to accurately control the end-point quality of molten steel in BOF steelmaking. The modeling method is based mainly on LSSVMs, proposed by Suykens and Vandewalle [44]. LSSVMs have faster computation speed compared with SVMs. Because the analytical formulas can be solved using linear equations in LSSVMs, rather than by means of quadratic optimization in SVMs. To improve the generalization ability of LSSVMs, this paper proposes an LSSVM method with a hybrid kernel function (HKLSSVM). The principle is introduced below.

Given a sample set of N data points $\{\mathbf{x}_i, y_i\}$, $i = 1, 2, \dots, N$, where $\mathbf{x}_i \in \mathbb{R}^d$ is the i th input data, d is the number of features, and $y_i \in \mathbb{R}$ is the corresponding i th output data, the input-output relationship is expressed by the following equation:

$$f(\mathbf{x}_i) = \mathbf{w}^T \cdot \phi(\mathbf{x}_i) + b, \quad i = 1, 2, \dots, N, \quad (1)$$

where $\phi(\mathbf{x}_i)$ is a feature map, with which \mathbf{x}_i can be mapped into a high-dimensional feature space. \mathbf{w}^T and b are the regression coefficients and the bias, respectively.

According to the principle of structural risk minimization, the LSSVM problem can be written as

$$\begin{aligned} \min_{\mathbf{w}, b, \mathbf{e}} J(\mathbf{w}, \mathbf{e}) &= \frac{1}{2} \|\mathbf{w}\|^2 + \frac{\gamma}{2} \sum_{i=1}^N e_i^2 \\ \text{s.t. } y_i &= \mathbf{w}^T \cdot \phi(\mathbf{x}_i) + b + e_i \\ \gamma &\geq 0, \end{aligned} \quad (2)$$

where γ is a penalty coefficient, which is set to a value in the range of [100, 1000], and e_i is a slack variable. Using the Lagrangian multiplier α_i , the above LSSVM problem can be expressed as

$$\text{Lag}(\mathbf{w}, b, \mathbf{e}, \alpha_i) = J(\mathbf{w}, \mathbf{e}) - \sum_{i=1}^N \alpha_i [\mathbf{w}^T \phi(\mathbf{x}_i) + b + e_i - y_i]. \quad (3)$$

By the Karush-Kuhn-Tucker condition, the following linear equations can be obtained:

$$\begin{bmatrix} 0 & \mathbf{I}^T \\ \mathbf{I} & \Omega + \gamma^{-1} \mathbf{I} \end{bmatrix} \begin{bmatrix} b \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{y} \end{bmatrix}, \quad (4)$$

where $\mathbf{I} = (1, \dots, 1)^T$, $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_N)^T$, $\mathbf{y} = (y_1, \dots, y_N)^T$, and $\Omega = \{\Omega_{ij} \mid i, j = 1, \dots, N\} = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$ is a symmetric matrix of the kernel function. $K(\mathbf{x}_i, \mathbf{x}_j)$ is a Mercer kernel function that is defined as follows:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \cdot \phi(\mathbf{x}_j), \quad i, j = 1, 2, \dots, N. \quad (5)$$

The LSSVM model can then be presented as

$$f(\mathbf{x}) = \sum_{i=1}^N \alpha_i K(\mathbf{x}, \mathbf{x}_i) + b. \quad (6)$$

To describe global and local performances by means of different kernel functions, a Gaussian kernel is combined with a polynomial kernel in this study to form a hybrid kernel function.

The Gaussian kernel function is defined as

$$K_1(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right), \quad i, j = 1, 2, \dots, N, \quad (7)$$

where σ is the width parameter of the kernel function, and the range of the parameter is set to $[1, 100]$.

The polynomial kernel function is defined as

$$K_2(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^T \mathbf{x}_j + 1)^{d_{\text{poly}}}, \quad i, j = 1, 2, \dots, N, \quad (8)$$

where d_{poly} is the order of the polynomial kernel function and is an integer in the range of $[1, 10]$. The hybrid kernel function can be expressed as

$$K(\mathbf{x}, \mathbf{x}_i) = (1 - \kappa) K_1(\mathbf{x}, \mathbf{x}_i) + \kappa K_2(\mathbf{x}, \mathbf{x}_i), \quad i = 1, 2, \dots, N, \quad (9)$$

where κ represents a scale coefficient, and it takes a value in the range of $[0, 0.005]$.

To obtain an accurate operation model, the values of the internal parameters (γ , σ , d_{poly} , and κ) are determined by a simple optimization algorithm [17] based on differential evolution (DE) [45].

Using the above data-driven operation model, we build a data-driven operation optimization model. To determine the values of the control parameters that would achieve the required quality of the molten steel, the final model with weight coefficients is transformed into the following form:

$$\begin{aligned} \min \hat{f}(\mathbf{x}) = & \omega_1 \sqrt{(f_1(\mathbf{x}) - y_1)^2} + \omega_2 \sqrt{(f_2(\mathbf{x}) - y_2)^2} \\ & + (1 - \omega_1 - \omega_2) \sum_{j=3}^6 \sqrt{(f_j(\mathbf{x}) - y_j)^2} \end{aligned} \quad (10)$$

$$s.t. \quad \mathbf{x}_{\min} \leq \mathbf{x} \leq \mathbf{x}_{\max},$$

where $\hat{f}(\mathbf{x})$ is the objective function, and $f_1(\mathbf{x})$, $f_2(\mathbf{x})$, $f_3(\mathbf{x})$, $f_4(\mathbf{x})$, $f_5(\mathbf{x})$, and $f_6(\mathbf{x})$ are operation models for temperature, carbon content, manganese content, silicon content, sulfur content, and phosphorus content, respectively. Given their importance to the molten steel quality, the temperature and carbon content have the greatest influences on molten steel. Thus, their weight coefficients (ω_1 and ω_2) are also considered as input features. They are both set to vary within the range of $[0.35, 0.5]$. Thus, the weight coefficient of the remaining component contents should be $1 - \omega_1 - \omega_2$. In Eq. (10), y_j is the target value for the corresponding operation optimization objective, where $j = 1, 2, \dots, 6$. The input control parameters \mathbf{x} are within the range of $[\mathbf{x}_{\min}, \mathbf{x}_{\max}]$.

In actual applications, the optimized input control parameters include the rate of blowing oxygen on the surface of molten steel, the rate of blowing argon at the bottom, the rate of blowing nitrogen at the bottom, the amounts of added supplementary coolants, the height of the oxygen lance, and two weight coefficients of the optimized model. The output

data are the temperature and the contents of carbon, manganese, silicon, sulfur, and phosphorus.

From the final operation optimization model, we can see that the objective function is complex: it is composed of six sub-functions and seven control parameters. Therefore, to optimize operation parameters, an effective operation optimization algorithm is necessary.

III. AN IMPROVED EDA

In this section, our research focuses on an operation optimization algorithm in BOF steelmaking. We first present an EDA with a hybrid distribution model, and then elaborate on ideas of resampling (R) and local improvement (L) that are incorporated in the improved EDA framework. Finally, the overall structure of EDA-RL is presented.

A. EDA with a Hybrid Distribution Model

Estimation of distribution algorithms [46] are EAs that were originally developed to solve combinatorial optimization problems. Subsequently, EDAs [47] were extended to continuous optimization. Unlike other EAs, EDAs are a type of sampling algorithms based on probability distributions, and the individuals in the population are generated by sampling rather than crossover and mutation operations. The basic idea of an EDA is that a promising probabilistic model is constructed by extracting relevant distribution information from superior individuals in the population. The sampled individuals are then derived from the probabilistic models.

Univariate marginal distribution algorithm (UMDA) [47] based on univariate Gaussian model was first proposed in 1998. This algorithm is based on the assumption that the variables are independent, which makes computation fast. Based on the assumption of Gaussian distribution with continuous variables, an extension of UMDA (UMDA^G) [47] was developed for many real-world continuous optimization problems. For Gaussian model with a small number of parameters, UMDA^G is generally considered as a typical EDA. However, with the increasing number of generations, Gaussian distribution gradually tends toward a small variance and an unchanged mean, leading to poor population diversity. In this study, to enhance the exploration ability of model sampling, Cauchy distribution is applied to the EDA, providing a wider sampling space than Gaussian model and avoiding the problem of sampled individuals falling into local optima. Thus, the hybrid distribution model (univariate Gaussian distribution combined with Cauchy distribution) is considered as the probabilistic model of the EDA.

Details of the EDA are described as follows:

1) *Initialization*: The overall parameters of the EDA are set in the initial stage, with NP denoting the number of individuals in the population, D representing the dimension of individuals, g_{\max} being the maximum generation number, and g being the current generation, which is initialized to 0. According to the search range of each individual, L_j and U_j denote the lower boundary and the upper boundary in the j th dimension, respectively, where $j = 1, 2, \dots, D$. The initial population is generated by sampling a uniform distribution. At each

Algorithm 1 EDA with a hybrid distribution model

Input: the overall parameters of the EDA, and set the proportions of individuals to be sampled from different distributions in the probability model.

- 1: Randomly generate an initial population by uniform distribution, and compute the objective function values of the individuals. The current best individual is set as the overall best one so far.
- 2: **While** the stopping criteria is not satisfied **do**
- 3: Select superior individuals from the current population;
- 4: Build a hybrid distribution probabilistic model based on the selected individuals;
- 5: Sample new individuals using the hybrid distribution model;
- 6: Repair the new individuals if necessary. Compute the objective function values of the sampled individuals, and update the current best individual;
- 7: Combine the selected individuals with the sampled individuals to form the new population;
- 8: Increase the generation number;

End while

Output: the overall best individual.

generation g , the i th individual can be regarded as \mathbf{z}_i^g , $i = 1, 2, \dots, NP$, and its corresponding objective function value can be expressed as $\hat{f}(\mathbf{z}_i^g)$. The parameter λ represents the selection rate of superior individuals, and then the number of superior individuals $H(=\lambda NP)$ is obtained.

2) *Selection*: Based on the above initialization, the l th most superior individual with D -dimension in the current generation g is represented as $\mathbf{z}_l^g = (z_{l,1}^g, z_{l,2}^g, \dots, z_{l,D}^g)$, $l = 1, 2, \dots, H$, which is selected from the superior objective function values in the current population.

3) *Modeling*: Two different types of distributions are used to establish the probabilistic model of the EDA. Based on univariate marginal distribution, Gaussian probability density function is constructed with superior individuals according to Eq. (11):

$$p(\mathbf{z}_{l,j}^g) = \frac{1}{\sqrt{2\pi(\delta_j^g)^2}} \exp\left(-\frac{1}{2(\delta_j^g)^2} (z_{l,j}^g - u_j^g)^2\right), \quad (11)$$

where $l = 1, 2, \dots, H$, $j = 1, \dots, D$, u_j^g and δ_j^g are the j th dimension mean and the j th dimension variance of superior individuals, respectively.

In addition, Cauchy distribution is also applied to the modeling of the EDA, Cauchy probability density function is expressed as

$$p(\mathbf{z}_{l,j}^g) = \frac{1}{\pi} \frac{\hat{b}}{\hat{b}^2 + (z_{l,j}^g - \hat{a})^2}, \quad l=1, \dots, H, \quad j=1, \dots, D, \quad (12)$$

where \hat{a} ($=0$) is a location parameter, and \hat{b} ($=1$) is a scale parameter.

4) *Sampling*: To achieve a trade-off between the exploration and exploitation search ability of the EDA, sampling strategies

[32, 38] are used to generate new individuals. The sampling process is described as follows:

$$\eta_p^g \leftarrow \begin{cases} \mathbf{u}^g + \text{Ga}_p^g \delta^g, & p=1, \dots, num, \\ \mathbf{u}^g + \text{Ca}_p^g \delta^g, & p=num, \dots, NP-H, \end{cases} \quad (13)$$

where η_p^g denotes a sampled individual. Ga_p^g and Ca_p^g are a standard normal random number and a standard Cauchy random number, respectively. The number of individuals sampled from Gaussian distribution $num(=\rho(NP-H))$ is obtained by a sampling rate ρ , where $0 < \rho < 1$. For the hybrid sampling mechanism, \mathbf{u}^g and δ^g are the mean vector and the variance vector of superior individuals in the g th generation, respectively.

5) *Repairing*: If the sampled individual is beyond the search range, a repairing strategy is carried out. Infeasible individuals are regenerated randomly within the search range. The objective function values of sampled individuals are then calculated, and the current best individual is updated.

6) *Replacement*: The offspring population consists of the sampled individuals and the previous superior individuals.

7) *Stopping condition*: If the algorithm satisfies the stopping criteria (the given maximum evaluation number S or the maximum generation number g_{\max}), the algorithm is terminated, and the overall best individual is acquired. Otherwise, the algorithm continues to the next generation.

The main flow of the EDA with a hybrid distribution model is outlined in **Algorithm 1**.

B. Resampling Strategy

In general, the objective functions behind complex black-box optimization problems are multimodal, and so there exist a large number of local optima. The aforementioned approach mainly emphasizes on the model structure. Although the hybrid distribution model can solve multimodal problems to a certain degree, it may take a long time to reach promising areas.

In this paper, with the increasing number of generations, the sampled individuals will concentrate gradually. In order to enlarge its sampling region, as well as to improve the global search ability of the population, we introduce a resampling strategy that can help guide the population sampling. The resampling strategy is executed when the following two conditions are satisfied. One condition is that the best objective function value in the current population is unchanged for certain generations. The other one is that $eva < r_eva$, where eva is the current evaluation number, and r_eva is given by $\theta \times S$, where θ represents a resampling rate, and S is a given maximum number of evaluations. The following shows the specific steps of resampling.

1) *Enlarge the sampling region*: Based on the current superior individuals' location information, the initial resampling region can be identified as follows:

$$\tau_j = \max_{1 \leq l \leq H} z_{l,j}^g, \quad v_j = \min_{1 \leq l \leq H} z_{l,j}^g, \quad j=1, \dots, D. \quad (14)$$

The hyper cube $[v_j, \tau_j]$, $j=1, 2, \dots, D$, contains all current superior individuals and represents the current sampling region.

Algorithm 2 Resampling strategy

Input: the initialization parameters of resampling.

1: Enlarge the sampling region:

For $j = 1 : D$

$$\tau_j = \max_{1 \leq l \leq L} z_{l,j}^g, \quad \nu_j = \min_{1 \leq l \leq L} z_{l,j}^g.$$

If $\delta_j^g > \zeta$

$$U_j^* = \tau_j, \quad L_j^* = \nu_j.$$

Else

$$U_j^* = \tau_j + \text{rand}(0, 1)(U_j - \tau_j).$$

$$L_j^* = \nu_j - \text{rand}(0, 1)(\nu_j - L_j).$$

End if

End for

2: Resampling:

For $o = 1 : 2NP$

For $j = 1 : D$

$$\hat{z}_{o,j} = L_j^* + \text{rand}(0, 1)(U_j^* - L_j^*).$$

End for

$$s_o = \|\hat{z}_o - \mathbf{z}_{best}\|_2.$$

End for

3: Select NP resampled individuals with the smallest s_o to form a population. Evaluate the individuals in the population and select superior individuals from this population, and recalculate \mathbf{u}^g and δ^g based on this set of superior individuals.

Output: the resampled population.

In order to avoid falling into the local optima, the sampling region needs to be enlarged. The region for resampling is determined by the following:

If $\delta_j^g > \zeta$

$$U_j^* = \tau_j, \quad L_j^* = \nu_j, \quad j = 1, \dots, D, \quad (15)$$

Otherwise,

$$\begin{aligned} U_j^* &= \tau_j + \text{rand}(0, 1)(U_j - \tau_j), \quad j = 1, \dots, D, \\ L_j^* &= \nu_j - \text{rand}(0, 1)(\nu_j - L_j), \quad j = 1, \dots, D, \end{aligned} \quad (16)$$

where ζ is a threshold value of variance, U_j^* and L_j^* are the upper and lower boundaries of the resampling region, respectively. $\text{rand}(0, 1)$ denotes a random number from 0 to 1.

2) *Resampling*: $W (= 2NP)$ individuals are generated from uniform distribution over the resampling region. The j th dimension of the o th resampled individual, $\hat{z}_{o,j}$, $o = 1, 2, \dots, W$, $j = 1, \dots, D$, is obtained as follows:

$$\hat{z}_{o,j} = L_j^* + \text{rand}(0, 1)(U_j^* - L_j^*). \quad (17)$$

The Euclidean distance s_o between each of the resampled individuals \hat{z}_o and the overall best individual \mathbf{z}_{best} is calculated as

$$s_o = \|\hat{z}_o - \mathbf{z}_{best}\|_2, \quad o = 1, 2, \dots, W. \quad (18)$$

After that, NP smallest distances are selected. The corresponding individuals are used to replace the current

population and their objective function values are calculated. Superior individuals from this population are then selected, and used to re-calculate \mathbf{u}^g and δ^g .

To retain the overall best objective function value and the overall best individual, the current best individual \mathbf{z}_{cbest} and the current best objective function value are stored to an external archive before each resampling. The main flow of the resampling strategy is shown in **Algorithm 2**.

Note that resampling may be performed only when $eva < r_eva$. For the first generation after $eva = r_eva$ (and only for this one generation), the overall best individual in the external archive is used to replace the current best individual.

C. Local Improvement

Because the distinctive sampling mechanism of the EDA lacks the local search ability, in order to make full use of superior information between sampled individuals, a local improvement step based on the simplified gravitational search [48] and information learning is introduced. In this step, the current best individual \mathbf{z}_{cbest} is updated by simplified gravitational search, evolving with information from some superior individuals in the current population. Moreover, \mathbf{z}_{cbest} learns from the current best sampled individual, as well as the evolved best individuals. This local improvement enhances the local search ability of the algorithm. It applies only to the recorded \mathbf{z}_{cbest} and so does not affect the current sampling.

1) *Evolution with superior individuals*: In the evolution, simplified gravitational search is applied to \mathbf{z}_{cbest} using information from some superior individuals in the current population. First, the $np (= \beta NP)$ individuals in the current population are selected and sequenced, here β is a proportion of superior individuals. These are denoted as $\mathbf{g}_1^g, \mathbf{g}_2^g, \dots, \mathbf{g}_{np}^g$ in ascending order of their objective function values.

Using each of the selected individuals \mathbf{g}_k^g , $k = 1, 2, \dots, np - 1$, an acceleration \mathbf{a}_k is computed and applied to \mathbf{z}_{cbest} . To do so, the gravitational mass M_k , Euclidean distance to another individual R_{k,k_1} , and the gravitational force \mathbf{F}_k are first calculated in the following steps.

Gravitational mass:

$$M_k = \frac{m_k}{\sum_{t=1}^{np-1} m_t}, \quad k = 1, 2, \dots, np - 1, \quad (19)$$

$$\text{where } m_k = \frac{\hat{f}(\mathbf{g}_k^g) - \hat{f}(\mathbf{g}_{np}^g)}{\hat{f}(\mathbf{z}_{cbest}) - \hat{f}(\mathbf{g}_{np}^g)}, \quad k = 1, 2, \dots, np - 1. \quad (20)$$

\mathbf{g}_{np}^g is the worst individual among the selected superior individuals.

Euclidean distance:

$$R_{k,k_1} = \|\mathbf{g}_k^g - \mathbf{g}_{k_1}^g\|_2, \quad k = 1, 2, \dots, np - 1. \quad (21)$$

k_1 is an integer number randomly chosen from $1, \dots, np - 1$,

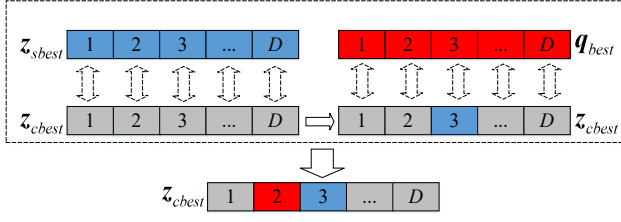


Fig. 4. Sketch of information learning.

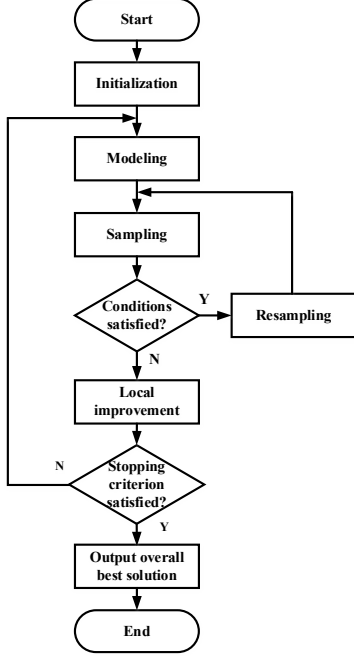


Fig. 5. Overall structure of the proposed algorithm.

and $\mathcal{G}_k^g \neq \mathcal{G}_{k_1}^g$.

Gravitational force:

$$\mathbf{F}_k = G \frac{M_k \times M_{k_1}}{R_{k,k_1}} (\mathcal{G}_k^g - \mathcal{G}_{k_1}^g), \quad k = 1, 2, \dots, np-1, \quad (22)$$

where G is a function with the initial value G_0 , we define it as

$$G = G_0 \exp\left(-\varepsilon \left(\frac{eva}{S}\right)^2\right), \quad (23)$$

where eva is the current number of evaluations in the algorithm, S is the total number of evaluations, and ε is a decreasing coefficient.

The acceleration \mathbf{a}_k can then be calculated:

$$\mathbf{a}_k = \frac{\mathbf{F}_k}{M_k}, \quad k = 1, 2, \dots, np-1. \quad (24)$$

Adding \mathbf{a}_k to \mathbf{z}_{cbest} , an evolved individual is obtained by

$$\begin{aligned} \mathbf{q}_k &= \mathbf{z}_{cbest} + \mathbf{a}_k \\ &= \mathbf{z}_{cbest} + G \frac{M_{k_1}}{R_{k,k_1}} (\mathcal{G}_k^g - \mathcal{G}_{k_1}^g), \quad k = 1, 2, \dots, np-1. \end{aligned} \quad (25)$$

For each $k = 1, 2, \dots, np-1$, there is an evolved individual \mathbf{q}_k . Let \mathbf{q}_{best} be the best among them. If \mathbf{q}_{best} is better than \mathbf{z}_{cbest} , \mathbf{z}_{cbest} is updated.

As can be seen from the above description, this gravitational

Algorithm 3 Local improvement

Input: the initialization parameters of local improvement.

1: Evolution with superior individuals: Apply an evolutionary strategy to \mathbf{z}_{cbest} using information of randomly chosen superior individuals from the current population. Update \mathbf{z}_{cbest} if \mathbf{q}_{best} is better than \mathbf{z}_{cbest} .

2: Information learning: For each dimension of \mathbf{z}_{cbest} , replace it with that of another solution if this leads to improvement. Update \mathbf{z}_{cbest} if the result is better. This learning operation may be performed for two rounds using \mathbf{z}_{sbest} and \mathbf{q}_{best} as “another solution”. If any of them is the same as \mathbf{z}_{cbest} , that round is skipped.

Output: the current best individual and its function value.

search is much simplified from its original version in [48]. It applies the gravitational forces individually to only \mathbf{z}_{cbest} , rather than using their combination to every solution. Moreover, as the best individual in the current population, \mathbf{z}_{cbest} is assumed not to change unless being improved. Therefore, each evolved solution is obtained by changing \mathbf{z}_{cbest} with the acceleration for the current generation only, and the velocity is assumed always gets back to 0 then. This is why velocity variables are not used.

If the differences between the superior individuals are very small, the resulting acceleration would not cause any obvious evolution. Therefore, the evolutionary strategy based on superior individuals will not be executed when the variance of the samples is below a certain value.

2) *Information learning*: Sometimes the quality of \mathbf{z}_{cbest} may be impaired because of the information in a few dimensions while the information in the corresponding dimensions of other individuals may be better. To make use of the potentially useful information of the superior individuals, we perform a learning operation for each dimension of \mathbf{z}_{cbest} . If the resulting individual is better, \mathbf{z}_{cbest} is updated. As shown in Fig. 4, information of \mathbf{z}_{cbest} learns from that of the current best sampled individual \mathbf{z}_{sbest} and \mathbf{q}_{best} . The corresponding learning operation is skipped if \mathbf{z}_{cbest} is the same to any of \mathbf{z}_{sbest} and \mathbf{q}_{best} .

The main flow of local improvement is shown in **Algorithm 3**. In later generations, it is expected that the search has reached the most promising region and the focus should be mainly on exploitation. Therefore, for the generations after $eva = r_eva$, if $f(\mathbf{z}_{cbest}) > f(\mathbf{z}_{best})$, then \mathbf{z}_{best} will be used as \mathbf{z}_{cbest} in the local improvement.

D. The Proposed EDA-RL

This paper makes full use of the sampled information, incorporating a resampling (R) strategy and a local improvement (L) strategy into the EDA. The overall structure of the proposed algorithm (EDA-RL) is described in Fig. 5.

The hybrid distribution function in modeling can help strengthen the diversity of the population. The resampling strategy enlarges the search scale of the EDA, which enhances the exploration ability of EDA-RL. Furthermore, the local improvement based on simplified gravitational search and information learning is added to improve the exploitation search ability of EDA-RL.

IV. EXPERIMENTS

The effectiveness of EDA-RL is verified in this section by its application to a practical end-point control problem in BOF steelmaking. To further demonstrate the generalization ability of our proposed algorithm, we compare it with other state-of-the-art black-box optimization algorithms through experiments on some complex benchmark problems.

A. Experimental Setting

1) *Experimental platform*: The experiments are conducted on a personal computer with an Intel Core i7-6700 3.40-GHz CPU, 16-GB RAM, and a 64-bit Windows 7 system. The algorithms are implemented in C++ on the Microsoft Visual Studio 2008 platform.

2) *Parameter setup*: The main parameters of EDA-RL are listed in TABLE I. The population size is set based on the problem dimension D . The selection rate λ is set to 30% so that the number of selected individuals is sufficient for modeling. In general, the resampling rate θ should be set larger for problems with a larger number of local optima. For our problem, the effects of θ and the decreasing coefficient ε are analyzed in the next subsection. To reduce the effect of randomness on the experimental results, 30 independent runs are conducted for the practical application problem. The stopping criterion is set to $1000 \times D$ maximum function evaluations (MaxFEs).

B. Results and Discussions

To verify the practicability of EDA-RL, we test it on an actual application problem. Based on real-world data from a Chinese steel plant, we solve the operation optimization problem in BOF steelmaking. The end-point control problem is formulated using the data-driven operation model that is based on one month of historical data. The problem is then solved with EDA-RL. The performance is analyzed by comparing the results of different strategies.

1) *Data processing*: Practical production data for one month are collected from the steelmaking process. Each charge of steel production has an end-point control problem and generates a set of operation data. A total of more than 300 data sets are collected in the month. Some of the data sets are abnormal. They are identified by means of clustering and then are removed. Some incomplete data sets are complemented using standard statistical method. After the preprocessing, 218

valid data sets are obtained and used in the experiment, 198 for building the black-box operation model and 20 for testing the operation optimization algorithm. In the optimization problem, the values of control variables vary within the following ranges. The rate of blowing oxygen in the reaction process varies from 0 to 1500 m³/min, the height of the oxygen lance varies from 8 to 20 m, the rate of blowing nitrogen varies from 0 to 50 m³/min, the rate of blowing argon at the bottom varies from 0 to 50 m³/min, the range of supplementary coolants varies from 0 to 1 ton, and two weight coefficients of operation optimization model vary between 0.35 and 0.5. Considering the large differences in the values of different variables, they are normalized in the operation optimization model.

2) *Comparisons with different kernel functions*: To verify the effectiveness of operation models (such as temperature [T], carbon content [C], manganese content [Mn], silicon content [Si], sulfur content [S], and phosphorus content [P]), the data-driven operation models are tested using different kernel functions. Three performance indices, including the root mean square error (RMSE), mean absolute error (MAE), and maximum absolute error (MAXAE), are used to evaluate the performance of operation models. The results are listed in TABLE II. We can find that the hybrid kernel function obtains 16 best results and yields better results than Gaussian kernel function and polynomial kernel function. It follows that the hybrid kernel function can be used to build accurate operation optimization model.

3) *Parameter analysis*: The parameters selection may sometimes have a great influence on the performance of the

TABLE II
COMPARISONS WITH DIFFERENT KERNEL FUNTIONS IN MODELS

Model	Index	Different Kernel Functions		
		Polynomial kernel	Gaussian kernel	Hybrid kernel
T	RMSE	6.20E+00	3.52E+00	3.44E+00
	MAE (°C)	4.77E+00	2.83E+00	2.80E+00
	MAXAE(°C)	1.71E+01	7.78E+00	7.56E+00
C	RMSE	9.35E-03	5.42E-03	5.41E-03
	MAE (%)	6.63E-03	4.34E-03	4.34E-03
	MAXAE(%)	2.36E-02	1.52E-02	1.52E-02
Mn	RMSE	4.72E-02	3.12E-02	3.14E-02
	MAE (%)	3.92E-02	2.56E-02	2.58E-02
	MAXAE(%)	1.02E-01	7.62E-02	7.60E-02
Si	RMSE	4.00E-04	3.75E-04	3.72E-04
	MAE (%)	1.65E-04	1.16E-04	1.12E-04
	MAXAE(%)	2.06E-03	2.05E-03	2.05E-03
S	RMSE	1.31E-02	1.23E-02	1.23E-02
	MAE (%)	8.30E-03	6.60E-03	6.60E-03
	MAXAE(%)	4.62E-02	4.07E-02	4.07E-02
P	RMSE	2.61E-03	1.90E-03	1.85E-03
	MAE (%)	1.99E-03	1.35E-03	1.29E-03
	MAXAE(%)	6.54E-03	5.85E-03	5.77E-03
No. of Best		0	8	16

TABLE I

THE SETTING OF MAIN PARAMETERS FOR EDA-RL

Parameter	NP	λ	ρ	ζ	β	θ	ε	G_0
Value	100	30%	90%	10	20%	60%	10	100

TABLE III

PERFORMANCE COMPARISONS OF EDA, EDA-R AND EDA-RL

Algorithm	Average Minimum Objective Function Value			
	MAXAE	MINAE	MAE	STD
EDA	3.69E-01	1.11E-02	1.22E-01	1.05E-01
EDA-R	3.55E-01	8.76E-03	1.19E-01	1.03E-01
EDA-RL	3.59E-01	8.28E-03	1.16E-01	1.01E-01

algorithm. We analyze this by considering two key parameters, the resampling rate θ and the decreasing coefficient ε , as shown in TABLE A-I and TABLE A-II in the Appendix. When θ is set to 60%, EDA-RL yields better results than others. For the local improvement, EDA-RL with $\varepsilon = 10$ is better than EDA-RL with other parameter settings. Based on these results, we infer that if the resampling rate θ is too small, the individuals in the current population will concentrate fast, and the diversity will also be reduced. Conversely, if θ is too large, EDA-RL will have a powerful exploration ability but poor exploitation ability. We also infer that, to a certain extent, a smaller ε yields better performance than a large ε . However, if the ε is too small, the speed will decrease more rapidly, and the search scope will be trapped into a local optimum solution. These findings illustrate why appropriate parameter values can maintain a balance between the global search and local search of the algorithm.

4) *Comparison results*: The results of 20 real-time test instances obtained after normalizing are listed in TABLE A-III in the Appendix. The mean error and the corresponding standard deviation are the main evaluation indices, and the best results are shown in bold font. If the results are smaller than $1.0E-08$, we regard them as zero (all results in the subsequent tables are interpreted in the same way). Standard deviations are indicated using the symbol “ \pm ”. Clearly, EDA-RL yields the best results for 16 of the 20 instances, outperforming the other comparison strategies tested.

The average performance index values for the 20 real-time test instances are shown in TABLE III, where MINAE and STD denote the minimum absolute error and standard deviation, respectively. Taking into consideration the main expectation indices, we define the expectation output of temperature from 1640°C to 1680°C , and the expectation output of carbon content is set between 0.02% and 0.04% . We can see that EDA-RL yields better performance according to MAE. The results illustrate that EDA-RL can provide better solutions for various production furnaces. In addition, EDA-RL also generates ideal outputs for the temperature and contents of molten steel components and provides a reliable control scheme for operators. For the four practical instances

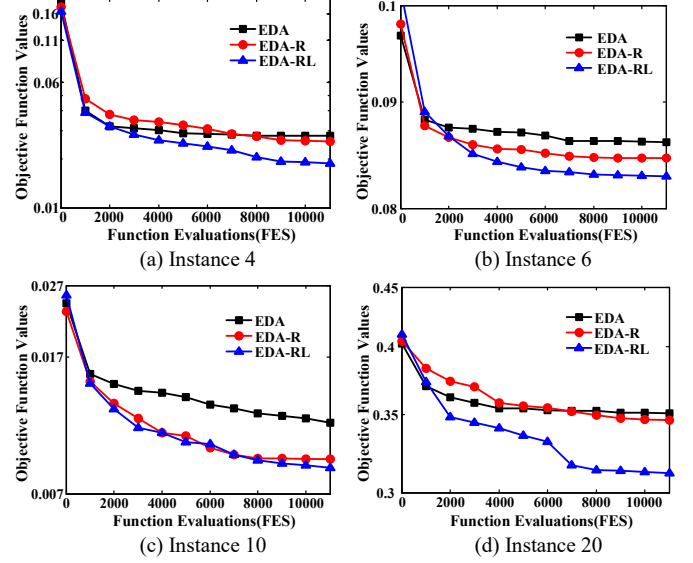


Fig. 6. The convergence curves of EDA, EDA-R and EDA-RL for four practical problems: (a) Instance 4, (b) Instance 6, (c) Instance 10, and (d) Instance 20.

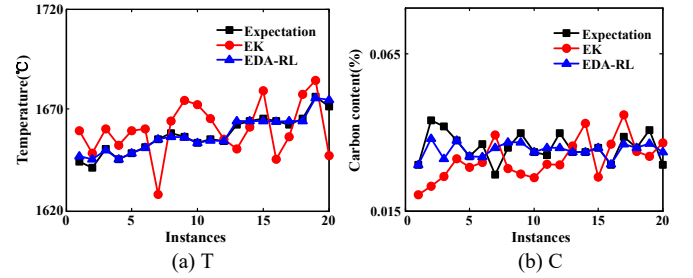


Fig. 7. Control graphs by EK and EDA-RL: (a) T, and (b) C.

listed in TABLE A-III in the Appendix, the convergence curves of EDA, EDA-R, and EDA-RL are shown in Fig. 6. The diagrams show that EDA-RL not only has a faster convergence than EDA-R and EDA, but also obtains better solutions than the other strategies.

5) *Practicability comparisons*: Under normal operating environment conditions, operators can adjust control parameters by combining their expert knowledge (EK) and experience with the mechanism model. However, smart industries often need reliable controls for quality and safety, and if the furnace state exhibits abnormal fluctuations, EK controls may lead to unreasonable or inaccurate operation results. The results obtained with EDA-RL and EK are presented in TABLE A-IV in the Appendix and in Fig. 7, where the expectation value is the given objective value. From these results, we can observe that EDA-RL obtains smaller absolute error values than EK and that the practical outputs of

TABLE IV

PERFORMANCE COMPARISONS BETWEEN EK AND EDA-RL

Objective	EK				EDA-RL			
	MAXAE($^{\circ}\text{C}/\%$)	MINAE($^{\circ}\text{C}/\%$)	MAE	STD	MAXAE($^{\circ}\text{C}/\%$)	MINAE($^{\circ}\text{C}/\%$)	MAE	STD
T	2.70E+01	1.00E+00	1.19E+01	6.78E+00	4.11E+00	6.22E-03	1.05E+00	1.21E+00
C	1.60E-02	1.00E-03	6.30E-03	3.63E-03	8.58E-03	4.00E-06	2.10E-03	2.46E-03

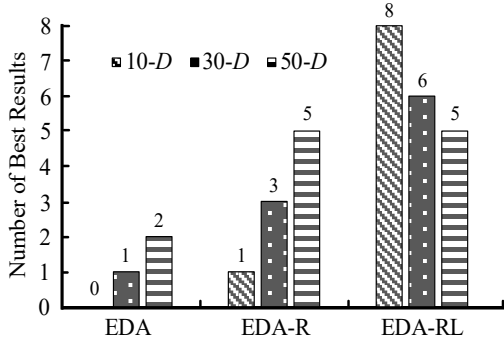


Fig. 8. Number of best results on comparison strategies.

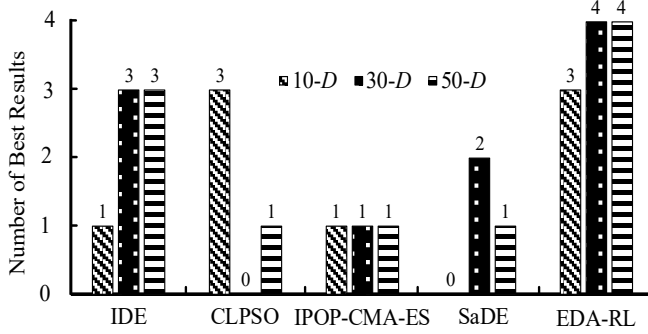


Fig. 9. Number of best results between EDA-RL and state-of-the-art algorithms.

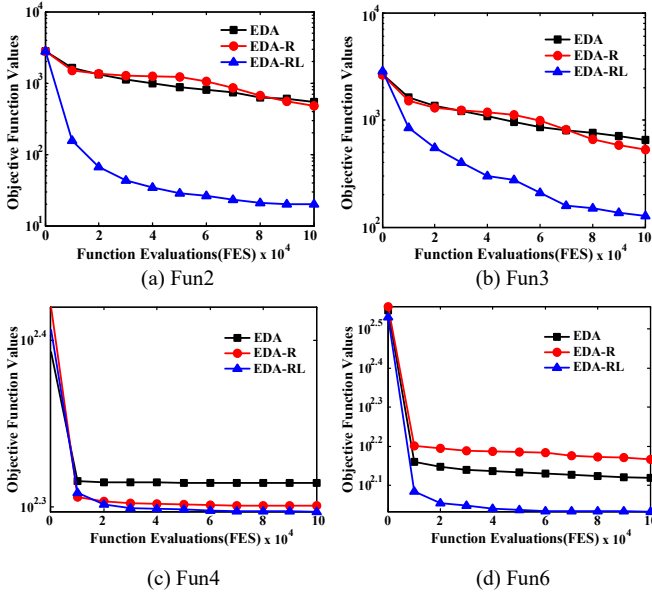


Fig. 10. The convergence curves of EDA, EDA-R and EDA-RL on four 10-D problems: (a) Fun2, (b) Fun3, (c) Fun4, and (d) Fun6.

temperature and carbon content are also closer to the expectation values.

TABLE IV shows the results of performance comparisons between EK and EDA-RL. From the comparison results, we can see that EDA-RL performs better than EK and that the calculated error results are consistent with the practical production requirements.

6) *Computational complexity analysis of EDA-RL*: Based on [35], some analysis of modeling, sampling, and improvement

strategies are conducted in this study. In $UMDA_C^G$, the computational complexity of modeling is $O(DA)$ in each generation, where $A = \lambda NP$, $0 < \lambda < 1$. The sampling process requires $O(DQ)$, where $Q = NP - A$. In EDA with a hybrid distribution model, the overall computational complexity requires $O(DA)$ in the modeling. The overall computational complexity of sampling requires $O(DY_1 + DY_2)$, where we note that $Y_1 = \rho(NP - A)$, $Y_2 = (1 - \rho)(NP - A)$, and $0 < \rho < 1$. Thus, the overall complexity of the sampling strategy has the same computational complexity as $UMDA_C^G$. For the resampling strategy, the modeling requires $O(DA)$, and the resampling requires $O(DW)$, where $A = \lambda NP$, $0 < \lambda < 1$, and $W = 2NP$. Thus, the computational complexity of resampling is $O(DNP)$. For the local improvement, the evolutionary search based on superior individuals requires $O(D\beta NP)$, and the information learning requires $O(D^2)$. Because $0 < \beta < 1$, the overall computational complexity of local improvement is between $O(D^2)$ and $O(DC)$, where we note that $0 < C < NP$. We conclude from the aforementioned comparisons and corresponding analysis that, the increase in computation cost is not significant. It is obvious that EDA-RL solves the operation optimization problem in BOF steelmaking efficiently.

C. Further Test of EDA-RL as a General Optimization Algorithm

To further verify the performance of EDA-RL, we carry out some numerical experiments described in this section. Some complex functions based on black-box optimization [49] are considered as benchmark problems. To ensure the fairness of comparisons, 51 independent runs are conducted for the black-box optimization benchmark problems, and the MaxFES is set to $10000 \times D$.

1) *Benchmark functions*: The benchmark problems belong to the class of real-parameter single objective minimum optimization problems without knowledge of the exact equations. These problems consist of several basic functions, and their landscapes may have contained multiple non-separable local optima. Furthermore, these black-box optimization functions have different properties as a result of variable shift and variable rotation operations (such as rotated high conditioned elliptic function, shifted and rotated Ackley function, shifted and rotated Schwefel function, etc.). As far as we know, a successful algorithm for the solution of these problems should not be influenced by shifting the position of the optimum and slightly changing the rotation matrix.

The selected benchmark problems are based on complex composition functions described in [49], which consist of unimodal functions, simple multimodal functions, and hybrid functions. These functions can match many data-driven operation optimization models in practical application problems. Fun1–Fun8 represent the composition functions from No. 21 to No. 28, respectively, in [49]. The search ranges of all the functions are set to $[-100, 100]$. More details on the numerical results can be found in the Appendix of this paper. The best result obtained for each benchmark function, as

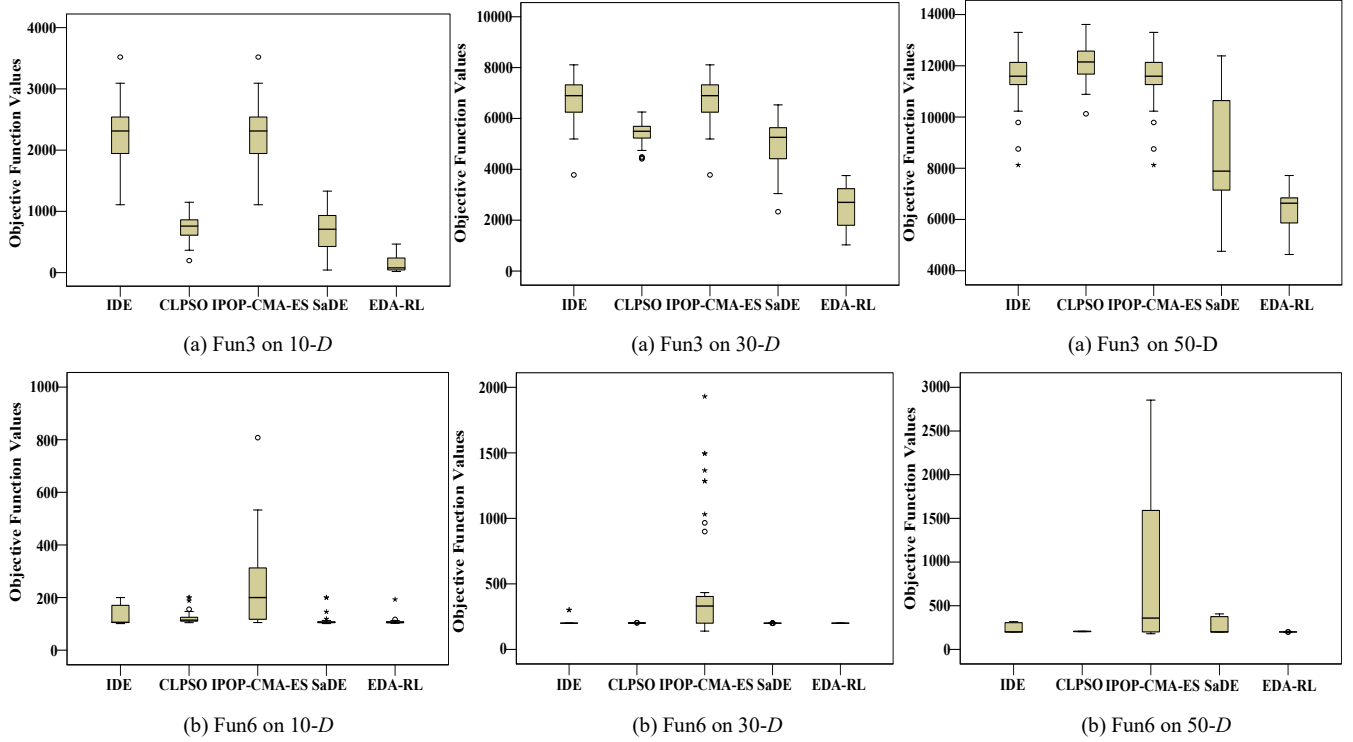


Fig. 11. Comparison results between EDA-RL and other algorithms on 10/30/50- D problems: (a) Fun3, and (b) Fun6.

indicated by the mean and standard deviation, is highlighted in bold font.

2) *Parameter analysis*: To further confirm the sensitivity of EDA-RL, parameter comparisons for some representative functions on 10/30/50-dimension problems are conducted. The results are shown in TABLE B-I and TABLE B-II in the Appendix. The final results are consistent with those obtained previously for the practical problem. The best results are also obtained at $\theta = 60\%$ and $\varepsilon = 10$. This suggests that the current parameter settings for EDA-RL are effective for solving different black-box optimization problems.

3) *Strategy comparisons*: The results obtained for different strategies are listed in TABLE B-III to TABLE B-V in the Appendix, respectively, where EDA-RL is compared to EDA and EDA-R on 10/30/50-dimension problems. To test the accuracy of each strategy, its solution for each instance is compared to the known optimal solution. Based on the calculated values of the mean error and standard deviation, the best results are shown in bold font. We can see from the results that the performance of EDA-RL is superior to that of EDA and EDA-R for most of the problems. The number of best results for each strategy is plotted in Fig. 8, which shows that EDA-RL has a prominent effect on 10/30-dimension problems.

To investigate the interaction between “R” and “L” in terms of the performance of EDA-RL for different problem structures, we conduct the analysis of different strategies. Based on the above results, we find that the “R” strategy is increasingly effective when the dimension increases. In dealing with low dimension problems, EDA-R exhibits no obvious effect. Although EDA converges faster than EDA-R in a few unimodal black-box optimization problems, the latter performs better in terms of its exploration ability for

multimodal problems. Moreover, with the “L” strategy, EDA-RL selects the superior individuals to yield the best individual in the current population, and employs operators with information learning. Thus, EDA-RL can take full advantage of population information in the later stage of the process to enhance the exploitation ability of the algorithm and can yield some better solutions than EDA and EDA-R.

4) *Comparisons with state-of-the-art algorithms*: To further demonstrate the competitiveness of our proposed algorithm, we compare EDA-RL to some state-of-the-art algorithms, using their reported results, such as differential evolution with an individual-dependent (IDE) [22], comprehensive learning particle swarm optimizer (CLPSO) [50], DE with an adaptive strategy (SaDE) [51], and restart covariance matrix adaptation evolution strategy with increasing population size (IPOP-CMA-ES) [52]. These state-of-the-art algorithms have been shown to be successful in solving black-box optimization problems, and have yielded better test results for some benchmark functions than some basic optimization algorithms such as DE, PSO, and CMA-ES.

Fig. 9 shows the number of best results obtained with EDA-RL and the state-of-the-art algorithms. Statistical results concerning significant differences are shown in TABLE B-VI–VIII in the Appendix. In comparing statistically significant differences, the two-sided Wilcoxon rank sum test [22] is conducted to evaluate performance differences at the 0.05 significance level. The symbol “+” indicates that the compared algorithm performs better than EDA-RL, the symbol “−” indicates that EDA-RL performs better, and the symbol “=” indicates no significant difference between the performance of EDA-RL and the compared algorithm.

As is evident from the aforementioned comparison results, it is notable that for 10/30/50-dimension problems, in most functions, EDA-RL is able to find solutions of the same or better quality, because “R” can identify multi-modal and different local optima for multiple variables effectively, and “L” can improve the local search ability to make convergence faster in the later search process. This implies that EDA-RL can address larger complex functions with large number of local optima more effectively. On the other hand, EDA-RL fails to perform the best for Fun2, Fun3, and Fun4, but it still remains competitive with other state-of-the-art algorithms. Therefore, as the results obtained show, our proposed algorithm can effectively solve some complex black-box optimization problems and exhibit a powerful generalization ability.

5) *Analysis of performance*: In order to investigate the convergence and stability of EDA-RL, we present the convergence curves for EDA, EDA-R, and EDA-RL for four 10-dimension problems in Fig. 10. The figures show that EDA-RL requires fewer evaluations to reach a steady state than the other strategies. As with the previous comparison results, EDA-RL obtains better objective function values than EDA and EDA-R. To graphically illustrate the best results obtained with each algorithm for the 51 independent runs, box plots of the minimum objective function values obtained by the different algorithms on 10/30/50- D problems for Fun3 and Fun6 are shown in Fig. 11. Each box illustrates lower quartile, median, and upper quartile values. From the box plots, we can see that EDA-RL performs more consistently than the other state-of-the-art algorithms and that EDA-RL performs better than the other algorithms in terms of the average minimum objective function value. As a whole, EDA-RL is competitive with the other algorithms for problems of different dimensions.

V. CONCLUSION

This paper has studied a control parameter optimization problem in the steelmaking process. Based on historical data, a black-box model was built to describe the endpoint control operation of the steelmaking process. This model was then used to formulate an optimization problem to set the control parameters for the required temperature and molten steel quality. To solve the problem, a new framework based on EDA with a resampling strategy and a local improvement (EDA-RL) was developed. In the EDA, we designed a hybrid distribution model to strengthen the diversity of population. A resampling strategy was proposed to enlarge the sampling region and to guide the new population to a more promising area. This new sampling model with resampling has good exploration ability. To enhance its exploitation ability, we incorporated local improvement, based on simplified gravitational search and information learning, to identify the current best individual. The EDA-RL was tested on practical operation data. The results illustrate that our proposed approach can optimize the control parameters of the steelmaking process to obtain the specified output accurately. To further test the performance of EDA-RL as a general optimization algorithm, an experiment was conducted on some complex black-box optimization benchmark problems. The results indicate that EDA-RL

outperforms other state-of-the-art algorithms across a wide range of problems.

In process industries, there are problems that involve optimizing the dynamic processes of operations instead of the final outputs. These problems can be considered as dynamic operation optimization problems. In further research, we will apply our proposed algorithm to solve dynamic operation optimization problems in practice.

REFERENCES

- [1] J. L. Ding, H. Modares, T. Y. Chai, and F. L. Lewis, “Data-based multiobjective plant-wide performance optimization of industrial processes under dynamic environments,” *IEEE Trans. Ind. Informat.*, vol. 12, no. 2, pp. 454–465, Apr. 2016.
- [2] Y. Zhang, Y. Yang, S. X. Ding, and L. Li, “Data-driven design and optimization of feedback control systems for industrial applications,” *IEEE Trans. Ind. Electron.*, vol. 61, no. 11, pp. 6409–6417, Nov. 2014.
- [3] R. H. Chi, Z. S. Hou, S. T. Jin, and B. Huang, “An improved data-driven point-to-point ILC using additional on-line control inputs with experimental verification,” *IEEE Trans. Syst., Man., Cyber. Syst.*, vol. 49, no. 4, pp. 687–696, Apr. 2019.
- [4] T. H. S. Li, M. C. Kao, and P. H. Kuo, “Recognition system for home-service-related sign language using entropy-based k-means algorithm and ABC-based HMM,” *IEEE Trans. Syst., Man., Cyber. Syst.*, vol. 46, no. 1, pp. 150–162, Jan. 2016.
- [5] X. Qin, S. Lysecky, and J. Sprinkle, “A data-driven linear approximation of HVAC utilization for predictive control and optimization,” *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 2, pp. 778–786, Mar. 2015.
- [6] H. X. Tian and Z. Z. Mao, “An ensemble ELM based on modified AdaBoost.RT algorithm for predicting the temperature of molten steel in ladle furnace,” *IEEE Trans. Autom. Sci. Eng.*, vol. 7, no. 1, pp. 73–80, Jan. 2010.
- [7] L.-F. Xu, W. Li, M. Zhang, S.-X. Xu, and J. Li, “A model of basic oxygen furnace (BOF) end-point prediction based on spectrum information of the furnace flame with support vector machine (SVM),” *Opt.-Int. J. Light Electron Opt.*, vol. 122, no. 7, pp. 594–598, Apr. 2011.
- [8] H. Liu, B. Wang, and X. Xiong, “Basic oxygen furnace steelmaking end-point prediction based on computer vision and general regression neural network,” *Opt.-Int. J. Light Electron Opt.*, vol. 125, no. 18, pp. 5241–5248, Sep. 2014.
- [9] Y. Shao, Q. Zhao, Y. R. Chen, Q. B. Zhang, and K. Wang, “Applying flame spectral analysis and multi-class classification algorithm on the BOS endpoint carbon content prediction,” *Opt.-Int. J. Light Electron Opt.*, vol. 126, no. 23, pp. 4539–4543, Dec. 2015.
- [10] F. He and L. Y. Zhang, “Prediction model of end-point phosphorus content in BOF steelmaking process based on PCA and BP neural network,” *J. Process Control*, vol. 66, pp. 51–58, Jun. 2018.
- [11] W. Birk, I. Arvanitidis, P. G. Jonsson, and A. Medvedev, “Physical modeling and control of dynamic foaming in an LD-converter process,” *IEEE Trans. Ind. Appl.*, vol. 37, no. 4, pp. 1067–1073, Jul. 2001.
- [12] C. Liu, X. Song, T. Xu, and L. Tang, “An operation optimization method based on improved EDA for BOF end-point control,” in *Proc. IEEE World Congr. Comput. Intell. (WCCI)*, Vancouver, BC, Canada, Jul. 2016, pp. 1077–1084.
- [13] B. Brunaud, I. E. Grossmann, “Perspectives in multilevel decision-making in the process industry,” *Front. Eng. Manag.*, vol. 4, no. 3, pp. 256–270, Jul. 2017.
- [14] S. L. Jiang, M. Liu, and J. H. Hao, “A two-phase soft optimization method for the uncertain scheduling problem in the steelmaking industry,” *IEEE Trans. Syst., Man., Cyber. Syst.*, vol. 47, no. 3, pp. 416–431, Mar. 2017.
- [15] S. P. Yu, T. Y. Chai, and Y. Tang, “An effective heuristic rescheduling method for steelmaking and continuous casting production process with multirefining modes,” *IEEE Trans. Syst., Man., Cyber. Syst.*, vol. 46, no. 12, pp. 1675–1688, Dec. 2016.
- [16] H. Stalford and G. Leitmann, “On representing a black box as a dynamical system,” *J. Math. Anal. Appl.*, vol. 38, no. 2, pp. 348–364, May 1972.
- [17] C. Liu, L. X. Tang, J. Y. Liu, and Z. H. Tang, “A dynamic analytics method based on multistage modeling for a BOF steelmaking process,” *IEEE Trans. Autom. Sci. Eng.*, vol. 16, no. 3, pp. 1097–1109, Jul. 2019.

- [18] C. L. P. Chen and Z. L. Liu, "Broad learning system: An effective and efficient incremental learning system without the need for deep architecture," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 1, pp. 10–24, Jan. 2018.
- [19] C. L. P. Chen, "A rapid supervised learning neural network for function interpolation and approximation," *IEEE Trans. Neural Netw.*, vol. 7, no. 5, pp. 1220–1230, Sep. 1996.
- [20] C. Liu, L. X. Tang, and J. Y. Liu, "Least squares support vector machine with self-organizing multiple kernel learning and sparsity," *Neurocomputing*, vol. 331, pp. 493–504, Feb. 2019.
- [21] A. Žilinskas and J. Calvin, "Bi-objective decision making in global optimization based on statistical models," *J. Glob. Optim.*, vol. 2, pp. 1–11, Feb. 2018.
- [22] L. X. Tang, Y. Dong, and J. Y. Liu, "Differential evolution with an individual-dependent mechanism," *IEEE Trans. Evol. Comput.*, vol. 19, no. 4, pp. 560–574, Aug. 2015.
- [23] W. Ewert, R. J. Marks, B. B. Thompson, and A. Yu, "Evolutionary inversion of swarm emergence using disjunctive combs control," *IEEE Trans. Syst., Man., Cyber. Syst.*, vol. 43, no. 5, pp. 1063–1076, Sep. 2013.
- [24] A. Che, P. Wu, F. Chu, and M. C. Zhou, "Improved quantum-inspired evolutionary algorithm for large-size lane reservation," *IEEE Trans. Syst., Man., Cyber. Syst.*, vol. 45, no. 12, pp. 1535–1548, Dec. 2015.
- [25] X. P. Wang, Z. M. Dong, and L. X. Tang, "Multiobjective differential evolution with personal archive and biased self-adaptive mutation selection," *IEEE Trans. Syst., Man., Cyber. Syst.*, in press, doi: 10.1109/TSMC.2018.2875043.
- [26] D. W. Gong, B. Xu, Y. Zhang, Y. N. Guo, and S. X. Yang, "A similarity-based cooperative co-evolutionary algorithm for dynamic interval multi-objective optimization problems," *IEEE Trans. Evol. Comput.*, to be published, doi: 10.1109/TEVC.2019.2912204.
- [27] D. W. Gong, J. Sun, and Z. Miao, "A set-based genetic algorithm for interval many-objective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 22, no. 1, pp. 47–60, Feb. 2018.
- [28] D. W. Gong, Y. P. Liu, and G. G. Yen, "A meta-objective approach for many-objective evolutionary optimization," *Evol. Comput.*, to be published, doi: 10.1162/evco_a_00243.
- [29] D. W. Gong, J. Sun, and X. F. Ji, "Evolutionary algorithms with preference polyhedron for interval multi-objective optimization problems," *Inf. Sci.*, vol. 233, pp. 141–161, Jun. 2013.
- [30] C. Echegoyen, A. Mendiburu, R. Santana, and J. A. Lozano, "Toward understanding EDAs based on Bayesian networks through a quantitative analysis," *IEEE Trans. Evol. Comput.*, vol. 16, no. 2, pp. 173–189, Apr. 2012.
- [31] A. Zhou, Q. Zhang, and Y. Jin, "Approximating the set of Pareto-optimal solutions in both the decision and objective spaces by an estimation of distribution algorithm," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 1167–1189, Oct. 2009.
- [32] Q. Yang, W. N. Chen, Y. Li, C. L. P. Chen, X. M. Xu, and J. Zhang, "Multimodal estimation of distribution algorithms," *IEEE Trans. Cybern.*, vol. 47, no. 3, pp. 636–650, Mar. 2017.
- [33] P. Yang, K. Tang, and X. Lu, "Improving estimation of distribution algorithm on multimodal problems by detecting promising areas," *IEEE Trans. Cybern.*, vol. 45, no. 8, pp. 1438–1449, Aug. 2015.
- [34] V. A. Shim, K. C. Tan, C. Y. Cheong, and J. Y. Chia, "Enhancing the scalability of multi-objective optimization via restricted Boltzmann machine-based estimation of distribution algorithm," *Inf. Sci.*, vol. 248, pp. 191–213, Nov. 2013.
- [35] W. Dong, T. Chen, P. Tiño, and X. Yao, "Scaling up estimation of distribution algorithms for continuous optimization," *IEEE Trans. Evol. Comput.*, vol. 17, no. 6, pp. 797–822, Dec. 2013.
- [36] A. Kabán, J. Bootkrajang, and R. J. Durrant, "Towards large scale continuous EDA: A random matrix theory perspective," in *Proc. Genet. Evol. Comput. (GECCO)*, Jul. 2013, pp. 383–390.
- [37] M. L. Sanyang, R. J. Durrant, A. Kabán, "How effective is Cauchy-EDA in high dimensions?" in *Proc. IEEE World Congr. Comput. Intell.*, Vancouver, Canada, Jul. 2016, pp. 3409–3416.
- [38] X. Yao and Y. Liu, "Fast evolution strategies," in *Evolutionary Programming*, vol. 1213 of Lect. Notes in Comput. Sci., Berlin, Germany: Springer, 1997, pp. 151–161.
- [39] P. Posik, "Comparison of cauchy EDA and BIPOP-CMA-ES algorithms on the BBOB noiseless testbed," in *Proc. Genet. Evol. Comput. (GECCO)*, Jul. 2010, pp. 1697–1702.
- [40] J. Zhong, J. Zhang, and Z. Fan, "MP-EDA: A robust estimation of distribution algorithm with multiple probabilistic models for global continuous optimization," in *Simu. Evol. Learn.*, Berlin, Germany: Springer, 2010, pp. 85–94.
- [41] A. Zhou, J. Sun, and Q. Zhang, "An estimation of distribution algorithm with cheap and expensive local search methods," *IEEE Trans. Evol. Comput.*, vol. 19, no. 6, pp. 807–822, Dec. 2015.
- [42] J. Sun, Q. Zhang, and E. P. K. Tsang, "DE/EDA: A new evolutionary algorithm for global optimization," *Inf. Sci.*, vol. 169, no. 3–4, pp. 249–262, Feb. 2005.
- [43] C. W. Ahn, J. An, and J.-C. Yoo, "Estimation of particle swarm distribution algorithms: Combining the benefits of PSO and EDAs," *Inf. Sci.*, vol. 192, pp. 109–119, Jun. 2012.
- [44] J. A. K. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Proc. Lett.*, vol. 9, no. 3, pp. 293–300, Jun. 1999.
- [45] R. Storn and K. V. Price, "Differential evolution: A simple and efficient adaptive scheme for global optimization over continuous spaces," *Int. Comput. Sci. Inst., Berkeley, CA, USA, Tech. Rep. TR-95-012*, 1995.
- [46] H. Mühlenbein, and G. Paaß, "From recombination of genes to the estimation of distribution. I. Binary parameters," in *Inter. Conf. Para. Prob. Solv. from Nature IV*, Berlin, Germany, Sep. 1996, pp. 178–187.
- [47] P. Larrañaga and J. Lozano, *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation*. Norwell, MA, USA: Kluwer, 2001.
- [48] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, "GSA: A gravitational search algorithm," *Inf. Sci.*, vol. 179, no. 13, pp. 2232–2248, Jun. 2009.
- [49] J. J. Liang, B. Y. Qu, P. N. Suganthan, and A. G. HernándezDíaz, "Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimization," *Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China, and Nanyang Technological University, Singapore, Technical Report 201212*, Jan. 2013. [Online]. Available: <http://www.ntu.edu.sg/home/epnsugan/>
- [50] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *IEEE Trans. Evol. Comput.*, vol. 10, no. 3, pp. 281–295, Jun. 2006.
- [51] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 398–417, Apr. 2009.
- [52] I. Loshchilov, "CMA-ES with Restarts for Solving CEC 2013 Benchmark Problems," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Cancún, México, Jun. 2013, pp. 369–376.



Lixin Tang (M'09–SM'14) received the B.E. degree in industrial automation, the M.E. degree in systems engineering and the Ph.D. degree in control theory and application from Northeastern University, Shenyang, China, in 1988, 1991, 1996, respectively.

He is currently a Cheung Kong Scholars Chair Professor, the Vice President of Northeastern University, the Director of the Institute of Industrial & Systems Engineering, Northeastern University. His research interests include industrial big data science, data analytics and machine learning, reinforcement learning and dynamic optimization, computational intelligent optimization, plant-wide production and logistics planning, production and logistics batching and scheduling and engineering applications in manufacturing (steel, petroleum-chemical, nonferrous), energy, resources industry and logistics systems. He has published papers in journals such as *Operations Research*, *IIE Transactions*, *Naval Research Logistics*, *European Journal of Operational Research*, *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION*, *IEEE TRANSACTIONS ON POWER SYSTEMS*, AND *IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY*.

Dr. Tang serves as an Associate Editor of *IIE Transactions*, *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION*, *IEEE TRANSACTIONS ON CYBERNETICS*, *Journal of Scheduling*, *International Journal of Production Research*, *Journal of the Operational Research Society*, in Editorial Board of *Annals of Operations Research*, and an Area Editor of the *Asia-Pacific Journal of Operational Research*.



Chang Liu (S'18) is currently pursuing the Ph.D. degree in systems engineering with the Institute of Industrial and Systems Engineering, Northeastern University, Shenyang, China.

His research interests include data analytics and optimization for smart industry, model predictive control, machine learning, computational intelligent optimization, and engineering applications in various industrial processes. He has published several papers in journals such as *IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING*, and *Neurocomputing*.

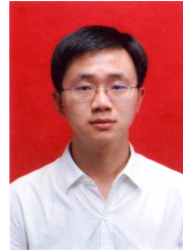


Jiyin Liu received the B.E. degree in industrial automation and the M.E. degree in systems engineering from Northeastern University, Shenyang, China, in 1982 and 1985, respectively, and the Ph.D. degree in manufacturing engineering and operations management from the University of Nottingham, Nottingham, U.K., in 1993.

He is currently a Professor of Operations Management with the School of Business and Economics, Loughborough University, Leicestershire, U.K. He is also a Cheung Kong Scholars Visiting Chair Professor at the Institute of Industrial & Systems Engineering, Northeastern University. His research interests include operations planning and

scheduling problems in production, logistics and supply chains, and mathematical modeling, optimization and heuristic methods. He has published papers in journals such as *European Journal of Operational Research*, *IEEE TRANSACTIONS*, *IIE Transactions*, *International Journal of Production Research*, *Naval Research Logistics*, *Operations Research*, and *Transportation Research*.

Xianpeng Wang (M'11) received the B.S. degree in Materials and Control Engineering from Shenyang University, Shenyang, China, and the Ph.D. degree in Systems Engineering from Northeastern University, Shenyang, China, in 2002 and 2007, respectively.



He is currently a professor with the Liaoning Engineering Laboratory of Data Analytics and Optimization for Smart Industry, and State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, China. His research interests include multi-objective optimization, machine learning, production scheduling, modeling and optimization in process industries based on data analytics, decision support systems, and process operation optimization. He has published more than 20 papers in international journals such as *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION*, *IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY*, *European Journal of Operational Research*, *Applied Soft Computing*, *Computers & Operations Research*, *IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING*, *Information Sciences*, and *Journal of the Operational Research Society*.

APPENDIX

A. Numerical Results on the Practical Problem

TABLE A-I

PARAMETER COMPARISONS FOR RESAMPLING ON THE PRACTICAL PROBLEM

Instance	$\theta = 20\%$ Mean/Std.	$\theta = 40\%$ Mean/Std.	$\theta = 60\%$ Mean/Std.	$\theta = 80\%$ Mean/Std.
1	1.46E-01 \pm 4.61E-03	1.45E-01 \pm 3.44E-03	1.44E-01 \pm 2.32E-03	1.43E-01 \pm 4.04E-03
2	3.55E-01 \pm 1.98E-02	3.59E-01 \pm 1.94E-02	3.59E-01 \pm 1.45E-02	3.59E-01 \pm 1.54E-02
3	2.62E-01 \pm 2.63E-02	2.64E-01 \pm 1.87E-02	2.57E-01 \pm 1.63E-02	2.58E-01 \pm 1.63E-02
4	2.13E-02 \pm 7.59E-03	2.15E-02 \pm 8.30E-03	1.89E-02 \pm 5.66E-03	2.40E-02 \pm 8.04E-03
5	1.80E-02 \pm 8.58E-03	2.09E-02 \pm 9.43E-03	1.69E-02 \pm 8.06E-03	1.75E-02 \pm 6.96E-03
6	8.50E-02 \pm 3.22E-03	8.34E-02 \pm 2.61E-03	8.29E-02 \pm 2.18E-03	8.32E-02 \pm 2.48E-03
7	1.48E-01 \pm 1.96E-03	1.48E-01 \pm 2.26E-03	1.47E-01 \pm 1.43E-03	1.48E-01 \pm 2.16E-03
8	1.26E-01 \pm 7.80E-03	1.25E-01 \pm 6.55E-03	1.25E-01 \pm 7.77E-03	1.28E-01 \pm 7.11E-03
9	8.67E-02 \pm 2.00E-02	7.54E-02 \pm 1.28E-02	8.00E-02 \pm 1.41E-02	7.55E-02 \pm 1.21E-02
10	9.69E-03 \pm 4.61E-03	9.43E-03 \pm 3.67E-03	8.28E-03 \pm 2.93E-03	8.77E-03 \pm 3.81E-03
11	5.55E-02 \pm 4.36E-03	5.37E-02 \pm 4.05E-03	5.27E-02 \pm 3.65E-03	5.22E-02 \pm 3.34E-03
12	1.02E-01 \pm 2.09E-03	1.01E-01 \pm 2.17E-03	1.00E-01 \pm 1.96E-03	1.00E-01 \pm 1.61E-03
13	9.69E-02 \pm 3.06E-03	9.59E-02 \pm 2.59E-03	9.51E-02 \pm 1.97E-03	9.53E-02 \pm 2.65E-03
14	1.51E-02 \pm 3.87E-03	1.49E-02 \pm 3.50E-03	1.24E-02 \pm 2.09E-03	1.33E-02 \pm 3.45E-03
15	6.47E-02 \pm 7.06E-03	6.17E-02 \pm 3.89E-03	6.07E-02 \pm 2.92E-03	6.01E-02 \pm 3.26E-03
16	3.63E-02 \pm 9.78E-03	3.23E-02 \pm 8.04E-03	2.85E-02 \pm 7.47E-03	2.92E-02 \pm 5.95E-03
17	1.61E-01 \pm 1.09E-02	1.54E-01 \pm 7.41E-03	1.55E-01 \pm 8.81E-03	1.52E-01 \pm 8.35E-03
18	6.10E-02 \pm 4.02E-03	5.89E-02 \pm 2.20E-03	5.85E-02 \pm 2.19E-03	5.88E-02 \pm 1.79E-03
19	2.20E-01 \pm 1.47E-02	2.14E-01 \pm 1.02E-02	2.12E-01 \pm 1.03E-02	2.14E-01 \pm 9.87E-03
20	3.43E-01 \pm 6.81E-02	3.45E-01 \pm 5.81E-02	3.12E-01 \pm 9.11E-02	2.94E-01 \pm 9.77E-02
No. of Best	1	2	13	6

TABLE A-II

PARAMETER COMPARISONS FOR LOCAL IMPROVEMENT ON THE PRACTICAL PROBLEM

Instance	$\varepsilon = 1$ Mean/Std.	$\varepsilon = 10$ Mean/Std.	$\varepsilon = 50$ Mean/Std.	$\varepsilon = 100$ Mean/Std.
1	1.43E-01 \pm 2.92E-03	1.44E-01 \pm 2.32E-03	1.44E-01 \pm 4.19E-03	1.44E-01 \pm 3.46E-03
2	3.58E-01 \pm 1.53E-02	3.59E-01 \pm 1.45E-02	3.59E-01 \pm 1.53E-02	3.56E-01 \pm 1.75E-02
3	2.54E-01 \pm 1.63E-02	2.57E-01 \pm 1.63E-02	2.58E-01 \pm 1.51E-02	2.59E-01 \pm 1.98E-02
4	2.11E-02 \pm 6.98E-03	1.89E-02 \pm 5.66E-03	1.91E-02 \pm 7.60E-03	2.25E-02 \pm 8.31E-03
5	1.75E-02 \pm 7.06E-03	1.69E-02 \pm 8.06E-03	1.69E-02 \pm 6.18E-03	1.73E-02 \pm 7.00E-03
6	8.28E-02 \pm 2.11E-03	8.29E-02 \pm 2.18E-03	8.27E-02 \pm 1.86E-03	8.30E-02 \pm 2.44E-03
7	1.48E-01 \pm 2.19E-03	1.47E-01 \pm 1.43E-03	1.47E-01 \pm 1.96E-03	1.47E-01 \pm 2.30E-03
8	1.27E-01 \pm 5.53E-03	1.25E-01 \pm 7.77E-03	1.29E-01 \pm 4.61E-03	1.27E-01 \pm 6.65E-03
9	7.31E-02 \pm 1.13E-02	8.00E-02 \pm 1.41E-02	7.66E-02 \pm 1.62E-02	7.66E-02 \pm 1.30E-02
10	8.50E-03 \pm 4.00E-03	8.28E-03 \pm 2.93E-03	7.55E-03 \pm 3.58E-03	9.21E-03 \pm 3.07E-03
11	5.18E-02 \pm 4.85E-03	5.27E-02 \pm 3.65E-03	5.16E-02 \pm 4.43E-03	5.14E-02 \pm 4.43E-03
12	1.01E-01 \pm 1.76E-03	1.00E-01 \pm 1.96E-03	1.01E-01 \pm 1.74E-03	9.99E-02 \pm 1.74E-03
13	9.55E-02 \pm 2.16E-03	9.51E-02 \pm 1.97E-03	9.57E-02 \pm 2.94E-03	9.45E-02 \pm 1.71E-03
14	1.27E-02 \pm 2.51E-03	1.24E-02 \pm 2.09E-03	1.28E-02 \pm 1.96E-03	1.26E-02 \pm 2.11E-03
15	6.14E-02 \pm 3.84E-03	6.07E-02 \pm 2.92E-03	6.33E-02 \pm 5.14E-03	6.15E-02 \pm 5.60E-03
16	2.93E-02 \pm 6.80E-03	2.85E-02 \pm 7.47E-03	3.16E-02 \pm 7.75E-03	2.96E-02 \pm 7.01E-03
17	1.57E-01 \pm 1.01E-02	1.55E-01 \pm 8.81E-03	1.56E-01 \pm 8.73E-03	1.54E-01 \pm 8.52E-03
18	5.93E-02 \pm 2.40E-03	5.85E-02 \pm 2.19E-03	5.86E-02 \pm 2.03E-03	5.95E-02 \pm 2.42E-03
19	2.13E-01 \pm 1.10E-02	2.12E-01 \pm 1.03E-02	2.12E-01 \pm 7.25E-03	2.13E-01 \pm 1.07E-02
20	3.14E-01 \pm 8.71E-02	3.12E-01 \pm 9.11E-02	3.31E-01 \pm 7.48E-02	3.17E-01 \pm 8.20E-02
No. of Best	3	11	4	6

TABLE A-III
COMPARISON RESULTS OF MINIMUM OBJECTIVE FUNCTION VALUES

Instance	EDA	EDA-R	EDA-RL
1	1.47E-01 \pm 4.66E-03	1.45E-01 \pm 3.95E-03	1.44E-01 \pm 2.32E-03
2	3.69E-01 \pm 1.99E-02	3.55E-01 \pm 1.82E-02	3.59E-01 \pm 1.45E-02
3	2.62E-01 \pm 2.16E-02	2.60E-01 \pm 1.57E-02	2.57E-01 \pm 1.63E-02
4	2.79E-02 \pm 8.99E-03	2.59E-02 \pm 9.56E-03	1.89E-02 \pm 5.66E-03
5	2.58E-02 \pm 1.26E-02	1.92E-02 \pm 7.99E-03	1.69E-02 \pm 8.06E-03
6	8.61E-02 \pm 3.53E-03	8.46E-02 \pm 2.65E-03	8.29E-02 \pm 2.18E-03
7	1.50E-01 \pm 2.81E-03	1.49E-01 \pm 2.97E-03	1.47E-01 \pm 1.43E-03
8	1.21E-01 \pm 8.51E-03	1.24E-01 \pm 7.94E-03	1.25E-01 \pm 7.77E-03
9	8.09E-02 \pm 2.00E-02	8.22E-02 \pm 1.49E-02	8.00E-02 \pm 1.41E-02
10	1.11E-02 \pm 4.17E-03	8.76E-03 \pm 3.33E-03	8.28E-03 \pm 2.93E-03
11	5.34E-02 \pm 5.36E-03	5.48E-02 \pm 3.53E-03	5.27E-02 \pm 3.65E-03
12	1.03E-01 \pm 3.64E-03	1.02E-01 \pm 2.07E-03	1.00E-01 \pm 1.96E-03
13	9.57E-02 \pm 2.20E-03	9.49E-02 \pm 2.09E-03	9.51E-02 \pm 1.97E-03
14	1.53E-02 \pm 3.80E-03	1.32E-02 \pm 2.07E-03	1.24E-02 \pm 2.09E-03
15	6.54E-02 \pm 7.01E-03	6.19E-02 \pm 3.34E-03	6.07E-02 \pm 2.92E-03
16	3.52E-02 \pm 1.08E-02	3.03E-02 \pm 6.37E-03	2.85E-02 \pm 7.47E-03
17	1.62E-01 \pm 1.31E-02	1.60E-01 \pm 9.79E-03	1.55E-01 \pm 8.81E-03
18	6.02E-02 \pm 2.79E-03	5.88E-02 \pm 2.42E-03	5.85E-02 \pm 2.19E-03
19	2.20E-01 \pm 9.84E-03	2.13E-01 \pm 9.65E-03	2.12E-01 \pm 1.03E-02
20	3.51E-01 \pm 6.52E-02	3.46E-01 \pm 5.76E-02	3.12E-01 \pm 9.11E-02
No. of Best	1	2	17

TABLE A-IV
COMPARISON RESULTS FOR THE PRACTICAL PROBLEM

Instance	T (°C)			C (%)		
	Expectation Value	Absolute Error		Expectation Value	Absolute Error	
		EK	EDA-RL		EK	EDA-RL
1	1.644E+03	1.50E+01	2.73E+00	2.30E-02	6.00E-03	4.40E-05
2	1.641E+03	7.00E+00	4.11E+00	3.50E-02	1.60E-02	5.44E-03
3	1.650E+03	1.00E+01	3.41E-01	3.30E-02	1.20E-02	8.58E-03
4	1.645E+03	7.00E+00	2.54E-01	2.90E-02	5.00E-03	1.80E-05
5	1.648E+03	1.10E+01	4.87E-02	2.50E-02	2.00E-03	6.90E-05
6	1.651E+03	9.00E+00	2.40E-01	2.80E-02	4.00E-03	3.14E-03
7	1.655E+03	2.70E+01	2.23E-02	2.10E-02	9.00E-03	6.02E-03
8	1.658E+03	6.00E+00	1.96E+00	2.70E-02	5.00E-03	1.53E-03
9	1.656E+03	1.80E+01	1.02E-01	3.10E-02	1.00E-02	2.53E-03
10	1.653E+03	1.90E+01	6.22E-03	2.60E-02	6.00E-03	6.50E-05
11	1.655E+03	1.00E+01	5.97E-01	2.50E-02	2.00E-03	1.71E-03
12	1.654E+03	1.00E+00	1.18E-02	3.10E-02	8.00E-03	3.86E-03
13	1.662E+03	1.20E+01	1.83E+00	2.60E-02	2.00E-03	3.00E-05
14	1.664E+03	3.00E+00	1.65E-01	2.60E-02	8.00E-03	2.70E-05
15	1.665E+03	1.40E+01	1.09E+00	2.70E-02	6.00E-03	4.00E-06
16	1.664E+03	1.90E+01	3.55E-01	2.30E-02	5.00E-03	2.12E-04
17	1.662E+03	6.00E+00	1.98E+00	3.00E-02	7.00E-03	2.07E-03
18	1.665E+03	1.20E+01	1.09E+00	2.70E-02	1.00E-03	1.50E-05
19	1.676E+03	8.00E+00	8.80E-01	3.20E-02	7.00E-03	3.79E-03
20	1.671E+03	2.40E+01	3.22E+00	2.30E-02	5.00E-03	2.95E-03

B. Numerical Results on Benchmark Problems

TABLE B-I

PARAMETER COMPARISONS FOR RESAMPLING ON BENCHMARK PROBLEMS

Fun.	D	$\theta = 20\%$ Mean/std.	$\theta = 40\%$ Mean/std.	$\theta = 60\%$ Mean/std.	$\theta = 80\%$ Mean/std.
Fun2	10	4.10E+01 \pm 3.69E+01	3.85E+01 \pm 4.58E+01	2.23E+01 \pm 1.91E+01	2.70E+01 \pm 2.51E+01
	30	2.52E+02 \pm 1.16E+02	2.49E+02 \pm 1.08E+02	2.16E+02 \pm 5.63E+01	2.50E+02 \pm 9.29E+01
	50	4.09E+02 \pm 2.32E+02	3.99E+02 \pm 2.57E+02	3.67E+02 \pm 1.96E+02	4.17E+02 \pm 1.95E+02
Fun3	10	3.42E+02 \pm 2.36E+02	2.34E+02 \pm 2.13E+02	2.34E+02 \pm 1.86E+02	2.49E+02 \pm 1.85E+02
	30	3.70E+03 \pm 7.12E+02	3.32E+03 \pm 6.12E+02	3.06E+03 \pm 5.06E+02	2.93E+03 \pm 4.04E+02
	50	7.65E+03 \pm 9.64E+02	6.98E+03 \pm 8.73E+02	6.37E+03 \pm 6.43E+02	6.58E+03 \pm 6.92E+02
Fun8	10	3.06E+02 \pm 8.78E+01	3.46E+02 \pm 1.28E+02	2.45E+02 \pm 9.01E+01	3.23E+02 \pm 8.80E+01
	30	2.96E+02 \pm 2.80E+01	3.42E+02 \pm 2.10E+02	2.84E+02 \pm 5.43E+01	3.00E+02 \pm 0.00E+00
	50	4.60E+02 \pm 4.25E+02	5.19E+02 \pm 5.95E+02	4.00E+02 \pm 0.00E+00	5.19E+02 \pm 5.95E+02
No. of Best		0	1	8	1

TABLE B-II

PARAMETER COMPARISONS FOR LOCAL IMPROVEMENT ON BENCHMARK PROBLEMS

Fun.	D	$\varepsilon = 1$ Mean/std.	$\varepsilon = 10$ Mean/std.	$\varepsilon = 50$ Mean/std.	$\varepsilon = 100$ Mean/std.
Fun2	10	3.52E+01 \pm 3.95E+01	2.23E+01 \pm 1.91E+01	5.37E+01 \pm 5.54E+01	4.65E+01 \pm 5.18E+01
	30	2.48E+02 \pm 9.76E+01	2.16E+02 \pm 5.63E+01	2.90E+02 \pm 1.02E+02	3.18E+02 \pm 1.33E+02
	50	3.07E+02 \pm 1.85E+02	3.67E+02 \pm 1.96E+02	5.87E+02 \pm 2.58E+02	6.24E+02 \pm 3.21E+02
Fun3	10	2.47E+02 \pm 2.04E+02	2.34E+02 \pm 1.86E+02	4.03E+02 \pm 2.44E+02	4.39E+02 \pm 2.56E+02
	30	2.81E+03 \pm 8.07E+02	3.06E+03 \pm 5.06E+02	3.36E+03 \pm 1.03E+03	3.08E+03 \pm 1.15E+03
	50	6.48E+03 \pm 7.09E+02	6.37E+03 \pm 6.43E+02	8.03E+03 \pm 9.56E+02	8.85E+03 \pm 1.23E+03
Fun8	10	3.52E+02 \pm 1.24E+02	2.45E+02 \pm 9.01E+01	3.61E+02 \pm 1.38E+02	3.78E+02 \pm 1.50E+02
	30	5.70E+02 \pm 6.56E+02	2.84E+02 \pm 5.43E+01	6.11E+02 \pm 7.66E+02	8.19E+02 \pm 8.21E+02
	50	6.96E+02 \pm 9.07E+02	4.00E+02 \pm 0.00E+00	5.21E+02 \pm 6.03E+02	5.81E+02 \pm 7.30E+02
No. of Best		2	7	0	0

TABLE B-III

EXPERIMENTAL RESULTS OF STRATEGY COMPARISONS ON 10- D BENCHMARK PROBLEMS

10- D	EDA	EDA-R	EDA-RL
Fun1	4.00E+02 \pm 0.00E+00	4.00E+02 \pm 0.00E+00	3.88E+02 \pm 4.32E+01
Fun2	5.46E+02 \pm 3.08E+02	4.74E+02 \pm 3.31E+02	2.00E+01 \pm 1.47E+01
Fun3	6.51E+02 \pm 3.42E+02	5.29E+02 \pm 2.89E+02	1.28E+02 \pm 1.17E+02
Fun4	2.06E+02 \pm 4.49E+00	2.00E+02 \pm 4.42E-02	1.98E+02 \pm 9.09E+00
Fun5	2.01E+02 \pm 1.98E+00	2.00E+02 \pm 1.59E-01	1.98E+02 \pm 9.92E+00
Fun6	1.32E+02 \pm 3.26E+01	1.47E+02 \pm 3.72E+01	1.08E+02 \pm 1.24E+01
Fun7	3.02E+02 \pm 4.31E+00	3.00E+02 \pm 2.31E-01	3.00E+02 \pm 6.09E-01
Fun8	3.00E+02 \pm 0.00E+00	3.00E+02 \pm 0.00E+00	2.45E+02 \pm 9.01E+01
No. of Best	0	1	8

TABLE B-IV

EXPERIMENTAL RESULTS OF STRATEGY COMPARISONS ON 30-D BENCHMARK PROBLEMS

30-D	EDA	EDA-R	EDA-RL
Fun1	3.45E+02 ± 7.73E+01	3.30E+02 ± 7.67E+01	2.52E+02 ± 6.04E+01
Fun2	5.37E+03 ± 1.41E+03	4.95E+03 ± 1.75E+03	2.01E+02 ± 6.06E+01
Fun3	6.59E+03 ± 3.59E+02	6.25E+03 ± 1.21E+03	2.55E+03 ± 8.10E+02
Fun4	2.18E+02 ± 5.92E+00	2.05E+02 ± 3.23E+00	2.10E+02 ± 4.47E+00
Fun5	2.65E+02 ± 7.83E+00	2.44E+02 ± 1.86E+01	2.10E+02 ± 1.65E+01
Fun6	2.00E+02 ± 8.27E-05	2.00E+02 ± 1.47E-04	2.00E+02 ± 1.79E-04
Fun7	4.50E+02 ± 5.98E+01	3.40E+02 ± 1.83E+01	3.91E+02 ± 4.79E+01
Fun8	3.00E+02 ± 0.00E+00	3.00E+02 ± 0.00E+00	2.88E+02 ± 4.75E+01
No. of Best	1	3	6

TABLE B-V

EXPERIMENTAL RESULTS OF STRATEGY COMPARISONS ON 50-D BENCHMARK PROBLEMS

50-D	EDA	EDA-R	EDA-RL
Fun1	8.26E+02 ± 3.86E+02	7.34E+02 ± 4.27E+02	4.97E+02 ± 4.14E+02
Fun2	6.78E+03 ± 5.09E+03	6.21E+03 ± 5.28E+03	2.87E+02 ± 1.50E+02
Fun3	1.29E+04 ± 4.58E+02	1.25E+04 ± 1.59E+03	6.41E+03 ± 6.96E+02
Fun4	2.47E+02 ± 9.71E+00	2.12E+02 ± 4.42E+00	2.20E+02 ± 6.67E+00
Fun5	3.26E+02 ± 9.00E+00	2.99E+02 ± 7.78E+00	3.02E+02 ± 7.99E+00
Fun6	2.00E+02 ± 1.82E-04	2.00E+02 ± 1.71E-04	2.00E+02 ± 2.66E-05
Fun7	8.92E+02 ± 1.20E+02	4.13E+02 ± 4.94E+01	5.47E+02 ± 1.14E+02
Fun8	4.00E+02 ± 0.00E+00	4.00E+02 ± 0.00E+00	4.00E+02 ± 0.00E+00
No. of Best	2	5	5

TABLE B-VI

COMPARISON RESULTS BETWEEN EDA-RL AND STATE-OF-THE-ART ALGORITHMS ON 10-D BENCHMARK PROBLEMS

10-D	IDE		CLPSO		IPOP-CMA-ES		SaDE		EDA-RL
Fun1	3.96E+02	=	2.88E+02	+	3.75E+02	=	4.00E+02	=	3.88E+02
	2.80E+01		1.21E+02		7.17E+01		0.00E+00		4.32E+01
Fun2	9.15E+00	=	4.46E+01	—	7.33E+01	—	2.65E+01	=	2.00E+01
	5.82E+00		5.04E+01		4.94E+01		2.63E+01		1.47E+01
Fun3	4.03E+02	—	7.42E+02	—	8.62E+01	+	6.74E+02	—	1.28E+02
	1.87E+02		1.94E+02		6.61E+01		3.19E+02		1.17E+02
Fun4	1.98E+02	=	1.42E+02	+	2.09E+02	—	1.94E+02	=	1.98E+02
	1.34E+01		3.46E+01		7.02E+00		2.17E+01		9.09E+00
Fun5	2.00E+02	=	1.92E+02	=	2.06E+02	—	1.99E+02	=	1.98E+02
	2.02E-05		3.19E+01		6.71E+00		1.12E+01		9.92E+00
Fun6	1.20E+02	—	1.22E+02	—	2.04E+02	—	1.11E+02	=	1.08E+02
	3.48E+01		2.16E+01		1.51E+01		1.93E+01		1.24E+01
Fun7	3.02E+02	=	3.25E+02	—	4.55E+02	—	3.04E+02	=	3.00E+02
	1.40E+01		3.53E+01		7.92E+01		1.96E+01		6.09E-01
Fun8	2.88E+02	—	2.50E+02	=	3.00E+02	—	2.92E+02	—	2.45E+02
	4.75E+01		8.60E+01		0.00E+00		3.92E+01		9.01E+01
No. of Best	1		3		1		0		3
—		3		4		6		2	
=		5		2		1		6	
+		0		2		1		0	

TABLE B-VII

COMPARISON RESULTS BETWEEN EDA-RL AND STATE-OF-THE-ART ALGORITHMS ON 30-D BENCHMARK PROBLEMS

30-D	IDE		CLPSO		IPOP-CMA-ES		SaDE		EDA-RL
Fun1	3.17E+02 6.01E+01	—	2.87E+02 2.78E+01	—	2.55E+02 5.03E+01	=	3.26E+02 8.06E+01	—	2.52E+02 6.04E+01
Fun2	1.21E+02 4.39E+00	+	1.43E+02 2.69E+01	+	5.02E+02 3.09E+02	—	1.14E+02 1.42E+01	+	2.01E+02 6.06E+01
Fun3	3.28E+03 3.80E+02	—	5.44E+03 4.14E+02	—	5.76E+02 3.50E+02	+	5.03E+03 9.93E+02	—	2.55E+03 8.10E+02
Fun4	2.00E+02 3.60E-01	+	2.72E+02 6.57E+00	—	2.86E+02 3.02E+01	—	2.26E+02 7.25E+00	—	2.10E+02 4.47E+00
Fun5	2.14E+02 2.09E+01	—	2.96E+02 5.43E+00	—	2.87E+02 2.85E+01	—	2.66E+02 1.23E+01	—	2.10E+02 1.65E+01
Fun6	2.00E+02 6.39E-03	=	2.02E+02 6.19E-01	—	3.15E+02 8.14E+01	—	2.00E+02 8.77E-03	—	2.00E+02 1.79E-04
Fun7	3.06E+02 5.37E+00	+	7.77E+02 3.14E+02	—	1.14E+03 2.90E+02	—	6.18E+02 7.16E+01	—	3.91E+02 4.79E+01
Fun8	3.00E+02 0.00E+00	=	3.00E+02 3.10E-05	=	3.00E+02 0.00E+00	=	3.00E+02 0.00E+00	=	2.88E+02 4.75E+01
No. of Best	3		0		1		2		4
—	3		6		5		6		
=	2		1		2		1		
+	3		1		1		1		

TABLE B-VIII

COMPARISON RESULTS BETWEEN EDA-RL AND STATE-OF-THE-ART ALGORITHMS ON 50-D BENCHMARK PROBLEMS

50-D	IDE		CLPSO		IPOP-CMA-ES		SaDE		EDA-RL
Fun1	7.32E+03 3.82E+02	—	5.31E+02 2.39E+02	=	5.17E+02 4.08E+02	=	9.18E+02 2.96E+02	—	4.97E+02 4.14E+02
Fun2	6.88E+01 2.03E+01	+	3.68E+02 1.58E+02	—	1.83E+03 2.86E+03	—	2.21E+01 4.87E+00	+	2.87E+02 1.50E+02
Fun3	7.32E+03 6.92E+02	—	1.21E+04 6.60E+02	—	2.99E+03 4.19E+03	+	8.75E+03 2.15E+03	—	6.41E+03 6.96E+02
Fun4	2.02E+02 1.14E+00	+	3.51E+02 7.23E+00	—	3.75E+02 3.34E+01	—	2.78E+02 9.38E+00	—	2.20E+02 6.67E+00
Fun5	3.03E+02 1.09E+01	=	3.92E+02 6.74E+00	—	3.74E+02 3.35E+01	—	3.44E+02 1.13E+01	—	3.02E+02 7.99E+00
Fun6	2.23E+02 4.46E+01	—	2.06E+02 9.17E-01	—	3.82E+02 1.29E+02	—	2.61E+02 8.73E+01	—	2.00E+02 2.66E-05
Fun7	3.58E+02 3.30E+01	+	1.50E+03 5.59E+02	—	1.94E+03 4.55E+02	—	1.17E+03 9.89E+01	—	5.47E+02 1.14E+02
Fun8	4.00E+02 0.00E+00	=	4.00E+02 1.63E-02	—	1.03E+03 1.22E+03	—	5.35E+02 6.75E+02	—	4.00E+02 0.00E+00
No. of Best	3		1		1		1		4
—	3		7		6		7		
=	2		1		1		0		
+	3		0		1		1		