

Enhancing the Performance of JADE Using Two-phase Parameter Control Scheme and Its Application

Qin-Qin Fan^{1,2} Yi-Lian Zhang³ Xue-Feng Yan² Zhi-Huan Wang¹

¹Logistics Research Center, Shanghai Maritime University, Shanghai 201306, China

²Key Laboratory of Advanced Control and Optimization for Chemical Processes of Ministry of Education,
East China University of Science and Technology, Shanghai 200237, China

³Key Laboratory of Marine Technology and Control Engineering Ministry of Communications,
Shanghai Maritime University, Shanghai 201306, China

Abstract: The search efficiency of differential evolution (DE) algorithm is greatly impacted by its control parameters. Although many adaptation/self-adaptation techniques can automatically find suitable control parameters for the DE, most techniques are based on population information which may be misleading in solving complex optimization problems. Therefore, a self-adaptive DE (i.e., JADE) using two-phase parameter control scheme (TPC-JADE) is proposed to enhance the performance of DE in the current study. In the TPC-JADE, an adaptation technique is utilized to generate the control parameters in the early population evolution, and a well-known empirical guideline is used to update the control parameters in the later evolution stages. The TPC-JADE is compared with four state-of-the-art DE variants on two famous test suites (i.e., IEEE CEC2005 and IEEE CEC2015). Results indicate that the overall performance of the TPC-JADE is better than that of the other compared algorithms. In addition, the proposed algorithm is utilized to obtain optimal nutrient and inducer feeding for the Lee-Ramirez bioreactor. Experimental results show that the TPC-JADE can perform well on an actual dynamic optimization problem.

Keywords: Differential evolution (DE) algorithm, evolutionary computation, dynamic optimization, control parameter adaptation, chemical processes.

1 Introduction

Differential evolution (DE) algorithm, which was introduced by Storn and Price^[1], is a simple yet competitive meta-heuristic algorithm. Although DE has been widely utilized to deal with a large number of benchmark test functions and industrial application problems, the search behavior of DE is mainly determined by three main parameters (i.e., mutation control parameter F , crossover control parameter CR , and population size NP) and two strategies (i.e., mutation and crossover)^[2, 3]. To improve DE performance, researchers have proposed different empirical guidelines to set control parameters in the previous studies^[4–6]. However, the above guidelines lack sufficient experimental justifications^[7] since they are achieved through some particular experiments.

Subsequently, a large number of researchers focus on the adaptation or self-adaptation of control parameters and strategies to enhance the search capability of DE and reduce manual tuning. For example, Liu and Lampinen^[8] proposed a fuzzy adaptive DE, in which fuzzy logic con-

trollers are utilized to produce F and CR . A DE with self-adaptive parameter control (jDE) is presented by Brest et al.^[9]. In the jDE, F and CR are generated by a normal distribution function. Moreover, each individual in the population has its own F and CR . In [7], a self-adaptive DE (SaDE) is introduced. In the SaDE, suitable mutation strategy and control parameters can be automatically achieved during the run. Zhang and Sanderson^[10] introduced a novel DE (JADE) which uses an improved DE/current-to-best/1 mutation strategy and adjusts control parameters in an adaptive fashion. In [11], an ensemble of mutation strategies and control parameters with DE algorithm (EPSDE) is proposed. In the EPSDE, some combinations of mutation strategy and control parameters are employed. Wang et al.^[12] presented a composite DE (CoDE) wherein three fixed parameter combinations were combined with three selected mutation strategies randomly. In [13], a novel DE (CoBiDE) is proposed to improve the performance of DE, in which the covariance matrix learning and the bimodal distribution parameter setting are used. Fan and Yan^[14] introduced a DE with self-adaptive mutation strategy and control parameters (SSCPDE), in which suitable control parameters and mutation strategy can be simultaneously achieved in different stages of the evolution. Recently, Fan and Yan^[15] proposed a self-adaptive DE with zoning

Research Article

Manuscript received May 5, 2017; accepted February 9, 2018;
published online May 4, 2018

Recommended by Associate Editor Matjaz Gams

© Institute of Automation, Chinese Academy of Sciences and
Springer-Verlag GmbH Germany, part of Springer Nature 2018

evolution of control parameters and adaptive mutation strategy (ZEPDE), in which F and CR can be automatically generated in their own zones and the suitable mutation strategy can be self-adaptively obtained in different phases of the evolution. In [16], a DE with dynamic parameters selection (DE-DPS) is proposed, in which the best performing control parameter combination can be automatically found in different evolution stages.

Besides the above studies, many methods have also been introduced to improve the search performance of the DE. For instance, in [17], an opposition-based learning is utilized to enhance the global search capability of the DE, the experimental results demonstrate that the approach is effective. Gong et al. [18] introduced a crossover rate repair method to enhance the performance of other self-adaptive DE algorithms. Recently, Gong et al. [19] introduced an adaptive ranking mutation operator (ARMOR) for the DE algorithm, which is used to deal with the constrained optimization problems introduced in CEC2006 and CEC2010. Yang et al. [20] introduced an auto-enhanced population diversity (AEPD) that is employed to improve the population diversity when the population is stagnant and premature. Fan et al. [21] introduced an auto-selection mechanism (ASM) to automatically choose a suitable algorithm from an algorithm pool. The results indicate that the ASM can improve the algorithm robustness. In [22], an eigenvector-based crossover strategy is used to improve the search ability of the DE. For more related studies, interested readers can refer to [23–31].

Although various adaptation techniques can improve the performance of the DE effectively, most of them are based on the population information. In most cases, this information may be deceptive or misleading in complex optimization environment [32], thus a well-known empirical guideline is incorporated into the JADE (called TPC-JADE) to improve the search efficiency of DE in the current study. In the TPC-JADE, the parameter adaptation technique used in the JADE is employed to search the promising region in the early evolution stages, while the empirical guideline is utilized to enhance the search efficiency of the JADE in the later evolution stages. The main reason is that the fitness landscape of the current population may be not very complex in the later phases of the search. The performance of the proposed algorithm is compared with that of four DE variants on two widely used test suites, i.e., IEEE CEC2005 [33] and IEEE CEC2015 [34]. The experimental results reveal that the TPC-JADE is competitive among all competitors.

Several problems in chemical/biochemical processes can be described as a set of nonlinear differential equations, which are named as dynamic optimization problems. To solve these dynamic optimization problems, three main approaches [35, 36], which include dynamic programming (DP) methods, indirect optimization methods, and direct optimization methods, have been introduced.

For the DP, it is based on Bellman's optimality conditions [37] and is a promising approach to solve dynamic optimization problems. However, DP is not always available for solving high-dimensional dynamic optimization problems. Subsequently, Luus [38] proposed an iterative dynamic programming (IDP) to overcome the defects of the DP, in which the coarse grid points and search region reduction strategy are utilized. The results indicate that the IDP is able to improve the solution accuracy and computation efficiency when compared with the DP.

In indirect methods, a dynamic optimization problem is converted into a boundary value problem (BVP), which can be addressed by some methods [36, 39, 40]. In fact, these approaches are very effective, but they are difficult to be used in solving dynamic optimization problems due to the active inequality constraints [41].

For direct methods, a dynamic optimization problem can be usually transcribed into a nonlinear programming (NLP) problem and is addressed by parameterization, which can be classified into two different techniques [42], namely, control vector parameterization (CVP) and complete parameterization. In the CVP, only control variables are discretized, whereas complete parameterization discretizes both control variable and state variable. In the current study, the CVP and the TPC-JADE are used to solve actual dynamic optimization problems. The results show that the TPC-JADE is competitive when compared with the other approaches.

2 Differential evolution

A minimization problem can be expressed as follows:

$$f(\mathbf{x}^*) = \min_{\mathbf{x}_i \in \Omega} f(\mathbf{x}_i), \mathbf{x}_i \in \mathbf{P}_0 = \prod_{j=1}^D [L_j, U_j] \quad (1)$$

where f denotes the objective function; $\mathbf{x}_i = [x_{i,1} \cdots x_{i,D}]$ and \mathbf{x}^* are the D -dimensional vector and the global optimum solution of problem, respectively; L_j and U_j ($j = 1, \dots, D$) are the lower and upper bounds of the j -th variable of \mathbf{x}_i respectively; and \mathbf{P}_0 is the feasible search space.

DE is a population-based stochastic optimization approach. $\mathbf{x}_i^G = [x_{i,1}^G \cdots x_{i,D}^G]$ and $\mathbf{X}^G = \{\mathbf{x}_1^G, \dots, \mathbf{x}_{NP}^G\}$ denote the i -th target vector and the population at the G -th generation, respectively.

The procedure of executing DE is described as follows [43]:

1) Initialization: Determine F , CR , NP , and maximum number of generations G_{\max} . The initial individuals \mathbf{x}_i^0 ($i = 1, \dots, NP$) are generated randomly in \mathbf{P}_0 . The current generation G is set as 0.

2) Mutation: After initialization operation, for each \mathbf{x}_i^G , the mutant vector \mathbf{v}_i^G is produced by mutation operation. Some of the most frequently used mutation strategies are given as follows:

“DE/rand/1”

$$\mathbf{v}_i^G = \mathbf{x}_{r_1}^G + F \times (\mathbf{x}_{r_2}^G - \mathbf{x}_{r_3}^G) \quad (2)$$

“DE/rand/2”

$$\mathbf{v}_i^G = \mathbf{x}_{r_1}^G + F \times (\mathbf{x}_{r_2}^G - \mathbf{x}_{r_3}^G) + F \times (\mathbf{x}_{r_4}^G - \mathbf{x}_{r_5}^G) \quad (3)$$

“DE/current-to-best/1”

$$\mathbf{v}_i^G = \mathbf{x}_i^G + F \times (\mathbf{x}_{\text{best}}^G - \mathbf{x}_i^G) + F \times (\mathbf{x}_{r_1}^G - \mathbf{x}_{r_2}^G) \quad (4)$$

“DE/current-to-best/2”

$$\mathbf{v}_i^G = \mathbf{x}_i^G + F \times (\mathbf{x}_{\text{best}}^G - \mathbf{x}_i^G) + F \times (\mathbf{x}_{r_1}^G - \mathbf{x}_{r_2}^G + \mathbf{x}_{r_3}^G - \mathbf{x}_{r_4}^G) \quad (5)$$

“DE/rand-to-best/1”

$$\mathbf{v}_i^G = \mathbf{x}_{r_1}^G + F \times (\mathbf{x}_{\text{best}}^G - \mathbf{x}_i^G) + F \times (\mathbf{x}_{r_2}^G - \mathbf{x}_{r_3}^G) \quad (6)$$

where r_1, r_2, r_3, r_4 and r_5 are mutually exclusive integers randomly chosen within the range $[1, NP]$, and are also different from the index i (i.e., $r_1 \neq r_2 \neq r_3 \neq r_4 \neq r_5 \neq i$); $\mathbf{x}_{\text{best}}^G$ is the target vector with the best fitness value at the G -th generation.

3) Crossover: For each \mathbf{x}_i^G , trial vector \mathbf{u}_i^G is generated as follows:

$$\mathbf{u}_{ij}^G = \begin{cases} v_{ij}^G, & \text{if } R_j \leq CR \text{ or } j = j_{\text{rand}} \\ x_{ij}^G, & \text{otherwise} \end{cases} \quad j = 1, 2, \dots, D \quad (7)$$

where R_j is a uniform random number within the range $[0, 1]$ and j_{rand} is a randomly chosen integer within the range $[1, D]$, respectively.

4) Selection: The trial vector \mathbf{u}_i^G competes with its target vector \mathbf{x}_i^G and the better one of them is selected for the next generation:

$$\mathbf{x}_i^{G+1} = \begin{cases} \mathbf{u}_i^G, & \text{if } f(\mathbf{u}_i^G) \leq f(\mathbf{x}_i^G) \\ \mathbf{x}_i^G, & \text{otherwise.} \end{cases} \quad (8)$$

5) $G = G + 1$.

6) Implement Steps 2 to 5 repeatedly until the number of generations is equal to G_{max} .

3 JADE using two-phase parameter control scheme

For most evolutionary algorithms, it is generally believed to be a good idea to encourage the global search in

the early stages of the evolution and ensure the local search in the later evolution phases^[44], thus the above view can be considered as the empirical guideline. For the DE algorithm, a large value of F can provide good exploration capability, whereas a small value of F can accelerate the convergence speed^[3]. Meanwhile, a large value of CR can provide good exploitation capability^[45]. In the current study, this empirical guideline can be used to improve the search efficiency of the DE in the later evolution stages. Namely, the value of F gradually decreases and the value of CR gradually increases during the entire evolutionary process.

The adaptation technique can enable control parameters to make timely adjustment when compared with the empirical guideline. In other words, the adaptability of the control parameters produced by the adaptation approach is better than that of the control parameters generated by the empirical guideline. Additionally, the fitness landscape of the population is usually very complex in the early stages of the search since individuals in the current population may be distributed in the entire search space, but individuals may gather in a small search region in the later phases of the evolution. Based on the above introduction, the parameters of JADE are adjusted through two different approaches in the current study. Namely, the parameter adaptation method proposed in JADE^[10] is used in the early evolution stages and the empirical guideline is utilized to guide the evolution of control parameters in the later phases of the evolution. The main target is to take advantages of the two different parameter control approaches.

3.1 Evolution of control parameters

If $G < gs \times G_{\text{max}}$, the F value of each individual is produced in [10]; otherwise, the F value of each individual is generated as follows:

$$F_i^{G+1} = \text{Cauchy} \left(\left(0.6 - 0.5 \times \frac{G - gs \times G_{\text{max}}}{(1 - gs) \times G_{\text{max}}} \right), \sigma_1 \right) \quad i = 1, \dots, NP \quad (9)$$

where *Cauchy* is a Cauchy distribution function, $\sigma_1 = 0.6$, $gs = 0.6$. It can be seen from (9) that TPC-JADE has good local search ability in the later search stages since Cauchy distribution function with a low location parameter tends to generate a small value of F (i.e., promoting exploitation). Moreover, the location parameter of Cauchy distribution function is within the range of $[0.1, 0.6]$.

If $G < gs \times G_{\text{max}}$, the CR value of each individual is generated in [10]; otherwise, the CR value of each individual is generated as follows:

$$CR_i^{G+1} = N \left(\left(1 - 0.5 \times \left(1 - \frac{G - gs \times G_{\max}}{(1 - gs) \times G_{\max}} \right) \right), \sigma_2 \right) \\ i = 1, \dots, NP \quad (10)$$

where N is a normal distribution function, $\sigma_2 = 0.6$. We can observe from (10) that a large CR value generated by a normal distribution function with large mean value can speed up convergence in the later phases of the search. Moreover, the mean value of normal distribution function is within the range of $[0.5, 1]$.

Based on the above descriptions, it can be observed that the empirical guideline can provide a good exploitation capability for the DE algorithm. Therefore, it is useful to improve the search efficiency of DE in the later search process. Additionally, the parameter adaptation approach used in the early evolution stages can help to locate the promising search region. This is because the adaptation technique based on the population information can automatically adjust the control parameters when the fitness landscape of the population is very complex.

3.2 Overall implementation of TPC-JADE

1) Initialization

Determine the population size NP and maximal number of generation G_{\max} . Initialize a population P_1^0 that is generated randomly within the feasible search space P_0 . Set the current generation $G = 0$, $gs = 0.6$.

2) Population evolution

Mutation operation: For each x_i^G , a mutant vector v_i^G is generated as follows^[10]:

$$v_i^G = x_i^G + F_i \times (x_{p_{best}}^G - x_i^G) + F_i \times (x_{r_1}^G - x_{r_2}^G) \quad (11)$$

where $x_{p_{best}}^G$ is randomly selected from the best 100p% individuals in the current population, and p is within the range $(0, 1]$.

Crossover operation: Equation (7) is used to generate a trial vector u_i^G .

3) Control parameters adaptation

The more detailed descriptions can be seen in Section 3.1.

4) $G = G + 1$.

5) Steps 2 to 4 are repeated until the maximum number of generations is equal to G_{\max} .

The framework of the proposed algorithm is shown in Fig. 1.

4 Experimental results

In this section, two famous test suites (i.e., IEEE CEC2005 and IEEE CEC2015) are utilized to assess the performance of the TPC-JADE. Moreover, the performance of the TPC-JADE is compared with that of four

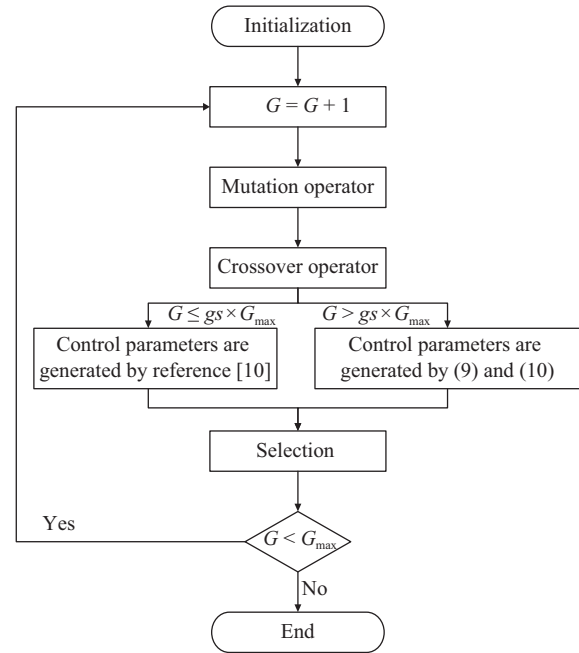


Fig. 1 Framework of the proposed algorithm

state-of-the-art DE variants, namely, jDE^[9], SaDE^[7], JADE^[10], and CoBiDE^[13]. All the compared algorithms are coded in Matlab (Matlab R2012a) and run on Windows 7 operating system (64bit). For each test function, the maximum numbers of fitness evaluations are set to 300 000 for 30-dimensional functions and 500 000 for 50-dimensional functions. Since each DE variant has its own suitable NP , recommended NP from their literatures are utilized, namely, 100 for jDE, JADE, and TPC-JADE, 50 for SaDE, and 60 for CoBiDE. Furthermore, to compare the performance of all the algorithms, the Wilcoxon's rank sum test^[46] at the 0.05 significance level and the Friedman's test^[47] are utilized. For the Wilcoxon's rank sum test, the “+”, “−” and “≈” signs denote that the search performance of the proposed algorithm is significantly better than, worse than, and almost similar to that of the other compared algorithms in a statistically significant way, respectively.

4.1 Comparison with four DE variants on 30-dimensional CEC2005 functions

In this part, twenty-five 30-dimensional CEC2005 test functions are used to evaluate the performance of the TPC-JADE. Each test function is independently run 30 times. The results are presented in Table 1. For unimodal test functions F1_{CEC2005}–F5_{CEC2005}, as shown in Table 1, the overall performance of TPC-JADE is better than that of jDE, SaDE, and CoBiDE since TPC-JADE uses a greedy mutation strategy which is the same as JADE. JADE significantly outperforms TPC-JADE on three test functions. The main reason may be that the evolution information produced by unimodal test function is less mis-

Table 1 Results of five DE variants on 30-dimensional CEC2005 test functions

Function		jDEMean (Std)	SaDEMean (Std)	JADEMean (Std)	CoBiDEMean (Std)	TPC-JADEMean (Std)
Unimodal functions	F1 _{CEC2005}	0.00E+00(0.00E+00)≈	0.00E+00(0.00E+00)≈	0.00E+00(0.00E+00)≈	0.00E+00(0.00E+00)≈	0.00E+00(0.00E+00)
	F2 _{CEC2005}	4.78E-07(6.58E-07)+	1.12E-05(2.35E-05)+	8.15E-29(9.14E-29)+	1.19E-12(2.04E-12)+	1.86E-29(4.74E-29)
	F3 _{CEC2005}	2.03E+05(1.32E+05)+	5.35E+05(2.30E+05)+	8.18E+03(6.86E+03)-	7.80E+04(5.03E+04)+	1.20E+04(8.25E+03)
	F4 _{CEC2005}	3.44E-02(1.04E-01)+	1.45E+02(1.98E+02)+	8.39E-16(3.08E-15)-	8.88E-04(1.93E-03)+	1.09E-13(4.29E-13)
	F5 _{CEC2005}	4.46E+02(3.67E+02)+	3.13E+03(4.95E+02)+	4.86E-07(2.35E-06)-	3.71E+01(5.26E+01)+	5.64E-05(1.70E-04)
Basic multimodal functions	F6 _{CEC2005}	2.25E+01(2.41E+01)+	4.01E+01(3.54E+01)+	2.58E+00(7.46E+00)≈	1.66E-01(7.23E-01)+	2.72E-01(9.64E-01)
	F7 _{CEC2005}	1.16E-02(9.04E-03)≈	1.84E-02(1.80E-02)+	9.36E-03(9.02E-03)≈	3.68E-03(8.30E-03)-	9.19E-03(7.11E-03)
	F8 _{CEC2005}	2.09E+01(7.51E-02)≈	2.09E+01(3.67E-02)≈	2.09E+01(1.43E-01)≈	2.07E+01(4.05E-01)≈	2.08E+01(2.06E-01)
	F9 _{CEC2005}	0.00E+00(0.00E+00)≈	9.95E-02(3.04E-01)≈	0.00E+00(0.00E+00)≈	0.00E+00(0.00E+00)≈	0.00E+00(0.00E+00)
	F10 _{CEC2005}	5.79E+01(9.38E+00)+	4.48E+01(1.25E+01)+	2.43E+01(5.23E+00)+	4.34E+01(1.40E+01)+	2.09E+01(4.71E+00)
	F11 _{CEC2005}	2.83E+01(1.25E+00)+	1.65E+01(2.94E+00)-	2.53E+01(1.29E+00)+	5.67E+00(2.36E+00)-	2.17E+01(2.76E+00)
F12 _{CEC2005}	1.20E+04(9.44E+03)+	2.17E+03(1.82E+03)+	6.68E+03(4.17E+03)+	2.96E+03(2.60E+03)+	1.31E+03(1.39E+03)	
Expanded multimodal functions	F13 _{CEC2005}	1.66E+00(1.28E-01)+	3.90E+00(4.64E-01)+	1.48E+00(1.26E-01)+	2.64E+00(1.08E+00)+	1.25E+00(1.75E-01)
	F14 _{CEC2005}	1.30E+01(2.69E-01)+	1.26E+01(2.86E-01)+	1.23E+01(2.96E-01)≈	1.22E+01(5.32E-01)≈	1.22E+01(3.87E-01)
Hybrid composition functions	F15 _{CEC2005}	3.18E+2(1.19E+02)≈	3.74E+2(6.42E+01)≈	3.76E+02(8.97E+01)≈	4.10E+02(5.47E+01)+	3.71E+02(6.91E+01)
	F16 _{CEC2005}	8.49E+01(3.59E+01)+	7.71E+01(2.74E+01)+	9.63E+01(1.08E+02)+	8.42E+01(6.31E+01)+	6.84E+01(8.67E+01)
	F17 _{CEC2005}	1.39E+02(1.25E+01)+	8.70E+01(8.49E+01)+	1.02E+02(6.68E+01)+	6.82E+01(2.05E+01)+	6.86E+01(4.05E+01)
	F18 _{CEC2005}	9.04E+02(7.96E-01)+	8.78E+02(6.06E+01)≈	9.04E+02(7.92E-01)+	9.04E+02(9.98E-01)+	9.04E+02(6.16E-01)
	F19 _{CEC2005}	9.04E+02(7.69E-01)≈	8.60E+02(6.13E+01)≈	9.04E+02(9.94E-01)+	9.04E+02(8.62E-01)≈	9.04E+02(7.44E-01)
	F20 _{CEC2005}	9.04E+02(9.216E-01)+	8.73E+02(6.14E+01)≈	9.04E+02(5.93E-01)+	9.04E+02(5.47E-01)+	9.04E+02(5.41E-01)
	F21 _{CEC2005}	5.00E+02(1.96E-13)+	5.43E+02(1.65E+02)+	5.10E+02(5.48E+01)≈	5.00E+02(1.75E-13)≈	5.00E+02(1.76E-13)
	F22 _{CEC2005}	8.67E+02(1.87E+01)≈	9.36E+02(1.88E+01)+	8.64E+02(2.57E+01)≈	8.54E+02(2.46E+01)≈	8.59E+02(2.19E+01)
	F23 _{CEC2005}	5.34E+02(2.98E-04)+	5.69E+02(1.34E+02)+	5.34E+02(2.17E-04)+	5.34E+02(4.70E-07)+	5.34E+02(3.11E-13)
	F24 _{CEC2005}	2.00E+02(2.89E-14)≈	2.00E+02(2.89E-14)≈	2.00E+02(2.89E-14)≈	2.00E+02(2.89E-14)≈	2.00E+02(2.89E-14)
	F25 _{CEC2005}	2.11E+02(7.01E-01)≈	2.13E+02(1.39E+00)+	2.11E+02(8.53E-01)+	2.10E+02(5.65E-01)-	2.11E+02(5.98E-01)
+	16	16	12	14		
-	0	1	3	3		
≈	9	8	10	8		

leading. Therefore, the adaptation technique is more effective than the empirical guideline. For multimodal test functions F6_{CEC2005}–F14_{CEC2005}, jDE and JADE cannot perform better than TPC-JADE on any test functions. Moreover, SaDE and CoBiDE perform better than TPC-JADE on one and two test functions, respectively. However, the performance of TPC-JADE is significantly better than that of SaDE and CoBiDE on six and four test functions, respectively. Therefore, the average performance of TPC-JADE is the best among five compared algorithms on multimodal test functions. For hybrid composition functions F15_{CEC2005}–F25_{CEC2005}, jDE, SaDE, and JADE cannot outperform TPC-JADE on any test functions since the adaptation technique can help TPC-JADE to adapt complex environment in the early stages of the evolution and the empirical guideline can improve the search efficiency of the DE in the later evolution phases.

Moreover, the CoBiDE performs better than TPC-JADE on one test function. However, the performance of TPC-JADE is significantly better than that of CoBiDE on six test functions.

Based on the above comparisons, it can be observed from Table 1 that the average performance of TPC-JADE is the best among five compared DE variants, the adaptation technique can assist DE to find optimal/near search region in the early evolution stages, while the empirical guideline can enhance the local search capability of DE at later phases of the search.

4.2 Comparison with four DE variants on 50-dimensional CEC2005 functions

In this section, twenty-five 50-dimensional CEC2005 test functions are employed to verify the performance of

the proposed algorithm. For each test function, the setting of run times is the same as in Section 4.1. The simulation and statistical analysis results are shown in Table 2. For unimodal test functions F1_{CEC2005}–F5_{CEC2005}, Table 2 indicates that the TPC-JADE outperforms all the compared algorithms except for JADE. However, JADE performs better than our proposed algorithm on only one test function. Based on the results shown in Table 1, we can find that the local search capability of TPC-JADE is not getting worse with the increasing of the dimension when compared with JADE. For multimodal test functions F6_{CEC2005}–F14_{CEC2005}, as shown in Table 2, jDE cannot perform better than the proposed algorithm on any test functions. SaDE, JADE, and CoBiDE perform better than TPC-JADE on one test function, respectively. However, TPC-JADE performs better than SaDE, JADE, and CoBiDE on seven, five, and six test functions, re-

spectively. Therefore, the optimization performance of TPC-JADE is the best among these selected algorithms on the multimodal test functions. For hybrid composition functions F15_{CEC2005}–F25_{CEC2005}, JADE and SaDE cannot perform better than TPC-JADE on any test functions. The performance of jDE and CoBiDE is significantly better than that of TPC-JADE on one and four test functions, respectively. However, TPC-JADE significantly performs better than jDE and CoBiDE on four and five test functions, respectively. Therefore, the average performance of TPC-JADE is similar as that of CoBiDE on complex functions and is better than that of jDE, SaDE, and JADE.

Based on the above analyses, the statistical analysis results shown in Table 2 indicate that the overall performance of TPC-JADE is better than that of other competitors on twenty-five 50-dimensional CEC2005 functions.

Table 2 Results of five DE variants on 50-dimensional CEC2005 test functions

Function		jDEMean (Std)	SaDEMean (Std)	JADEMean (Std)	CoBiDEMean (Std)	TPC-JADEMean (Std)
Unimodal functions	F1 _{CEC2005}	0.00E+00(0.00E+00)≈	3.36E-30(1.28E-29)≈	0.00E+00(0.00E+00)≈	0.00E+00(0.00E+00)≈	0.00E+00(0.00E+00)
	F2 _{CEC2005}	8.35E-03(9.43E-03)+	8.03E-02(7.62E-02)+	3.97E-27(2.00E-027)-	1.39E-06(1.77E-06)+	9.05E-24(1.42E-23)
	F3 _{CEC2005}	4.78E+05(1.99E+05)+	9.31E+05(2.63E+05)+	1.77E+04(9.43E+03)≈	2.47E+05(1.00E+05)+	2.26E+04(1.17E+04)
	F4 _{CEC2005}	4.33E+02(3.21E+02)+	5.88E+03(2.52E+03)+	2.21E+00(7.63E+00)≈	2.30E+02(1.97E+02)+	8.43E-01(1.65E+00)
	F5 _{CEC2005}	3.25E+03(7.18E+02)+	8.31E+03(1.56E+03)+	1.53E+03(4.76E+02)≈	2.60E+03(6.25E+02)+	1.58E+03(4.09E+02)
Basic multimodal functions	F6 _{CEC2005}	4.32E+01(2.85E+01)+	9.48E+01(4.81E+01)+	6.64E-01(1.51E+00)-	2.89E+01(2.33E+01)+	4.81E+00(1.31E+01)
	F7 _{CEC2005}	4.18E-03(9.89E-03)+	5.08E-03(9.98E-03)+	5.33E-03(8.76E-03)≈	4.92E-03(7.25E-03)+	3.19E-03(8.68E-03)
	F8 _{CEC2005}	2.11E+01(3.53E-02)+	2.11E+01(3.44E-02)+	2.10E+01(3.56E-01)+	2.08E+01(5.26E-01)≈	2.05E+01(3.41E-01)
	F9 _{CEC2005}	0.00E+00(0.00E+00)≈	1.46E+00(1.25E+00)+	0.00E+00(0.00E+00)≈	8.52E-13(3.14E-12)+	0.00E+00(0.00E+00)
	F10 _{CEC2005}	1.05E+02(1.72E+01)+	1.30E+02(2.42E+01)+	4.94E+01(8.04E+00)+	8.51E+01(2.04E+01)+	4.21E+01(5.48E+00)
	F11 _{CEC2005}	5.42E+01(2.07E+00)+	3.92E+01(4.78E+00)-	5.16E+01(2.36E+00)+	1.89E+01(4.43E+00)-	4.82E+01(4.07E+00)
	F12 _{CEC2005}	1.59E+04(1.34E+4)≈	1.58E+04(1.10E+04)≈	1.56E+04(1.14E+04)≈	1.68E+04(1.79E+04)≈	1.25E+04(1.07E+04)
Expanded multimodal	F13 _{CEC2005}	3.05E+00(2.19E-01)+	9.90E+00(1.08E+00)+	2.75E+00(1.39E-01)+	4.36E+00(1.33E+00)+	2.37E+00(2.05E-00)
	F14 _{CEC2005}	2.25E+01(2.13E-01)+	2.23E+01(2.88E-01)+	2.17E+01(5.07E-01)+	2.19E+01(4.06E-01)+	2.15E+01(4.43E-01)
Hybrid composition functions	F15 _{CEC2005}	3.20E+02(9.97E+01)≈	3.64E+02(8.01E+01)+	3.49E+02(8.74E+01)+	3.87E+02(5.07E+01)+	2.74E+02(9.36E+01)
	F16 _{CEC2005}	8.69E+01(2.07E+01)+	9.29E+01(5.96E+01)+	8.06E+01(9.15E+01)+	8.48E+01(6.33E+01)+	4.25E+01(6.68E+00)
	F17 _{CEC2005}	1.83E+02(4.92E+01)+	8.23E+01(1.49E+01)+	1.58E+02(1.27E+02)+	7.82E+01(2.97E+01)+	6.17E+01(6.58E+01)
	F18 _{CEC2005}	9.20E+02(2.45E+00)≈	9.91E+02(1.58E+01)+	9.22E+02(2.74E+00)≈	9.18E+02(2.80E+00)-	9.20E+02(6.45E+00)
	F19 _{CEC2005}	9.20E+02(3.59E+00)≈	9.79E+02(3.72E+01)+	9.24E+02(3.51E+00)+	9.14E+02(2.17E+01)-	9.21E+02(3.60E+00)
	F20 _{CEC2005}	9.20E+02(3.17E+00)-	9.84E+02(1.49E+01)+	9.21E+02(4.01E+00)≈	9.18E+02(2.73E+00)-	9.22E+02(4.04E+00)
	F21 _{CEC2005}	6.85E+02(2.47E+02)≈	6.39E+02(2.82E+02)+	5.51E+02(1.55E+02)+	5.50E+02(1.53E+02)+	5.00E+02(7.61E-13)
	F22 _{CEC2005}	9.00E+02(9.42E+00)≈	9.85E+02(1.27E+01)+	9.00E+02(2.30E+01)≈	8.86E+02(2.80E+01)≈	8.91E+02(2.20E+01)
	F23 _{CEC2005}	8.39E+02(2.32E+02)+	7.36E+02(3.06E+02)≈	6.34E+02(1.92E+02)≈	6.17E+02(1.79E+02)≈	5.87E+02(1.43E+02)
	F24 _{CEC2005}	2.00E+02(1.56E-12)+	6.29E+02(4.87E+02)+	2.00E+02(1.51E-12)+	2.00E+02(1.56E-12)+	2.00E+02(5.99E-13)
	F25 _{CEC2005}	2.16E+02(1.39E+00)≈	2.24E+02(3.94E+00)+	2.19E+02(2.76E+00)+	2.16E+02(1.13E+00)-	2.17E+02(1.40E+00)
+	15	21	12	15		
-	1	1	2	5		
≈	9	3	11	5		

4.3 Comparison with four DE variants on 30-dimensional CEC2015 functions

In this experiment, fifteen 30-dimensional functions introduced in IEEE CEC2015 are employed to demonstrate the performance of TPC-JADE. For each function, the run times are set to be 30, and the mean and standard deviation values are presented in Table 3. Additionally, the statistical analysis results obtained by the Wilcoxon's rank sum test are also shown in Table 3.

According to the results summarized in Table 3, it can be observed that the performance of TPC-JADE is significantly better than that of jDE, SaDE, JADE, and CoBiDE on ten, twelve, nine, and seven functions, respectively. Note that jDE, SaDE, and JADE cannot provide significantly better results than TPC-JADE on any functions. We can find that the control parameters guided by the empirical guideline can improve the search efficiency of JADE. It can be also seen from Table 3 that CoBiDE significantly outperforms TPC-JADE on four functions. This is because the covariance matrix learning is used in the DE. Overall, TPC-JADE provides the best average performance among all compared algorithms on 15 30-dimensional IEEE CEC2015 functions.

5 Parameter analysis

In this part, 25 30-dimensional CEC2005 test functions are utilized to investigate the effect of σ , which is selected from the set (0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8).

The maximum number of function evaluations and run times are set to be 300 000 and 30, respectively.

The statistical analysis results achieved by the Friedman's test are plotted in Fig. 2. As shown in Fig. 2, we can find that the overall performance of TPC-JADE is susceptible to the value of sigma and is the best when $\sigma = 0.6$. Therefore, $\sigma = 0.6$ is chosen in our proposed algorithm.

6 Case study

A typical dynamic optimization problem can be formulated as follows^[39, 48]:

$$\begin{aligned} \min \quad & J(\mathbf{u}(t)) = G(x(t_f)) + \int_{t_0}^{t_f} F(\mathbf{x}, \mathbf{u}) dt \\ \text{s.t.} \quad & \frac{d\mathbf{x}(t)}{dt} = f(\mathbf{x}(t), \mathbf{u}(t)), \quad \mathbf{x}(0) = \mathbf{x}_0 \\ & 0 \leq t \leq t_f, \quad \mathbf{u}_{\min} < \mathbf{u} < \mathbf{u}_{\max} \end{aligned} \quad (12)$$

where J denotes the performance index; \mathbf{x} and \mathbf{u} are the state and control vectors, respectively; G and F are the terminal time performance index and integrated performance, respectively; t_0 is the initial time and t_f is the final time; \mathbf{u}_{\min} and \mathbf{u}_{\max} are the lower and upper bounds of \mathbf{u} , respectively.

6.1 Feeding-rate optimization for Lee-Ramirez bioreactor

In this section, TPC-JADE is utilized to solve a dy-

Table 3 Results of all compared algorithms on 15 30-dimensional CEC2015 test functions

Function	jDEMean (Std)	SaDEMean (Std)	JADEMean (Std)	CoBiDEMean (Std)	TPC-JADEMean (Std)
F1 _{CEC2015}	3.43E+04(2.25E+04)+	5.40E+05(3.33E+05)+	1.42E+01(4.73E+01)≈	6.04E+03(5.15E+03)+	1.38E+01(2.65E+01)
F2 _{CEC2015}	2.90E-09(8.29E-09)+	1.22E+03(1.51E+03)+	3.12E-14(8.67E-15)≈	5.68E-15(1.37E-14)-	2.74E-14(5.18E-15)
F3 _{CEC2015}	2.03E+01(2.95E-02)+	2.05E+01(4.01E-02)+	2.03E+01(2.79E-02)+	2.04E+01(2.54E-01)+	2.01E+01(9.66E-02)
F4 _{CEC2015}	4.64E+01(5.09E+00)+	4.00E+01(9.60E+00)+	2.55E+01(4.33E+00)≈	4.09E+01(1.46E+01)+	2.33E+01(4.54E+00)
F5 _{CEC2015}	2.59E+03(1.88E+02)+	3.20E+03(7.59E+02)+	1.80E+03(2.63E+02)+	1.81E+03(5.07E+02)+	1.46E+03(2.75E+02)
F6 _{CEC2015}	2.32E+03(1.95E+03)+	1.36E+04(1.26E+04)+	1.10E+03(3.76E+02)+	2.16E+02(1.97E+02)-	8.48E+02(3.71E+02)
F7 _{CEC2015}	8.34E+00(8.19E-01)+	6.81E+00(1.38E+00)+	8.00E+00(8.07E-01)+	3.16E+00(9.88E-01)+	2.61E+00(5.12E-01)
F8 _{CEC2015}	1.48E+02(1.11E+02)≈	5.77E+03(5.41E+03)+	1.07E+04(3.37E+04)+	7.72E+01(8.08E+01)-	1.55E+02(1.18E+02)
F9 _{CEC2015}	1.02E+02(9.72E-02)≈	1.03E+01(2.10E-01)≈	1.03E+02(1.60E-01)+	1.03E+02(1.84E-01)≈	1.03E+02(1.68E-01)
F10 _{CEC2015}	8.01E+02(2.08E+02)≈	8.66E+03(7.95E+03)+	4.69E+03(1.48E+04)≈	4.17E+02(1.47E+02)-	8.03E+02(2.96E+02)
F11 _{CEC2015}	3.84E+02(5.49E+01)≈	4.62E+02(1.39E+02)≈	4.26E+02(5.92E+01)≈	3.57E+02(6.44E+01)≈	4.08E+02(3.49E+01)
F12 _{CEC2015}	1.06E+02(4.21E-01)+	1.06E+02(5.34E-01)+	1.05E+02(5.43E-01)+	1.06E+02(5.96E-02)+	1.05E+02(4.28E-01)
F13 _{CEC2015}	1.01E+02(3.20E+00)+	1.09E+02(3.07E+00)+	9.65E+01(2.81E+00)+	9.88E+01(4.83E+00)+	9.37E+01(5.16E+00)
F14 _{CEC2015}	3.25E+04(9.22E+02)+	3.32E+04(8.00E+02)+	3.22E+04(9.05E+02)+	3.17E+04(9.45E+02)≈	3.16E+04(5.92E+02)
F15 _{CEC2015}	1.00E+01(8.50E-14)≈	1.00E+01(8.50E-14)≈	1.00E+01(7.22E-14)≈	1.00E+01(7.22E-02)≈	1.00E+02(7.22E-14)
+	10	12	9	7	
-	0	0	0	4	
≈	5	3	6	4	

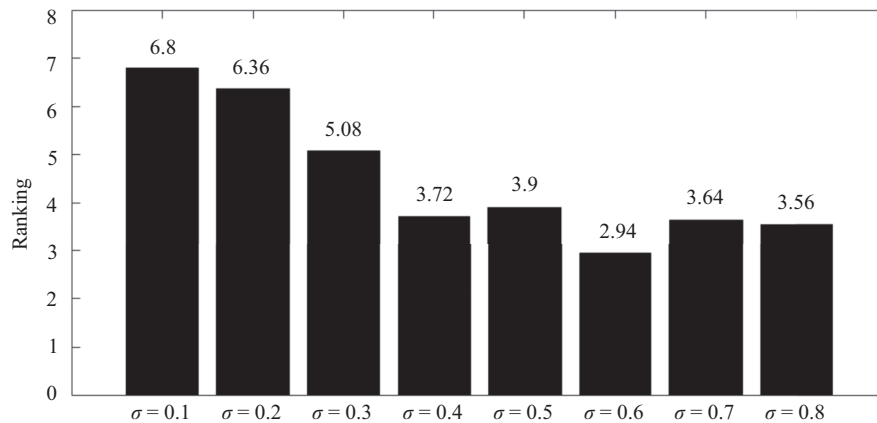


Fig. 2 Rankings obtained by Friedman's test under different values of sigma

dynamic optimization problem proposed by Lee and Ramirez^[49]. Additionally, the direct method adopted in ^[50] is employed. This dynamic optimization problem can be described as follows:

$$\begin{aligned}
 \frac{dx_1}{dt} &= u_1 + u_2, & x_1(0) &= 1 \\
 \frac{dx_2}{dt} &= \mu x_2 - \frac{u_1 + u_2}{x_1} x_2, & x_2(0) &= 0.1 \\
 \frac{dx_3}{dt} &= \frac{u_1}{x_1} C_{nf} - \frac{u_1 + u_2}{x_1} x_3 - Y^{-1} \mu x_2, & x_3(0) &= 40 \\
 \frac{dx_4}{dt} &= R_{fp} x_2 - \frac{u_1 + u_2}{x_1} x_4, & x_4(0) &= 0 \\
 \frac{dx_5}{dt} &= \frac{u_2}{x_1} C_{if} - \frac{u_1 + u_2}{x_1} x_5, & x_5(0) &= 0 \\
 \frac{dx_6}{dt} &= -k_1 x_6, & x_6(0) &= 1 \\
 \frac{dx_7}{dt} &= k_2 (1 - x_7), & x_7(0) &= 0 \\
 0 &\leq u_1, u_2 \leq 0.01
 \end{aligned} \tag{13}$$

where x_1 (L), x_2 (g/L), x_3 (g/L), x_4 (g/L) and x_5 (g/L) are the reaction volume, cell density, nutrient concentration, foreign protein concentration, and inducer concentration, respectively; x_6 is inducer shock factor and x_7 is inducer recovery factor on the cell growth rate; u_1 (L/h) is the glucose feed rate, and u_2 (L/h) is the inducer feed rate; Y , C_{nf} (g/h), C_{if} (g/h), μ (1/h), and R_{fp} (1/h) are the growth yield coefficient, nutrient concentration in the nutrient feed, inducer concentration in the inducer feed, specific growth rate, and foreign protein production rate, respectively; k_1 is the shock parameter and k_2 is recovery parameter.

In (13), some parameters can be defined as follows:

$$\mu = \frac{0.407x_3}{0.108 + x_3 + x_3^2} \left(x_6 + \frac{0.22}{0.22 + x_3} x_7 \right)$$

$$k_1 = k_2 = \frac{0.09x_5}{0.034 + x_5}$$

$$R_{fp} = \frac{0.095x_3}{\frac{0.108 + x_3 + x_3^2}{14\,814.8}} \left(\frac{0.005 + x_5}{0.022 + x_5} \right)$$

$$C_{if} = 4, C_{nf} = 100, Y = 0.51.$$

For the above fed-batch production of induced foreign protein, the objective is to maximize the economic benefit of this fed-batch system. Therefore, the objective function can be defined as

$$\max J(u_1, u_2) = x_1(t_f) x_4(t_f) - Q \int_0^{t_f} u_2(t) dt \tag{14}$$

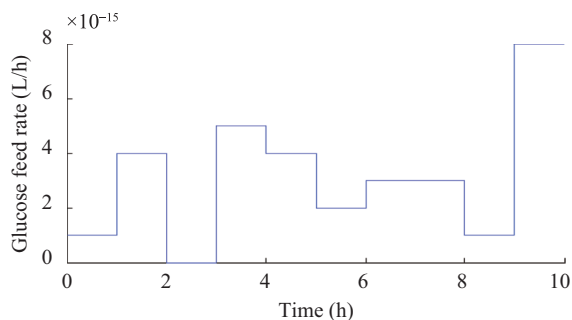
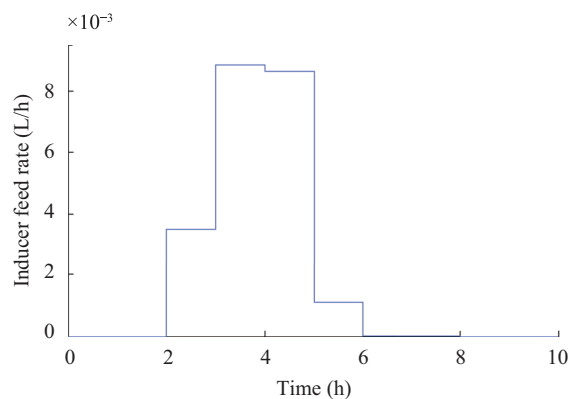
where $Q = 5$ and $t_f = 10$ h.

For this dynamic optimization problem, researchers have introduced various methods to obtain satisfactory results, which are 0.8149^[51], 0.8167^[52], 0.8158^[53] and 0.8164^[54].

In this experiment, NP and G_{\max} of TPC-JADE are set to be 100 and $D \times 10\,000$, respectively. Moreover, three different discrete time degrees (i.e., D) are discussed as follows.

1) The value 0.81643 is achieved by TPC-JADE when D is set to be 10. It can be seen that our result is better than the results achieved by Roubos et al.^[51], Zhang et al.^[53], and Fan et al.^[54] respectively. However, the result of TPC-JADE is slightly worse than the result obtained by Sarkar and Modak^[52]. The main reason may be that Sarkar and Modak^[52] consume much more computational resources and the discrete time length is smaller than our proposal. In addition, the curves of two control variables (i.e., u_1 and u_2) are plotted in Figs. 3 and 4, respectively.

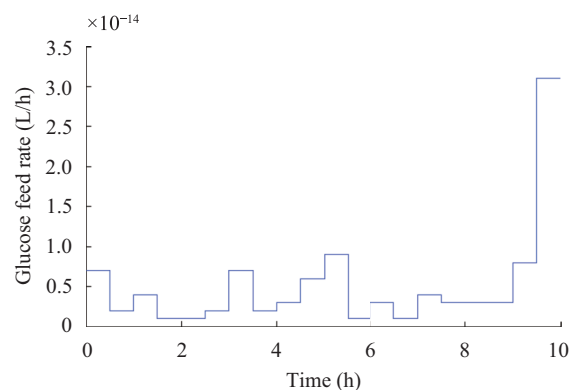
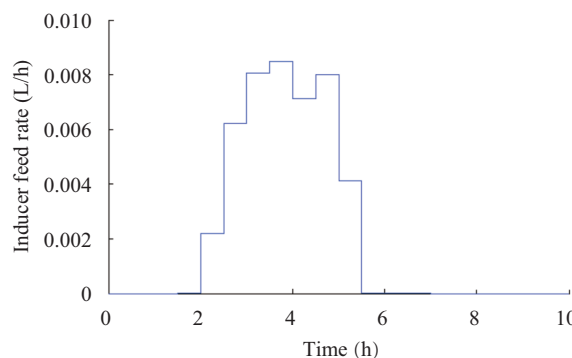
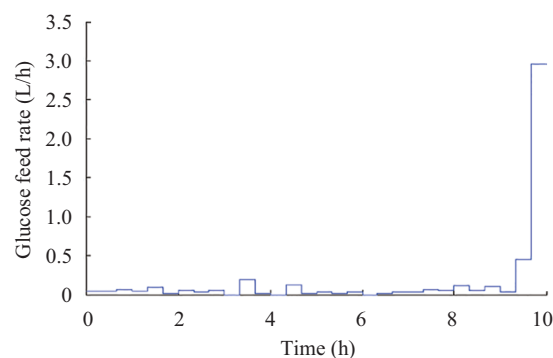
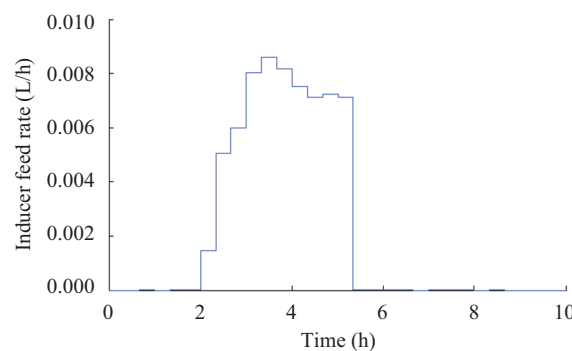
2) Let $D = 20$, the value 0.81647 is achieved by using TPC-JADE. Clearly, the result achieved by TPC-JADE is better than the other competitors except for the result obtained by Sarkar and Modak^[52]. This is because Sarkar

Fig. 3 Optimal control profile of glucose feed rate ($D = 10$)Fig. 4 Optimal control profile of inducer feed rate ($D = 10$)

and Modak consume more computational resources when compared with the proposed algorithm. Moreover, the obtained result is better than the result obtained by TPC-JADE when $D = 10$. It means that the solution precision is directly influenced by the discrete time degree. Additionally, the curves of the optimal glucose feed rate and inducer feed rate are presented in Figs. 5 and 6, respectively.

3) In this experiment, D is set to be 30. The value 0.81648 is achieved by the proposed algorithm. Although the result is slightly worse than Sarkar and Modak[52], the obtained result is better than the results reported by Roubos et al.[51], Zhang et al.[53] and Fan et al.[54], respectively. The obtained result is also better than the results achieved by TPC-JADE when $D = 20$. It can be observed from the obtained result that TPC-JADE is a very competitive optimization tool for solving a complex dynamic optimization problem. The curves of optimal u_1 and u_2 are shown in Figs. 7 and 8, respectively.

Based on the above simulation results, it can be concluded that the solution precision is highly influenced by the dimensionality of the problem and the TPC-JADE can perform well when the dimensionality of the problem is increasing. It should be noted that the value 0.8167 is achieved by Sarkar and Modak[52] when the number of function evaluations is more than 2000000. For our proposal, the maximum number of function evaluations is 300000; therefore, the proposed algorithm is very compet-

Fig. 5 Optimal control profile of glucose feed rate ($D = 20$)Fig. 6 Optimal control profile of inducer feed rate ($D = 20$)Fig. 7 Optimal control profile of glucose feed rate ($D = 30$)Fig. 8 Optimal control profile of inducer feed rate ($D = 30$)

itive for solving this dynamic optimization problem. Additionally, from Figs. 3–8, it can be observed that the nutrient can be neglected when $Q = 5$. This is because glucose feed rate is maintained at zero during the entire period of operation.

7 Conclusions

In the current study, a self-adaptive DE using two-phase parameter control scheme (TPC-JADE) is proposed to improve the search efficiency of DE. In the TPC-JADE, the adaptation technique and empirical guideline are used to produce the control parameters in the early evolution phases and later evolution stages, respectively. Therefore, TPC-JADE can adapt to the complex optimization environment and improve the search efficiency of DE in two different stages of the search. The search performance of TPC-JADE is compared with that of jDE, SaDE, JADE, and CoBiDE on two well-known test suites. Moreover, the Wilcoxon's rank sum test is utilized to distinguish the performance difference between TPC-JADE and its competitors. The simulation and statistical analysis results indicate that TPC-JADE performs better than jDE, SaDE, JADE, and CoBiDE on these selected test functions, and the empirical guideline can enhance the search efficiency of DE.

The sensitivity of sigma in TPC-JADE is analyzed by twenty-five 30-dimensional CEC2005 test functions. The results indicate that the average performance of TPC-JADE is influenced by the value of σ , i.e., a large value of σ can provide better performance when compared with a small value of sigma. Additionally, TPC-JADE is employed to solve the dynamic optimization problem, the experimental results present that TPC-JADE is an effective and efficient tool in dealing with complex dynamic optimization problems.

Acknowledgements

This work was supported by National Natural Science Foundation of China (Nos. 61603244 and 41505001) and Fundamental Research Funds for the Central Universities (No. 222201717006).

References

- [1] R. Storn, K. Price. *Differential Evolution—A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces*, Berkeley, USA: ICSI, 1995.
- [2] F. Neri, V. Tirronen. Recent advances in differential evolution: A survey and experimental analysis. *Artificial Intelligence Review*, vol.33, no.1–2, pp.61–106, 2010. DOI: [10.1007/s10462-009-9137-2](https://doi.org/10.1007/s10462-009-9137-2).
- [3] S. Das, P. N. Suganthan. Differential evolution: A survey of the state-of-the-art. *IEEE Transactions on Evolutionary Computation*, vol.15, no.1, pp.4–31, 2011. DOI: [10.1109/TEVC.2010.2059031](https://doi.org/10.1109/TEVC.2010.2059031).
- [4] R. Gämperle, S. D. Müller, P. Koumoutsakos. A parameter study for differential evolution. In *Advances in Intelligent Systems, Fuzzy Systems, Evolutionary Computation*, Interlaken, Switzerland: WSEAS Press, pp.293–298, 2002.
- [5] R. Storn, K. Price. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, vol.11, no.4, pp.341–359, 1997. DOI: [10.1023/A:1008202821328](https://doi.org/10.1023/A:1008202821328).
- [6] J. Ronkkonen, S. Kukkonen, K. V. Price. Real-parameter optimization with differential evolution. In *Proceedings of IEEE Congress on Evolutionary Computation*, IEEE, Edinburgh, Scotland, pp.506–513, 2005. DOI: [10.1109/CEC.2005.1554725](https://doi.org/10.1109/CEC.2005.1554725).
- [7] A. K. Qin, V. L. Huang, P. N. Suganthan. Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE Transactions on Evolutionary Computation*, vol.13, no.2, pp.398–417, 2009. DOI: [10.1109/TEVC.2008.927706](https://doi.org/10.1109/TEVC.2008.927706).
- [8] J. Liu, J. Lampinen. A fuzzy adaptive differential evolution algorithm. *Soft Computing*, vol.9, no.6, pp.448–462, 2005. DOI: [10.1007/s00500-004-0363-x](https://doi.org/10.1007/s00500-004-0363-x).
- [9] J. Brest, S. Greiner, B. Bošković, M. Mernik, V. Zumer. Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems. *IEEE Transactions on Evolutionary Computation*, vol.10, no.6, pp.646–657, 2006. DOI: [10.1109/TEVC.2006.872133](https://doi.org/10.1109/TEVC.2006.872133).
- [10] J. Q. Zhang, A. C. Sanderson. JADE: Adaptive differential evolution with optional external archive. *IEEE Transactions on Evolutionary Computation*, vol.13, no.5, pp.945–958, 2009. DOI: [10.1109/TEVC.2009.2014613](https://doi.org/10.1109/TEVC.2009.2014613).
- [11] R. Mallipeddi, P. N. Suganthan, Q. K. Pan, M. F. Tasgetiren. Differential evolution algorithm with ensemble of parameters and mutation strategies. *Applied Soft Computing*, vol.11, no.2, pp.1679–1696, 2011. DOI: [10.1016/j.asoc.2010.04.024](https://doi.org/10.1016/j.asoc.2010.04.024).
- [12] Y. Wang, Z. X. Cai, Q. F. Zhang. Differential evolution with composite trial vector generation strategies and control parameters. *IEEE Transactions on Evolutionary Computation*, vol.15, no.1, pp.55–66, 2011. DOI: [10.1109/TEVC.2010.2087271](https://doi.org/10.1109/TEVC.2010.2087271).
- [13] Y. Wang, H. X. Li, T. W. Huang, L. Li. Differential evolution based on covariance matrix learning and bimodal distribution parameter setting. *Applied Soft Computing*, vol.18, pp.232–247, 2014. DOI: [10.1016/j.asoc.2014.01.038](https://doi.org/10.1016/j.asoc.2014.01.038).
- [14] Q. Q. Fan, X. F. Yan. Differential evolution algorithm with self-adaptive strategy and control parameters for P-xylene oxidation process optimization. *Soft Computing*, vol.19, no.5, pp.1363–1391, 2015. DOI: [10.1007/s00500-014-1349-y](https://doi.org/10.1007/s00500-014-1349-y).
- [15] Q. Q. Fan, X. F. Yan. Self-adaptive differential evolution algorithm with zoning evolution of control parameters and adaptive mutation strategies. *IEEE Transactions on Cybernetics*, vol.46, no.1, pp.219–232, 2016. DOI: [10.1109/TCYB.2015.2399478](https://doi.org/10.1109/TCYB.2015.2399478).
- [16] R. A. Sarker, S. M. Elsayed, T. Ray. Differential evolution with dynamic parameters selection for optimization problems. *IEEE Transactions on Evolutionary Computation*, vol.18, no.5, pp.689–707, 2014. DOI: [10.1109/TEVC.2013](https://doi.org/10.1109/TEVC.2013).

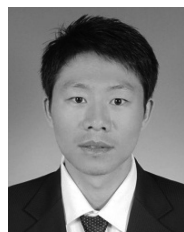
2281528.

- [17] S. Rahnamayan, H. R. Tizhoosh, M. M. A. Salama. Opposition-based differential evolution. *IEEE Transactions on Evolutionary Computation*, vol.12, no.1, pp.64–79, 2008. DOI: [10.1109/TEVC.2007.894200](https://doi.org/10.1109/TEVC.2007.894200).
- [18] W. Y. Gong, Z. H. Cai, Y. Wang. Repairing the crossover rate in adaptive differential evolution. *Applied Soft Computing*, vol.15, pp.149–168, 2014. DOI: [10.1016/j.asoc.2013.11.005](https://doi.org/10.1016/j.asoc.2013.11.005).
- [19] W. Y. Gong, Z. H. Cai, D. W. Liang. Adaptive ranking mutation operator based differential evolution for constrained optimization. *IEEE Transactions on Cybernetics*, vol.45, no.4, pp.716–727, 2015. DOI: [10.1109/TCYB.2014.2334692](https://doi.org/10.1109/TCYB.2014.2334692).
- [20] M. Yang, C. H. Li, Z. H. Cai, J. Guan. Differential evolution with auto-enhanced population diversity. *IEEE Transactions on Cybernetics*, vol.45, no.2, pp.302–315, 2015. DOI: [10.1109/TCYB.2014.2339495](https://doi.org/10.1109/TCYB.2014.2339495).
- [21] Q. Q. Fan, X. F. Yan, Y. L. Zhang. Auto-selection mechanism of differential evolution algorithm variants and its application. *European Journal of Operational Research*, to be published. DOI: [10.1016/j.ejor.2017.10.013](https://doi.org/10.1016/j.ejor.2017.10.013).
- [22] S. M. Guo, C. C. Yang. Enhancing differential evolution utilizing eigenvector-based crossover operator. *IEEE Transactions on Evolutionary Computation*, vol.19, no.1, pp.31–49, 2015. DOI: [10.1109/TEVC.2013.2297160](https://doi.org/10.1109/TEVC.2013.2297160).
- [23] J. H. Zhong, M. E. Shen, J. Zhang, H. S. H. Chung, Y. H. Shi, Y. Li. A differential evolution algorithm with dual populations for solving periodic railway timetable scheduling problem. *IEEE Transactions on Evolutionary Computation*, vol.17, no.4, pp.512–527, 2013. DOI: [10.1109/TEVC.2012.2206394](https://doi.org/10.1109/TEVC.2012.2206394).
- [24] S. Biswas, S. Kundu, S. Das. Inducing niching behavior in differential evolution through local information sharing. *IEEE Transactions on Evolutionary Computation*, vol.19, no.2, pp.246–263, 2015. DOI: [10.1109/TEVC.2014.2313659](https://doi.org/10.1109/TEVC.2014.2313659).
- [25] R. Tanabe, A. S. Fukunaga. Improving the search performance of SHADE using linear population size reduction. In *Proceedings of IEEE Congress on Evolutionary Computation*, IEEE, Beijing, China, pp.1658–1665, 2014. DOI: [10.1109/CEC.2014.6900380](https://doi.org/10.1109/CEC.2014.6900380).
- [26] Y. L. Li, Z. H. Zhan, Y. J. Gong, W. N. Chen, J. Zhang, Y. Li. Differential evolution with an evolution path: A deep evolutionary algorithm. *IEEE Transactions on Cybernetics*, vol.45, no.9, pp.1798–1810, 2015. DOI: [10.1109/TCYB.2014.2360752](https://doi.org/10.1109/TCYB.2014.2360752).
- [27] S. M. Guo, C. C. Yang, P. H. Hsu, J. S. H. Tsai. Improving differential evolution with a successful-parent-selecting framework. *IEEE Transactions on Evolutionary Computation*, vol.19, no.5, pp.717–730, 2015. DOI: [10.1109/TEVC.2014.2375933](https://doi.org/10.1109/TEVC.2014.2375933).
- [28] H. Lu, J. Yin, Y. X. Yuan, J. H. Wang, Y. Z. Lin, X. L. Wang. Energy consumption analysis of sludge transport pipeline system based on GA-DE hybrid algorithm. *Journal of Chemical Engineering of Japan*, vol.47, no.8, pp.621–627, 2014. DOI: [10.1252/jcej.13we233](https://doi.org/10.1252/jcej.13we233).
- [29] X. H. Qiu, Y. T. Hu, B. Li. Sequential fault diagnosis using an inertial velocity differential evolution algorithm. *International Journal of Automation and Computing*, Online First. DOI: [10.1007/s11633-016-1008-0](https://doi.org/10.1007/s11633-016-1008-0).
- [30] G. H. Lin, J. Zhang, Z. H. Liu. Hybrid particle swarm optimization with differential evolution for numerical and engineering optimization. *International Journal of Automation and Computing*, vol.15, no.1, pp.103–114, 2018. DOI: [10.1007/s11633-016-0990-6](https://doi.org/10.1007/s11633-016-0990-6).
- [31] H. T. Ye, Z. Q. Li. PID neural network decoupling control based on hybrid particle swarm optimization and differential evolution. *International Journal of Automation and Computing*, Online First. DOI: [10.1007/s11633-015-0917-7](https://doi.org/10.1007/s11633-015-0917-7).
- [32] Q. Q. Fan, X. F. Yan, Y. Xue. Prior knowledge guided differential evolution. *Soft Computing*, vol.21, no.22, pp.6841–6858, 2017. DOI: [10.1007/s00500-016-2235-6](https://doi.org/10.1007/s00500-016-2235-6).
- [33] P. N. Suganthan, N. Hansen, J. J. Liang, K. Deb, Y. P. Chen, A. Auger, S. Tiwari. Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization, KanGAL Report 2005005, Nanyang Technological University, Singapore, 2005.
- [34] J. J. Liang, B. Y. Qu, P. N. Suganthan, Q. Chen. Problem Definitions and Evaluation Criteria for the CEC 2015 Competition on Learning-based Real-parameter Single Objective Optimization, Technical Report 201411A, Computational Intelligence Laboratory, Zhengzhou University, China, 2014.
- [35] J. R. Banga, E. Balsa-Canto, C. G. Moles, A. A. Alonso. Dynamic optimization of bioprocesses: Efficient and robust numerical strategies. *Journal of Biotechnology*, vol.117, no.4, pp.407–419, 2005. DOI: [10.1016/j.jbiotec.2005.02.013](https://doi.org/10.1016/j.jbiotec.2005.02.013).
- [36] B. Srinivasan, S. Palanki, D. Bonvin. Dynamic optimization of batch processes: I. Characterization of the nominal solution. *Computers & Chemical Engineering*, vol.27, no.1, pp.1–26, 2003. DOI: [10.1016/S0098-1354\(02\)00116-3](https://doi.org/10.1016/S0098-1354(02)00116-3).
- [37] R. Bellman. Dynamic programming and Lagrange multipliers. *Proceedings of the National Academy of Sciences of the United States of America*, vol.42, no.10, pp.767–769, 1956. DOI: [10.1073/pnas.42.10.767](https://doi.org/10.1073/pnas.42.10.767).
- [38] R. Luus. On the application of iterative dynamic programming to singular optimal control problems. *IEEE Transactions on Automatic Control*, vol.37, no.11, pp.1802–1806, 1992. DOI: [10.1109/9.173155](https://doi.org/10.1109/9.173155).
- [39] W. H. Ray, J. Szekeley. *Process Optimization with Applications in Metallurgy and Chemical Engineering*, New York, USA: John Wiley & Sons, 1973.
- [40] A. E. Bryson. *Dynamic Optimization*, Menlo Park, USA: Addison Wesley Longman, 1999.
- [41] D. Sarkar, J. M. Modak. Optimisation of fed-batch bioreactors using genetic algorithms. *Chemical Engineering Science*, vol.58, no.11, pp.2283–2296, 2003. DOI: [10.1016/S0009-2509\(03\)00095-2](https://doi.org/10.1016/S0009-2509(03)00095-2).
- [42] R. W. H. Sargent, G. R. Sullivan. The development of an efficient optimal control package. In *Proceedings of the 8th IFIP Conference on Optimization Techniques*, Springer, Würzburg, Germany, pp.158–168, 1978. DOI: [10.1007/BFb0006520](https://doi.org/10.1007/BFb0006520).
- [43] K. V. Price, R. M. Storn, J. A. Lampinen. *Differential*

Evolution: A Practical Approach to Global Optimization, Berlin, Germany: Springer Heidelberg, 2005. DOI: [10.1007/3-540-31306-0](https://doi.org/10.1007/3-540-31306-0).

- [44] S. Das, A. Konar, U. K. Chakraborty. Two improved differential evolution schemes for faster global search. In *Proceedings of the Conference on Genetic and Evolutionary Computation*, ACM, Washington DC, USA, pp. 991–998, 2005. DOI: [10.1145/1068009.1068177](https://doi.org/10.1145/1068009.1068177).
- [45] J. Montgomery, S. Chen. An analysis of the operation of differential evolution at high and low crossover rates. In *Proceedings of IEEE Congress on Evolutionary Computation*, IEEE, Barcelona, Spain, 2010. DOI: [10.1109/CEC.2010.5586128](https://doi.org/10.1109/CEC.2010.5586128).
- [46] F. Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, 1945. DOI: [10.2307/3001968](https://doi.org/10.2307/3001968).
- [47] M. Friedman. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association*, vol. 32, no. 200, pp. 675–701, 1937. DOI: [10.2307/2279372](https://doi.org/10.2307/2279372).
- [48] J. A. Roubos, C. D. De Gooijer, G. Van Straten, A. J. B. Van Bortel. Comparison of optimization methods for fed-batch cultures of hybridoma cells. *Bioprocess Engineering*, vol. 17, no. 2, pp. 99–102, 1997. DOI: [10.1007/s004490050360](https://doi.org/10.1007/s004490050360).
- [49] J. Lee, W. F. Ramirez. Optimal fed-batch control of induced foreign protein production by recombinant bacteria. *AIChE Journal*, vol. 40, no. 5, pp. 899–907, 1994. DOI: [10.1002/aic.690400516](https://doi.org/10.1002/aic.690400516).
- [50] Q. Q. Fan, Z. M. Lü, X. F. Yan, M. J. Guo. Chemical process dynamic optimization based on hybrid differential evolution algorithm integrated with Alopex. *Journal of Central South University*, vol. 20, no. 4, pp. 950–959, 2013. DOI: [10.1007/s11771-013-1570-3](https://doi.org/10.1007/s11771-013-1570-3).
- [51] J. A. Roubos, G. Van Straten, A. J. B. Van Bortel. An evolutionary strategy for fed-batch bioreactor optimization; concepts and performance. *Journal of Biotechnology*, vol. 67, no. 2-3, pp. 173–187, 1999. DOI: [10.1016/S0168-1656\(98\)00174-6](https://doi.org/10.1016/S0168-1656(98)00174-6).
- [52] D. Sarkar, J. M. Modak. ANNSA: A hybrid artificial neural network/simulated annealing algorithm for optimal control problems. *Chemical Engineering Science*, vol. 58, no. 14, pp. 3131–3142, 2003. DOI: [10.1016/S0009-2509\(03\)00168-4](https://doi.org/10.1016/S0009-2509(03)00168-4).
- [53] B. Zhang, D. Z. Chen, W. X. Zhao. Iterative ant-colony algorithm and its application to dynamic optimization of chemical process. *Computers & Chemical Engineering*, vol. 29, no. 10, pp. 2078–2086, 2005. DOI: [10.1016/j.compchemeng.2005.05.020](https://doi.org/10.1016/j.compchemeng.2005.05.020).
- [54] Q. Q. Fan, X. H. Wang, X. F. Yan. Harmony search algorithm with differential evolution based control paramet-

er co-evolution and its application in chemical process dynamic optimization. *Journal of Central South University*, vol. 22, no. 6, pp. 2227–2237, 2015. DOI: [10.1007/s11771-015-2747-8](https://doi.org/10.1007/s11771-015-2747-8).

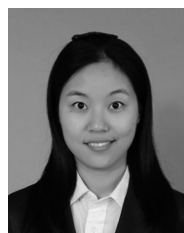


Qin-Qin Fan received the B.Sc. degree in automation from Institute of Technology, China in 2007, the M.Sc. degree in control science and engineering from East China University of Science and Technology, China in 2011, and the Ph.D. degree control science and engineering from East China University of Science and Technology, China in 2015, respectively. He is currently a lecturer with Shanghai Maritime University, China.

His research interests include differential evolution algorithm, particle swarm optimization, constrained optimization, multi-objective optimization, and their real-world applications.

E-mail: forever123fan@163.com (Corresponding author)

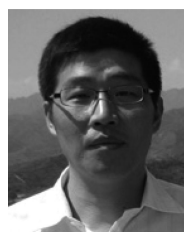
ORCID iD: 0000-0002-8603-5161



Yi-Lian Zhang received the Ph.D. degree in control science and engineering from East China University of Science and Technology, China in 2015. She is now a lecturer with Shanghai Maritime University, China.

Her research interests include networked control systems, set-membership filtering and evolutionary computation.

E-mail: zyl1030@126.com

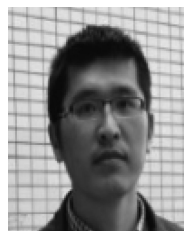


Xue-Feng Yan received the Ph.D. degree in control science and engineering from Zhejiang University, China. He is now a professor of East China University of Science and Technology, China.

His research interests include complex chemical process modeling, optimizing and controlling, process monitoring, fault diagnosis and intelligent information processing.

processing.

E-mail: xfyang@ecust.edu.cn



Zhi-Huan Wang received the B.Sc. degree in mechanical manufacture and automation from Harbin Institute of Technology, China in 2002, and the M.Sc. degree in information management and information system from Monash University, Australia in 2009.

His research interests include big data, evolutionary computation and intelligent

information processing.

E-mail: zhwang@shmtu.edu.cn