Computer Modeling in
Engineering & Sciences

Tech Science Press

DOI: 10.32604/cmes.2023.027055

ARTICLE

An Enhanced Adaptive Differential Evolution Approach for Constrained Optimization Problems

Wenchao Yi, Zhilei Lin, Yong Chen, Zhi Pei[*] and Jiansha Lu

College of Mechanical Engineering, Zhejiang University of Technology, Hangzhou, 310023, China

*Corresponding Author: Zhi Pei. Email: peizhi@zjut.edu.cn

Received: 11 October 2022 Accepted: 25 November 2022

ABSTRACT

Effective constrained optimization algorithms have been proposed for engineering problems recently. It is common to consider constraint violation and optimization algorithm as two separate parts. In this study, a *p*best selection mechanism is proposed to integrate the current mutation strategy in constrained optimization problems. Based on the improved *p*best selection method, an adaptive differential evolution approach is proposed, which helps the population jump out of the infeasible region. If all the individuals are infeasible, the top 5% of infeasible individuals are selected. In addition, a modified truncated ε-level method is proposed to avoid trapping in infeasible regions. The proposed adaptive differential evolution approach with an improved ε constraint process mechanism (IεJADE) is examined on CEC 2006 and CEC 2010 constrained benchmark function series. Besides, a standard IEEE-30 bus test system is studied on the efficiency of the IεJADE. The numerical analysis verifies the IεJADE algorithm is effective in comparison with other effective algorithms.

KEYWORDS

*p*best selection mechanism; adaptive differential evolution; ε constrained method

1 Introduction

Numerous real-world engineering optimization problems, such as the milling process [1] and design problems of the car sides [2], are typical constrained optimization problems (COPs). A set of constraints and the objective function are needed to meet at the same time. The mathematical model can be presented in the following way:

$$\begin{aligned}
&\min f\left(\vec{X}\right)\\
&\textit{subject to}\\
&g_l\left(\vec{X}\right) \le 0, l = 1, 2, \ldots, Q\\
&h_k\left(\vec{X}\right) \le 0, k = 1, 2, \ldots, K\\
&L_j \le x_j \le U_j, j = 1, 2, \ldots, D
\end{aligned} \tag{1}$$

This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

where $\vec{X} = [x_0, x_1, \ldots, x_D]^T$ is the D dimensional vector of decision variables. $f\left(\vec{X}\right)$ denotes the objective function. $g_l\left(\vec{X}\right)$ stands for the l-th inequality constraints, $h_k\left(\vec{X}\right)$ represents the k-th equality constraints. L_j and U_j are used to specify the lower and upper bounds of the j-th decision variable.

In the literature, traditional mathematical programming-based approaches and computational intelligence algorithms are applied to tackle COPs. Traditional methods, such as the gradient-based approaches, often require a priori knowledge of the mathematical properties of the model. While computational intelligence methods do not rely on that a priori knowledge, but can still present more robust and better solutions than traditional methods.

Evolutionary algorithms, e.g., the differential evolution (DE) algorithm, are effective ways to deal with the COPs. DE algorithm is effectively used as the search engine. Since the first proposal of Storn et al. [3], many scholars have contributed continuously to improve it. Zhang et al. [4] introduced the adaptive DE algorithm, JADE, by proposing an effective mutation strategy and adaptive control parameter generation method. Gong et al. proposed a series of essential and effective improvements to JADE [5]. A strategy adaption mechanism and a crossover rate repairing technique are presented. The researches show that the improvements of JADE are competitive compared with other algorithms.

Combined with the constraint handling techniques, DE can be utilized in COPs [6–9]. The most effective ones are penalty function method [10], multi-objective-based method [11], ε constraint handling technique [12], stochastic ranking (SR) [13], and so on. As a classical constraint handling technique, penalty function method requires appropriate penalty parameters to get better results. The superiority of feasible solutions (SF) [14] is embedded in the selection phase, which gives priority to ensuring all solutions are feasible, and then optimizes the objective function values. However, promising infeasible solutions should be retained at the beginning of the iteration to help the population move to a better region. Fu et al. [9] proposed the hyperspace dynamic region to store infeasible solutions, which can improve the performance of the algorithm.

ε constraint handling technique is a promising method proposed nowadays, which accepts the promising constraint violation individuals at first, and then accepts those feasible solutions at the end of the evolution process. Takahama et al. [12] first introduced the ε constraint method into DE algorithm (εDE). The feasible solutions can be obtained quickly, and better objective values are achieved as well. Then they brought in a revised εDE via the kernel regression-based approximation [15]. Unfortunately, this method is time-consuming due to its approximation procedure. Wang et al. [16] integrated both SF and ε constraint handling technique into composite differential evolution. Its comprehensive performance is better than those using only one constraint handling method. Zhang et al. [17] presented an adaptive ε control heuristic method in place of using exponential function to control ε value. Stanovov et al. [18] integrated a selective pressure technique in DE, which takes function values, constraint violations measured by ε-constraint level into consideration simultaneously. Aiming to the multipath estimation in non-Gaussian noise problem, Ni et al. [19] integrated ε-constrained method into differential evolution. Duc Nguyen et al. [20] integrated differential evolution in Adaptive-Network-based Fuzzy Inference System. The experiment results prove the method is better than Reduced Error Pruning Trees and Decision Stump. Yi et al. [21] presented an improved version of ε-constrained method with adaptive differential evolution.

ε constraint method can adjust the feasible region adaptively in a more effective way. And the ε-level control method plays an important role in balancing exploration and exploitation. Hence, an improved version of ε constraint method is proposed in this paper.

The main purpose of the paper is to design an effective constraint technique and optimization algorithm for COPs. Therefore, the following two innovations are made:

1) First, the *p*best selection mechanism is proposed. Rather than selecting the individuals with respect to the objective values, the individuals are chosen based on constraint violation and fitness.
2) Second, an improved ε constraint method is proposed. The reasonable setting of ε is important. Hence, a novel truncated setting is proposed to ensure that.

The framework of the paper can be summarized as below. Section 2 presented the proposed algorithm, with an improved top *p*best individual selection mechanism in JADE for COPs. In Section 3, the improved ε constrained method is presented. In Section 4, the proposed algorithm is compared with several DE variants via benchmark testing problems. In Section 5, the standard IEEE-30 bus test system is used to test the efficiency of the proposed algorithm. And Section 6 concludes the paper.

2 Adaptive Differential Evolution for COPs

An improved JADE that tackles the COPs is proposed. The *p*best selection mechanism will be introduced.

2.1 DE/Current-to-pbest/2 Strategy

DE/current-to-*p*best/2 strategy is:

$$v_{i,G+1} = x_{i,G} + F_i \left(x^{p}_{best,G} - x_{i,G} \right) + F_i \left(x_{r1,G} - x_{r2,G} + x_{r3,G} - x_{r4,G} \right) \tag{2}$$

where x_{r3}, x_{r4} are two randomly selected individuals in population.

In JADE, the archive is used to collect unsuccessful solutions, and it can serve to improve the global search capability. The difference vector can point to the better area by archive set. The archive can be helpful in exploitation, which can also be harmful in investing unpromising areas. Emphasis on the exploration, the archive is not included in our paper. To give a vivid description on the proposed strategy, Function 8 with two-dimensional variables in CEC 2006 benchmark [22] is used as an example to illustrate our idea. The objective function, constraint violation and the mutation vector generation with the proposed strategy diagram are drawn as Figs. 1 and 2.

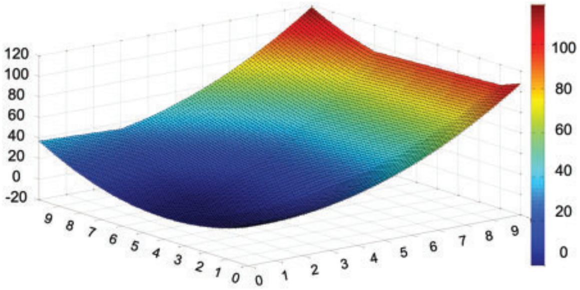


Figure 1: Constraint violation of F08 in CEC 2006

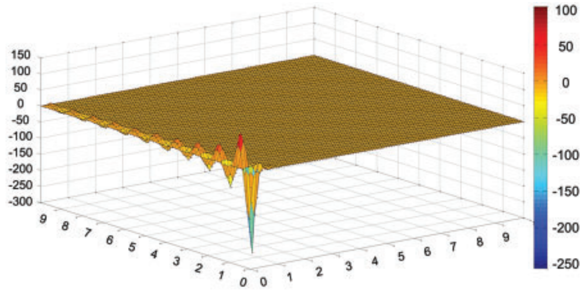


Figure 2: Objective function value of F08 in CEC 2006

We can see from Fig. 3, the proposed strategy can guide the target vector moving towards the promising area, and moving away from the more constraint violation area. In this way, more exploration can be made within the promising area.

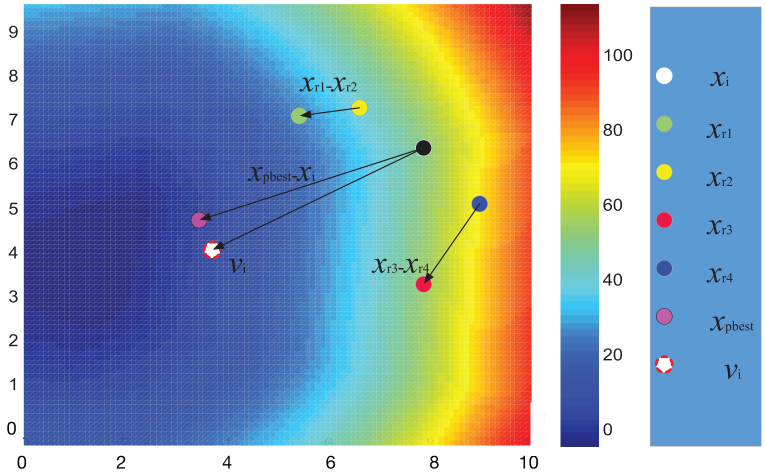


Figure 3: Illustration of the proposed strategy in F08

2.2 Parameter Adaption Strategy

Control parameters F_i and CR_i are from various distribution and mean values. In this way, they can move towards the promising areas.

The CR_i generation in JADE [4] is given as:

$$CR_i = randn_i\left(\mu_{CR}, 0.1\right) \tag{3}$$

where μ_{CR} is mean value; 0.1 is standard deviation; $randn_i\left(x, y\right)$ indicates a random value sampled from a uniform distribution.

The F_i is similar to CR_i generation with following formula:

$$F_i = randc_i\left(\mu_F, 0.1\right) \tag{4}$$

where $randc_i\left(x, y\right)$ is based on Cauchy distribution with location x and scale y.

$$\begin{cases} F_{i,\mathrm{G}} = rand \\ CR_{i,G} = rand \end{cases}, \forall i, G \tag{5}$$

3 ε Constrained Algorithm

Fitness and constraint violation are both vital in COPs. ε constrained algorithm is effective in dealing with the two indexes. The improved ε setting will be given in the following part.

3.1 ε Comparison Strategy

In-line equations/expressions are embedded into the paragraphs of the text. For example, $\mathrm{E} = mc^2$. In-line equations or expressions should not be numbered and should use the same/similar font and size as the main text.

Constraint violation $\phi(x)$ is the maximum or the sum of all constraints.

$$\phi = \max\left\{\max\{0, g_i(x)\}, \left|h_j(x)\right|\right\}, i \in \{1, \dots, q\}, j \in \{1, \dots, m\} \tag{6}$$

$$\phi(x) = \sum_{i=1}^{q} \|\max\{0, g_i(x)\}\|^p + \sum_{j=q+1}^{m} \left\|\left|h_j(x)\right|\right\|^p \tag{7}$$

where $\| \|^p$ represents p-norm. Without loss of generality, the maximum of all the constraints is applied and is denoted as $\phi(x)$ in the proposed algorithm.

Based on the concept of constraint violation, COPs can be presented as follows:

$$\begin{aligned} &\min f(x) \\ &s.t.\ \phi(\mathrm{x}) \leq 0 \end{aligned} \tag{8}$$

In usual comparison strategy, x is defined as feasible when $\phi(x) = 0$. The comparison on the ε level yields the compact constraints by tuning from 0 to parameter ε. That is, x is named ε-feasible, if $\phi(x)$ is not bigger than ε. Then, comparison can be presented by the formula below:

$$(f_1, \phi_1) \leq_\varepsilon (f_2, \phi_2) \Leftrightarrow \begin{cases} f_1 \leq f_2, \ if\ \phi_1, \phi_2 \leq \varepsilon \\ f_1 \leq f_2, \ if\ \phi_1 = \phi_2 \\ \phi_1 \leq \phi_2, \ otherwise \end{cases} \tag{9}$$

where (f_1, ϕ_1) is the objective value and constraint violation of x_1.

3.2 ε-Level Setting

ε-level setting is vital in ε constrained method [21,23]. Above researches show that ε should be larger at first and decrease along with constraint violation.

Static ε control method was firstly proposed [12] and it uses exponent function control the decrease of ε-level:

$$\begin{cases} \varepsilon(0) = \phi(x_\theta) \\ \varepsilon(t) = \begin{cases} \varepsilon(0)\left(1 - \frac{G}{T}\right)^{cp}, 0 < G < T \\ 0, \qquad\qquad\qquad G \geq T \end{cases} \end{cases} \tag{10}$$

where x_θ means the θth constraint violation value among the population. During the iteration, along with the increase of G, ε-level is reduced to 0; cp and T are control parameters;

Truncated ε-level method uses the maximum and minimum constraint violation in the population to calculate ε-level:

$$\varepsilon = \begin{cases} 0, & \varphi_{ap\times NP} = 0 \\ [ap \times \varphi_{\min}, ap \times \varphi_{\max}], & else \end{cases} \tag{11}$$

where $\varphi_{\min}$ and $\varphi_{\max}$ indicate the least and most constraint violation of the current population, respectively; NP is the size of the population; ap is control parameter. So, in truncated ε-level method, if a defined percent of the population is feasible, ε is assumed to be 0; Otherwise, a random value is generated as based on $\varphi_{\min}$ and $\varphi_{\max}$.

Using static ε control method, the ε-level is related to the initial constraint violation during the generation, excluding current information of population. If the initial violation value is too large, it will slow down the convergence speed and lean to be stuck in the local optimum within the infeasible region. Although the truncated ε-level method exploits the information of the present population, for complicate equality constraints, the strategy may lead the population trapping into the infeasible region.

In this paper, an improved ε-level setting is presented as follows:

$$\begin{cases} \varepsilon'(t) = \overline{\varepsilon}\left(1 - \frac{G}{T}\right)^{cp}, \; \mathit{if}\ 0 < G < T \\ \varepsilon(t) = r\left[\min\left(\varphi_{\max}, \varepsilon'(t)\right), \max\left(\varphi_{\max}, \varepsilon'(t)\right)\right] \end{cases} \tag{12}$$

A proper setting of $\overline{\varepsilon}$ is proposed as follows:

$$\overline{\varepsilon} = \begin{cases} \varepsilon(0), \; \mathit{if}\ \varepsilon(0) \le \varepsilon^* \\ \varepsilon(t), \; \mathit{else} \end{cases} \tag{13}$$

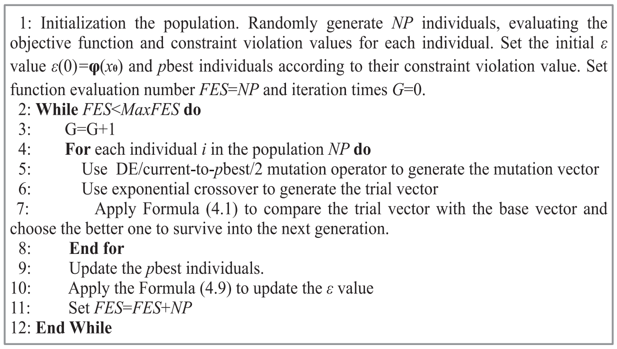
A revised truncated strategy is implemented. $\varphi_{\min}$ is replaced by $\varepsilon'(t)$. ε is generated between $\varphi_{\max}$ and $\varepsilon'(t)$, which multiplied by an important coefficient r. In the proposed strategy, the following formula is used:

$$r = 0.9 + 0.1 * \mathrm{rand} \tag{14}$$

3.3 pbest Selection Method

Fitness value and feasibility are both vital in COPs. Feasibility often is more important than fitness value. A selection strategy for *p*best individuals with respect to the constraint violation is presented. If two individuals are both infeasible, the one with less constraint violation is better. If they are both feasible, their objective function values are compared instead.

Based on the above analysis, the pseudocode of the proposed algorithm is shown in Fig. 4.



```
Initialization the population. Randomly generate $NP$ individuals, evaluating the objective function and constraint violation values for each individual. Set the initial $\varepsilon$ value $\varepsilon(0)=\boldsymbol{\varphi}(x_0)$ and pbest individuals according to their constraint violation value. Set function evaluation number $FES=NP$ and iteration times $G=0$.
While $FES<MaxFES$ do
    G=G+1
    For each individual $i$ in the population $NP$ do
        Use DE/current-to-pbest/2 mutation operator to generate the mutation vector
        Use exponential crossover to generate the trial vector
        Apply Formula (4.1) to compare the trial vector with the base vector and choose the better one to survive into the next generation.
    End for
    Update the pbest individuals.
    Apply the Formula (4.9) to update the $\varepsilon$ value
    Set $FES=FES+NP$
End While
```

Figure 4: Pseudocode of the proposed improved adaptive differential evolution with modified ε constrained method

4 Experiment Results

To examine the performance quantitatively, two benchmark function sets are used: CEC 2006 and CEC 2010, where $D = 10, 30$ are analyzed. For simplicity's sake, the detailed forms of the test functions are omitted. The details can be referred in Liang et al. [22] and Mallipeddi et al. [24], respectively.

It is worth mentioning that the CEC 2010 benchmark functions with 10D have twelve functions which feasible region ratio less than 10E−06, and thirteen functions with 30D benchmark functions. Referring to the feasible region ratio, it is more difficult to solve the CEC 2010 benchmark functions than CEC 2006 series.

4.1 Parameter Setting

For the proposed algorithm, the parameters are set as below:

- population size: 80 (CEC 2006), 20 (10D, CEC 2010), 50 (30D, CEC 2010);
- crossover rate: $CR_i = randn_i\,(\mu_{CR}, 0.1)$;
- scaling factor: $F_i = randc_i(\mu_F, 0.1)$;
- ε setting control parameter: $cp = 5$; $r \in [0.9, 1]$;
- maximum number of fitness evaluations: 500,000; $D * 20{,}000$ (CEC 2010);

where μ_{CR} indicates mean value. 0.1 is standard deviation; $randn_i\,(x, y)$ is randomly generated from the uniformly distribution. μ_F indicates the expected value, and the standard deviation is assumed to be 0.1; $randc_i\,(x, y)$ stands for Cauchy distribution with location x and scale y. D in the maximum number of fitness evaluations stands for dimension in CEC 2010, which are 10 and 30, respectively.

4.2 Performance on CEC 2006

Three state-of-the-art algorithms APF-GA [25], Cultural CPSO [26] and AIS [27] are used as the baselines to measure the performance of the proposed IεJADE method. Since all the three algorithms cannot guarantee feasibility for functions G20 and G22, the comparisons on these two benchmark functions are omitted. The experimental results of APF-GA, Cultural CPSO and AIS are from Zhang et al. [23].

We can conclude from Table 1, the APF-GA method converges to the most desirable results in 16 functions, which are G01, G03-04, G06-09, G11-14, G16, G18 and G24 in terms of the "Best" and "Mean" indexes. The Cultural PSO algorithm can attain the best outcome in 17 out of 22 functions, which are G01, G03-12, G14, G16, G18, G21 and G23-24 referring to the "Best" and "Mean" indexes. 7 out of 20 functions can obtain the most desirable results for AIS algorithm with respect to the "Best" and "Mean" indexes, which are G01, G08, G11-12, G15-16 and G19. The proposed IεJADE algorithm secure the best solution in 13 out of 22 functions, which are G02, G04-G14 and G16. The proposed IεJADE algorithm rivals the APF-GA and Cultural-CPSO algorithm, and outperforms the AIS algorithm with respect to the overall performance. In particular, APF-GA algorithm attains the best outcome in G05 and G10 in terms of the "Best" index without a tie. Similarly, Cultural-PSO reaches the best results in G02 and G03, AIS in G02, G04-05, G07, G09-10, G14, G18, G21 and G23, IεJADE in G03, G15, G17-18, G21 and G23. As for the "Best" index, the IεJADE algorithm shows superior performance than APF-GA and Cultural-CPSO algorithms. To sum up, the proposed IεJADE is an effective algorithm in solving the CEC 2006 benchmark functions.

Table 1: Statistical results for solving 22 benchmark functions from CEC 2006

CEC 2006		APF-GA	Cultural CPSO	AIS	IεJADE
G01	Best	**−15.0000**	**−15.0000**	**−15.0000**	**−15**
	Mean	**−15.0000**	**−15.0000**	**−15.0000**	**−15**
	std	0.0E+00	0.0E+00	0.0E+00	0.0E+00
G02	Best	−0.803601	**−0.8036191**	**−0.803619**	**−0.803619**
	Mean	−0.803518	−0.8016532	−0.802193	**−0.803619**
	std	1.00E−04	4.68E−03	5.19E−10	1.6E−06
G03	Best	**–1.001**	**−1.0005**	**−1.0005**	**−1.0005**
	Mean	**–1.001**	**−1.0005**	**−1.0005**	−1.0004
	std	0.0E+00	3.68E−13	1.77E−11	6.7E−04
G04	Best	**−30665.539**	**−30665.5387**	**−30665.53867**	**−30665.53867**
	Mean	**−30665.539**	**−30665.5387**	−30665.53866	**−30665.53867**
	std	1.00E−04	1.79E−16	3.69E−13	3.7E−12
G05	Best	**5126.497**	**5126.4967**	**5126.4967**	**5126.4967**
	Mean	5127.5423	**5126.4967**	5126.4981	**5126.4967**
	std	1.43E+00	6.09E−12	1.7E−02	2.8E−12
G06	Best	**−6961.814**	**−6961.8139**	−6961.81385	**−6961.813875**
	Mean	**−6961.814**	**−6961.8139**	−6961.81385	**−6961.813875**
	std	0.0E+00	3.81E−11	1.9E−12	0.0E+00
G07	Best	**24.3062**	**24.3062**	**24.306209**	**24.306209**
	Mean	**24.3062**	**24.3062**	24.35572	**24.306209**
	std	0.0E+00	1.37E−12	8.2E−03	1.0E−11
G08	Best	**−0.0958250**	**−0.09582504**	**−0.0958250**	**−0.095825**
	Mean	**−0.0958250**	**−0.09582504**	**−0.0958250**	**−0.095825**
	std	0.0E+00	7.81E−11	0.0E+00	1.5E−17
G09	Best	**680.630**	**680.630057**	**680.630057**	**680.630057**
	Mean	**680.630**	**680.630057**	680.650308	**680.630057**
	std	0.0E+00	5.88E−17	1.2E−08	3.2E−13
G10	Best	**7049.24802**	**7049.2480**	**7049.248020**	**7049.24802**
	Mean	7077.6821	**7049.2480**	7049.570318	**7049.24802**
	std	5.12E+01	4.68E−17	4.5E−04	2.1E−06
G11	Best	**0.7499**	**0.7499**	**0.749999**	**0.7499**
	Mean	**0.7499**	**0.7499**	**0.749999**	**0.7499**
	std	0.0E+00	4.68E−17	1.4E−08	1.1E−16
G12	Best	**−1.0000**	**−1.0000**	**−1.0000**	**−1**
	Mean	**−1.0000**	**−1.0000**	**−1.0000**	**−1**
	std	0.0E+00	1.76E−14	0.0E+00	0.0E+00

(Continued)

Table 1 (continued)

CEC 2006		APF-GA	Cultural CPSO	AIS	IεJADE
G13	Best	**0.053942**	**0.053941514**	0.05393852	**0.053941514**
	Mean	**0.053942**	0.053941519	0.05394037	**0.053941514**
	std	0.0E+00	1.54E−07	7.8E−10	1.6E−17
G14	Best	**−47.76479**	**−47.7649**	**−47.764888**	**−47.764888**
	Mean	**−47.76479**	**−47.7649**	**−47.764881**	**−47.764888**
	std	1.00E−04	6.78E−11	1.0E−12	5.0E−13
G15	Best	**961.71502**	**961.715022**	**961.71502**	**961.71502**
	Mean	**961.71502**	**961.715022**	**961.71502**	963.72618∗
	std	0.00E+00	2.66E−16	0.0E+00	3.0E+00
G16	Best	**−1.905155**	**−1.905155**	**−1.905155**	**−1.905155**
	Mean	**−1.905155**	**−1.905155**	**−1.905155**	**−1.905155**
	std	0.00E+00	3.96E−13	0.0E+00	4.5E−16
G17	Best	8853.5398	8853.5397	8853.53967	**8853.53387**
	Mean	8888.4876	**8853.5397**	**8853.53967**	8905.09808
	std	29.0347	1.53E−11	1.9E−09	8.5E+01
G18	Best	**−0.866025**	**−0.86602540**	**−0.8660254**	**−0.8660254**
	Mean	**−0.865925**	**−0.86602540**	−0.8660253	−0.8367392
	std	1.00E−04	8.09E−14	1.3E−15	1.0E−01
G19	Best	**32.655593**	**32.6556**	**32.655592**	**32.655593**
	Mean	**32.655593**	**32.6556**	**32.655592**	**32.655593**
	std	0.00E+00	5.91E−12	0.0E+00	1.2E−08
G21	Best	196.63301	**193.7245**	**193.72451**	**193.72451**
	Mean	199.51581	**193.7245**	196.72451	240.87671
	std	2.3565	4.65E−05	1.1E+00	6.4E+01
G23	Best	−399.7624	**−400.0551**	**−400.05509**	**−400.0551**
	Mean	−394.7627	**−400.0551**	−399.87432	−385.5666
	std	3.8656E+00	6.23E−11	2.0E+00	6.0E+01
G24	Best	**−5.508013**	**−5.5080**	**−5.5080132**	**−5.5080133**
	Mean	**−5.508013**	**−5.5080**	**−5.5080132**	**−5.5080133**
	std	0.00E+00	5.88E−15	0.0E+00	9.1E−16

4.3 Performance on CEC 2010

For comprehensive comparison, AIS and εDEg are chosen to compare with IεJADE. εDEg [28] is ranked first in the competition at CEC 2010, while the AIS algorithm is recently proved to be effective for COPs. And the best outcome of all the three algorithms is written in bold. The experimental results of CEC 2010 on 10D and 30D are given in Tables 2 and 3, respectively.

Table 2: Statistical results for solving 18 benchmark functions from CEC 2010 at 10D

CEC 2010 10D		Best	Median	Mean	Worst	Std
C01	IεJADE	**−0.74731**	**−0.74731**	−0.74704	−0.74056	1.35E−03
	AIS	**−0.74731**	**−0.74731**	**−0.74705**	**−0.74357**	1.30E−03
	εDEg	**−0.74731**	**−0.74731**	−0.74704	−0.74055	1.32E−03
C02	IεJADE	**−2.27771**	−2.1754	−2.1839	−2.1134	4.39E−02
	AIS	**−2.27771**	**−2.2755**	**−2.2748**	**−2.2709**	2.00E−03
	εDEg	**−2.27771**	−2.2695	−2.2588	−2.1744	2.38E−02
C03	IεJADE	8.8756E+00	8.8756E+00	8.8756E+00	8.8756E+00	1.57E−10
	AIS	**0.00E+00**	1.1772E−10	3.7472E−09	2.1342E−08	4.81E−04
	εDEg	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	0.00E+00
C04	IεJADE	**−1.000E−05**	**−1.000E−05**	**−1.000E−05**	**−1.000E−05**	3.55E−16
	AIS	**−1.000E−05**	−9.9867E−06	−9.9712E−06	−9.9322E−06	4.28E−03
	εDEg	−9.9923E−06	−9.9773E−06	−9.9185E−06	−9.2823E−06	1.54E−07
C05	IεJADE	−483.6106	**−483.6106**	**−483.6106**	**−483.6106**	3.48E−13
	AIS	**−483.6110**	−483.5293	−479.9600	−460.1242	6.3E+00
	εDEg	−483.6106	**−483.6106**	**−483.6106**	**−483.6106**	3.89E−13
C06	IεJADE	−577.1759	−577.1759	−577.1759	−577.1759	9.28E−14
	AIS	**−579.9984**	**−579.9843**	**−579.9514**	**−579.7358**	7.3E−08
	εDEg	−578.6581	−578.6533	−578.6528	−578.6448	3.62E−03
C07	IεJADE	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	0.00E+00
	AIS	**0.00E+00**	1.1279E−11	1.1735E−08	0.21778	2.7E+00
	εDEg	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	0.00E+00
C08	IεJADE	**0.00E+00**	**0.00E+00**	**3.6969E+00**	1.0941E+01	4.83E+00
	AIS	5.55E−12	1.2416	4.0919	**9.4817**	1.46E+00
	εDEg	**0.00E+00**	10.94154	6.727528	15.37535	5.56E+00
C09	IεJADE	**0.00E+00**	**0.00E+00**	1.3575E+01	3.3747E+02	6.75E+01
	AIS	5.201E−08	2.506E−03	2.698E+01	2.9953E+02	7.50E+01
	εDEg	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	0.00E+00
C10	IεJADE	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	0.00E+00
	AIS	2.684E−17	3.379E−13	1.620E+03	2.5737E+03	5.0E+02
	εDEg	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	0.00E+00
C11	IεJADE	−1.52271E−03	−8.7342E−02*	−8.0476E−02*	−8.7342E−02*	2.37E−02
	AIS	**−1.52272E−03**	**−1.52272E−03**	−9.19953E−04	1.39717E−03	8.23E−04
	εDEg	−1.52271E−03	−1.52271E−03	**−1.52271E−03**	**−1.52271E−03**	6.34E−11
C12	IεJADE	−1.9925E−01	−8.9092E+02*	−7.3483E+02*	−8.9092E+02*	3.34E+02
	AIS	**−570.1992**	**−452.6516**	**−435.7736**	**−255.3529**	6.02E+01
	εDEg	−570.0899	−423.1332	−336.734	−198.9129	1.78E+02

(Continued)

Table 2 (continued)

CEC 2010 10D		Best	Median	Mean	Worst	Std
C13	IεJADE	−68.42937	**−68.42937**	−67.2830	−42.4097	5.20E+00
	AIS	**−68.4298**	−68.2759	−67.8735	−67.0705	3.11E−01
	εDEg	−68.42937	−68.42936	**−68.42936**	**−68.42936**	1.02E−06
C14	IεJADE	**0.00E+00**	**0.00E+00**	1.5946E−01	3.9866E+00	7.97E−01
	AIS	1.5314E−14	1.0720E−12	1.2213E−04	2.2116E−04	2.9E−08
	εDEg	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	0.00E+00
C15	IεJADE	8.8596E+07	5.9227E+09	3.5738E+11	6.6612E+12	1.34E+12
	AIS	2.5265E−20	2.2083E−09	**5.1855E−09**	**2.8151E−08**	1.1E−08
	εDEg	**0.00E+00**	**0.00E+00**	0.1799	4.4974	8.81E−01
C16	IεJADE	9.0457E−02	5.1530E−01	4.9625E−01	9.4367E−01	2.39E−01
	AIS	7.2556E−20	**9.6902E−18**	**9.9593E−18**	**1.9631E−18**	6.27E−15
	εDEg	**0.00E+00**	0.2820	0.2702	1.0183	3.71E−01
C17	IεJADE	7.8210E−02	3.6965E−01	5.5203E−01	1.9225E+00	4.78E−01
	AIS	1.4250E−09	4.6817	2.9340	4.7120	2.29E+00
	εDEg	**1.4632E−17**	**5.6533E−03**	**0.1250**	**0.7302**	1.93E−01
C18	IεJADE	**0.00EE+00**	**2.4289E−28**	**1.0678E−26**	**2.4632E−24**	4.92E−26
	AIS	1.2521E−18	2.6675E−13	1.6590	6.6334	1.27E+00
	εDEg	3.7314E−20	4.0979E−19	9.6788E−19	9.2270E−18	1.81E−18

Note: * means infeasible solutions.

In contrast with the two other algorithms in 10D problems, the proposed IεJADE algorithm obtains the best results of index "Best" in 9 out of 18 functions, which are C01-02, C04, C07-10, C14 and C18. As for the "Median", "Mean" and "Worst" indexes, the proposed IεJADE algorithm obtains best outcome in 10, 6, and 6 out of 18 functions, respectively. AIS attains the best result in 10, 5, 6, and 6 out of 18 functions, as for the indexes of "Best", "Median", "Mean" and "Worst". The εDEg algorithm achieves 11, 9, 9, 9 out of 18 functions referring to "Best", "Median", "Mean" and "Worst" indexes. In summary, the εDEg approach outperforms the other algorithms in 10D problems.

Table 3: Statistical results for solving 18 benchmark functions from CEC 2010 at 30D

CEC 2010 30D		Best	Median	Mean	Worst	Std
C01	IεJADE	−0.8218844	**−0.8218843**	**−0.8216994**	**−0.8179475**	2.43E−03
	AIS	**−0.8219648**	−0.8217607	−0.8201145	−0.7577310	3.25E−04
	εDEg	−0.8218255	−0.8206172	−0.8208687	−0.7405572	7.10E−04
C02	IεJADE	−2.1557118	−2.0920230	−2.0938590	−2.0494674	3.01E−02
	AIS	**−2.2244440**	**−2.2222491**	**−2.2125155**	**−2.2027971**	2.84E−03
	εDEg	−2.169248	−2.152145	−2.174499	−2.117096	1.20E−02

(Continued)

Table 3 (continued)

CEC 2010 30D		Best	Median	Mean	Worst	Std
C03	IεJADE	28.67347	28.67347	88.0271	2033.3647	4.05E+03
	AIS	**8.537446E−05**	**2.920680E−04**	66.75758	89.20102	4.26E+02
	εDEg	28.67347	28.67347	**28.83785**	**32.78014**	8.0E−01
C04	IεJADE	3.4933E−05	9.9704E−01	3.8790E+00	1.9365E+01	6.90E+00
	AIS	−**2.07820E−02**	**2.3197E−03**	**1.9761E−03**	**1.29167E−02**	1.61E−03
	εDEg	4.6981E−03	6.9476E−03	8.1630E−03	1.7779E−02	3.06E−03
C05	IεJADE	−**483.6106**	−**483.6106**	−**483.6106**	−**483.6106**	1.95E−10
	AIS	−483.3385	−458.0834	−436.1129	−425.0065	2.51E+01
	εDEg	−453.1307	−450.0404	−449.5460	−442.1590	2.89E+00
C06	IεJADE	−**530.6248**	−**530.6164**	−**530.6127**	−**530.5953**	8.56E−03
	AIS	−530.3266	−528.1700	−454.2575	−424.5415	4.79E+01
	εDEg	−528.5750	−528.0407	−527.9068	−526.4539	4.74E−01
C07	IεJADE	**0.0000E+00**	**0.0000E+00**	4.7839E−01	3.9866E+00	1.32E+00
	AIS	1.7096E−15	1.3074E−12	1.0730051	2.3370275	1.61E+00
	εDEg	1.1471E−15	2.1144E−15	**2.6036E−15**	**5.4819E−15**	1.23E−15
C08	IεJADE	**0.0000E+00**	**0.0000E+00**	6.0348E+00	1.0614E+02	2.27E+01
	AIS	9.7601E−17	2.2556E−15	1.6530908	3.5832882	6.41E−01
	εDEg	2.51869E−14	6.51151E−14	**7.83146E−14**	**2.57811E−13**	4.85E−14
C09	IεJADE	**0.0000E+00**	**0.0000E+00**	**3.5203E−01**	4.4004E+00	1.22E+00
	AIS	8.91564E−20	1.5713E−12	1.565417	**2.418812**	1.96E+00
	εDEg	2.7707E−16	1.1246E−08	1.0721E+01	1.0528E+02	2.82E+01
C10	IεJADE	**0.0000E+00**	**3.4940E−26**	**3.2925E+00**	**2.6644E+01**	6.29E+00
	AIS	2.18338E−05	2.21765456	1.7847E+01	5.5992E+01	1.88E+01
	εDEg	32.5200	33.2890	33.2617	34.6324	4.54E−01
C11	IεJADE	−**3.9234E−04**	−**3.9234E−04**	3.4203E−03	1.8671E−02∗	7.78E−03
	AIS	−1.6409E−04	−3.214E−04	−1.579E−04	−1.793E−04	4.67E−05
	εDEg	−3.2684E–04	−2.8432E−04	−**2.8638E−04**	−**2.2363E−04**	2.71E−05
C12	IεJADE	−**1.9926E−01**	−**1.9926E−01**	−**1.9926E−01**	−**1.9926E−01**	5.89E−09
	AIS	−1.96415E−01	−6.07923E−02	4.28807E−06	6.19328E−06	4.52E−04
	εDEg	−1.9914E−01	5.3371E+02∗	3.5623E+02	5.4617E+02∗	2.89E+02
C13	IεJADE	−65.071846	−63.287401	−63.331724	−61.198101	1.04E+00
	AIS	−**68.428779**	−**67.956008**	−**66.235783**	−**66.002354**	2.27E−01
	εDEg	−66.42473	−65.31507	−65.35310	−64.29690	5.73E+01
C14	IεJADE	**0.0000E+00**	**0.0000E+00**	6.3785E−01	3.9866E+00	1.49E+00
	AIS	6.7351E−15	5.9906E−12	8.6828E−07	1.5555E−06	3.14E−07
	εDEg	5.01586E−14	1.35930E−13	**3.08940E−13**	**2.92351E−12**	5.61E−13

(Continued)

Table 3 (continued)

CEC 2010 30D		Best	Median	Mean	Worst	Std
C15	IεJADE	**0.0000E+00**	**0.0000E+00**	**3.3720E−01**	**4.2150E+00**	1.17E+00
	AIS	1.9950E−09	1.8800E−10	3.4128E+01	7.2308E+01	3.82E+01
	εDEg	2.1603E+01	2.1603E+01	2.1603E+01	2.1604E+01	1.10E−04
C16	IεJADE	2.2177E−01	5.8719E−01	5.7947E−01	8.6939E−01	1.61E−01
	AIS	3.9634E−11	6.20829E−10	8.20616E−02	5.474569E−01	1.12E−01
	εDEg	**0.0E+00**	**0.0E+00**	**2.1684E−21**	**5.4210E−20**	1.06E−20
C17	IεJADE	3.7691E+01	1.0063E+02	1.1696E+02	3.3964E+02	7.40E+01
	AIS	**4.1942E−07**	**3.06839E−06**	**3.60514123**	**8.94521185**	2.54E+00
	εDEg	2.16571E−01	5.315949	6.326487	18.89064	4.99E+00
C18	IεJADE	**7.1943E−21**	**1.0099E−17**	**1.3517E−07**	**3.3792E−06**	6.76E−07
	AIS	1.98864E−08	1.54857008	4.015246E+01	6.264366E+01	1.80E+01
	εDEg	1.226054	26.79497	87.54569	737.5363	1.66E+02

Note: * means infeasible solutions.

Compared with two competitive algorithms in 30D problems, the proposed IεJADE algorithm attains the best results in terms of "Best", "Median", "Mean", and "Worst" in 11, 13, 8, and 7 out of 18 functions. The AIS algorithm reaches 6, 5, 4, and 5 best results out of 18 functions in the above indexes. εDEg algorithm obtains 1, 1, 5, and 5 best results in the "Best", "Median", "Mean", and "Worst" indexes, respectively. It is worth mentioning that the proposed IεJADE algorithm demonstrates superior capabilities in handling the 30D problems, in comparison with the other two algorithms.

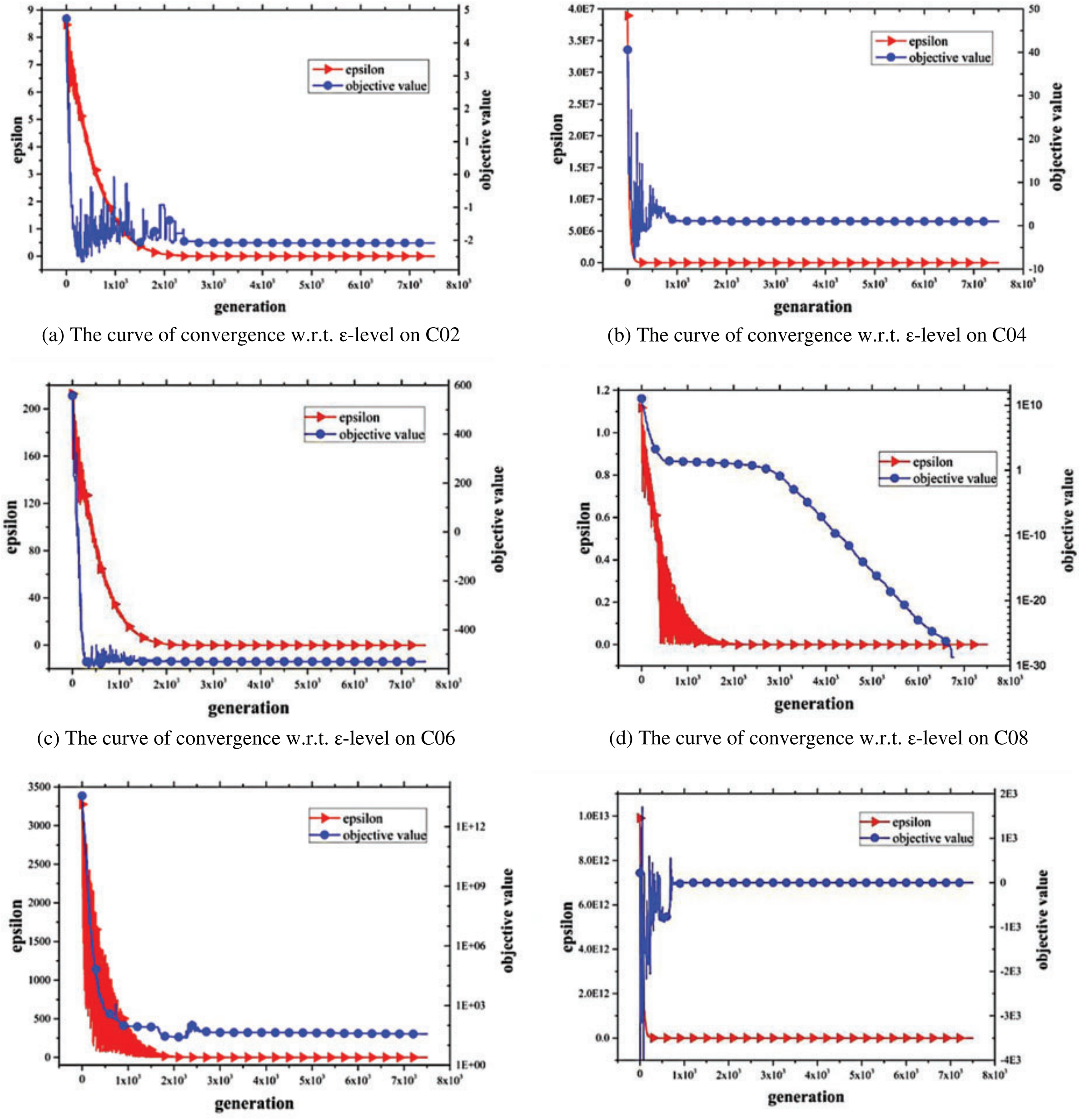
To conclude, the proposed algorithm has potential in solving complex and high dimensional problems while the performance in low dimensional problems is acceptable when compared with the two competitive algorithms.

4.4 Convergence Curve of ε-Level

The convergence curves of the 10 functions with 30 dimensions are given as follows. From Fig. 5, it is observed that the IεJADE algorithm can achieve convergence with a rapid speed. The convergence curves of ε-level also indicate that the proposed setting can be convergent to the original feasible region rapidly with a small number of fitness evaluations.

5 Case Study

A standard IEEE-30 bus test system is utilized to evaluate the effectiveness of the IεJADE algorithm. As one of the leading cases in power systems optimization, optimal power flow (OPF) aims to obtain the optimal solutions and minimize the objective function. Equality and inequality constraints are two different optimal objectives in this system, which are described in following subsection.



(a) The curve of convergence w.r.t. ε-level on C02

(b) The curve of convergence w.r.t. ε-level on C04

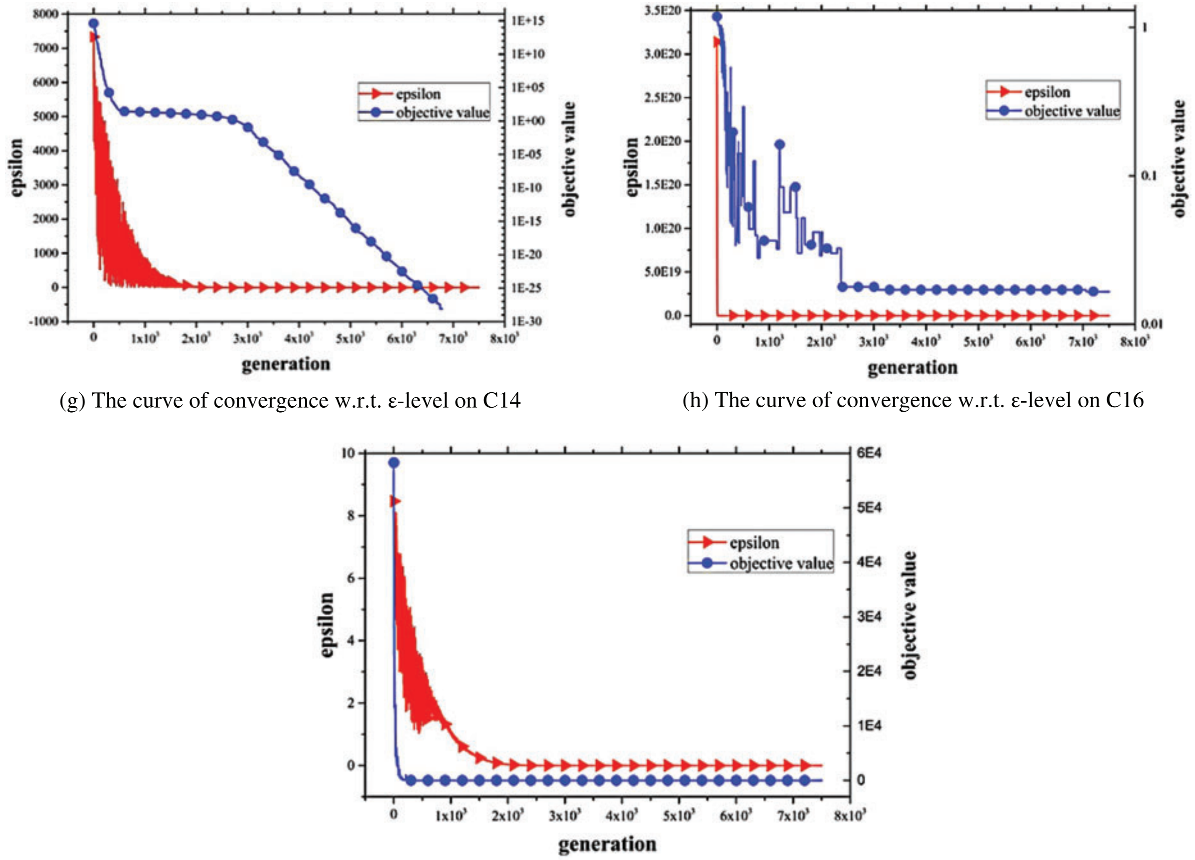
(c) The curve of convergence w.r.t. ε-level on C06

(d) The curve of convergence w.r.t. ε-level on C08

(e) The curve of convergence w.r.t. ε-level on C10

(f) The curve of convergence w.r.t. ε-level on C12

Figure 5: (Continued)



(g) The curve of convergence w.r.t. ε-level on C14

(h) The curve of convergence w.r.t. ε-level on C16

(i) The curve of convergence w.r.t. ε-level on C18

Figure 5: The curve of convergence w.r.t. ε-level

5.1 Equality Constraints

In OPF, typical load flow equations can be illustrated as follows:

$$P_{G_i} = P_{D_i} + V_i \sum_{j=1}^{NB} V_j \left[G_{ij} \cos\left(\delta_i - \delta_j\right) + B_{ij} \sin\left(\delta_i - \delta_j\right) \right], i = 1, \ldots, NB \tag{15}$$

$$Q_{G_i} = Q_{D_i} + V_i \sum_{j=1}^{NB} V_j \left[G_{ij} \cos\left(\delta_i - \delta_j\right) + B_{ij} \sin\left(\delta_i - \delta_j\right) \right], i = 1, \ldots, NB \tag{16}$$

where P_{G_i} and Q_{G_i} are the active power generation and reactive power generation, respectively. P_{D_i} and Q_{D_i} are the active power demand and reactive power demand, respectively. V_i and V_j are the voltages of ith and jth bus, respectively. G_{ij}, B_{ij} and δ_{ij} are the conductance, susceptance and phase difference of voltages between bus i and bus j, respectively.

5.2 Inequality Constraints

The inequality constraints are as follows:

$$P_{G_i}^{\min} \leq P_{G_i} \leq P_{G_i}^{\max}, i = 1, \ldots, NG \tag{17}$$

$$Q_{G_i}^{\min} \le Q_{G_i} \le Q_{G_i}^{\max}, i = 1, \ldots, NG \tag{18}$$

$$V_{G_i}^{\min} \le V_{G_i} \le V_{G_i}^{\max}, i = 1, \ldots, NG \tag{19}$$

$$Q_{C_j}^{\min} \le Q_{C_j} \le Q_{C_j}^{\max}, j = 1, \ldots, NC \tag{20}$$

$$T_k^{\min} \le T_k \le T_k^{\max}, i = 1, \ldots, NT \tag{21}$$

$$V_{Lm}^{\min} \le V_{Lm} \le V_{Lm}^{\max}, m = 1, \ldots, NL \tag{22}$$

$$S_{l_n} \le S_{l_n}^{\max}, n = 1, \ldots, nl \tag{23}$$

where Q_{C_j}, T_k, V_{Lm}, S_{l_n} are VAR injection of jth shunt compensator, transformer tap settings, load voltage of mth unit and apparent power flow of nth branch, respectively.

5.3 *Optimal Objectives*

Two single optimal objectives, minimization of emission and minimization of real power loss are studied. Emission in traditional electrical power system have been concerned as it emits harmful gases including SO_x, NO_x. The equation of minimization of emission is given by [29]:

$$f_{Emission} = \sum_{i=1}^{NG} \alpha_i + \beta_i P_{G_i} + \gamma_i P_{G_i}^2 + \varrho_i e^{(\phi_i P_{G_i})} \tag{24}$$

where $f_{Emission}$ is emission of test system. α_i, β_i, γ_i, ρ and ϕ_i are ith generator emission coefficients. The settings are listed in Table 4. The details can be referred in [29].

Table 4: Coefficient values of generators [30]

Generator	Bus	α	β	Γ	ϱ	ϕ
		(t/h)	(t/p.u.MWh)	(t/p.u.MW2h)	($/h)	(p.u.MW-1)
G1	1	4.091	−5.554	6.49	0.0002	2.857
G2	2	2.543	−6.047	5.638	0.0005	3.333
G3	5	4.258	−5.094	4.586	0.000001	8
G4	8	5.326	−3.55	3.38	0.002	2
G5	11	4.258	−5.094	4.586	0.000001	8
G6	13	6.131	−5.555	5.151	0.00001	6.667

The power loss is unavoidable but it can be reduced by adjusting voltage and voltage angle. The equation of real power loss is expressed as:

$$f_{Ploss} = \sum_{i=1}^{nl} \sum_{i \ne j}^{nl} G_{ij} \left[V_i^2 + V_j^2 - 2V_i V_j \cos\left(\delta_i - \delta_j\right) \right] \tag{25}$$

5.4 *Experimental Results*

The standard IEEE-30 bus system is used and four optimization objectives, including fuel cost, emission, fuel cost with valve-point effect and voltage deviation are considered. The ACDE [29], ECHT-DE [30], SF-DE [30], and SP-DE [30] are compared with the proposed approach. The results are given in Table 5.

Table 5: Statistical analysis outcome for the IEEE-30 bus system test

Case#	Algorithm	Min	Max	Mean	Std
Case 1	ECHT-DE	**0.2048**	**0.2048**	**0.2048**	**0.0000**
	SF-DE	**0.2048**	**0.2048**	**0.2048**	**0.0000**
	SP-DE	**0.2048**	**0.2048**	**0.2048**	**0.0000**
	ACDE	**0.2048**	**0.2048**	**0.2048**	**0.0000**
	IεJADE	**0.2048**	**0.2048**	**0.2048**	**0.0000**
Case 2	ECHT-DE	3.0850	3.0871	3.0858	0.0005
	SF-DE	3.0845	3.0857	3.0849	**0.0003**
	SP-DE	3.0844	3.0854	3.0848	0.0003
	ACDE	**3.0840**	3.0862	**3.0845**	0.0005
	IεJADE	3.0842	**3.0851**	**3.0845**	**0.0003**

The best results are in bold. It can be concluded that the proposed IεJADE algorithm has competitiveness in this application. In Case 1, the proposed algorithm can obtain the best results as other state-of-the-art algorithms. In Case 2, the proposed algorithm can achieve the best Max, Mean and Std values, which means IεJADE has the most stable performance than other algorithms.

In the IEEE 30-bus system, the load bus voltage has its limits [0.95−1.05 p.u.] [29]. Fig. 6 shows the load bus voltages are within the permissible range.

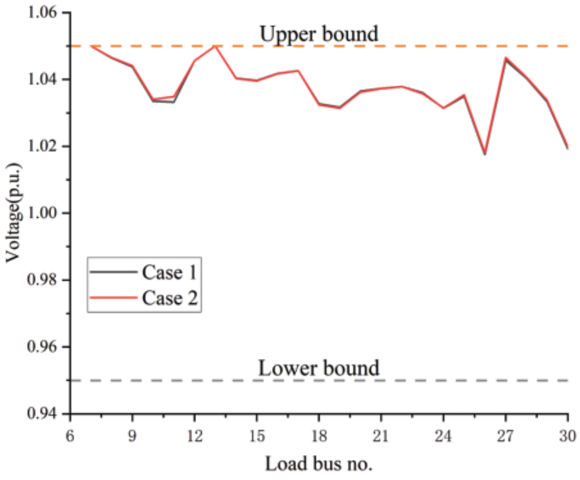


Figure 6: The voltage of load bus by IεJADE

6 Concluding Remarks

To effectively tackle the constraints in COPs, an improved *p*best selection mechanism and an adaptive ε setting are presented. The effectiveness of the proposed algorithm is tested on benchmark test functions in CEC 2006, CEC 2010, and the standard IEEE 30-bus test system. The numerical results illustrate that the proposed IεJADE algorithm is effective in tackling the constrained benchmark test functions and real-world applications. The proposed *p*best selection mechanism presents a way to select and compare the infeasible and feasible individuals, which is effective to tackle them as integrity rather than two separate parts.

The improved *p*best selection mechanism focus on finding feasible solution first. Therefore, it is more suitable for solving problems with smaller feasible domains. If a feasible solution can be easily found, the *p*best strategy which directly optimizes the objective function is more suitable.

It is challenging to obtain suitable ε values. The proposed ε setting can be improved by machine learning and other self-learning ways. Historical information can be used to obtain the values. IεJADE algorithm can be used to tackle more complex engineering-based constrained optimization problems [31–33] in the future. Besides, more optimal objectives in IEEE 30-bus test system can be considered in further research.

Funding Statement: This research was supported by National Natural Science Foundation of China under Grant Nos. 52005447, 72271222, 71371170, 71871203, L1924063, Zhejiang Provincial Natural Science Foundation of China under Grant No. LQ21E050014, and Foundation of Zhejiang Education Committee under Grant No. Y201840056.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Sönmez, A. İ., Baykasoğlu, A., Dereli, T., Fılız, İ. H. (1999). Dynamic optimization of multipass milling operations via geometric programming. *International Journal of Machine Tools and Manufacture, 39(2),* 297–320. https://doi.org/10.1016/S0890-6955(98)00027-3
2. Gu, L., Yang, R. J., Tho, C., Makowskit, M., Faruquet, O. et al. (2001). Optimisation and robustness for crashworthiness of side impact. *International Journal of Vehicle Design, 26(4),* 348. https://doi.org/10.1504/IJVD.2001.005210
3. Storn, R., Price, K. (1997). Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization, 11(4),* 341–359. https://doi.org/10.1023/A:1008202821328
4. Zhang, J., Sanderson, A. C. (2009). JADE: Adaptive differential evolution with optional external archive. *IEEE Transactions on Evolutionary Computation, 13(5),* 945–958. https://doi.org/10.1109/TEVC.2009.2014613
5. Gong, W., Cai, Z., Wang, Y. (2014). Repairing the crossover rate in adaptive differential evolution. *Applied Soft Computing, 15(4),* 149–168. https://doi.org/10.1016/j.asoc.2013.11.005
6. Segundo, G. A. S., Krohling, R. A., Cosme, R. C. (2012). A differential evolution approach for solving constrained min-max optimization problems. *Expert Systems with Applications, 39(18),* 13440–13450. https://doi.org/10.1016/j.eswa.2012.05.059
7. Xu, B., Tao, L., Chen, X., Cheng, W. (2019). Adaptive differential evolution with multi-population-based mutation operators for constrained optimization. *Soft Computing, 23(10),* 3423–3447. https://doi.org/10.1007/s00500-017-3001-0
8. Nguyen, T. H., Vu, A. T. (2023). An efficient differential evolution for truss sizing optimization using adaboost classifier. *Computer Modeling in Engineering & Sciences, 134(1),* 429–458. https://doi.org/10.32604/cmes.2022.020819

9. Fu, L., Ouyang, H., Zhang, C., Li, S., Mohamed, A. W. (2022). A constrained cooperative adaptive multi-population differential evolutionary algorithm for economic load dispatch problems. *Applied Soft Computing, 121(2),* 108719. https://doi.org/10.1016/j.asoc.2022.108719
10. Gong, W., Cai, Z., Liang, D. (2015). Adaptive ranking mutation operator based differential evolution for constrained optimization. *IEEE Transactions on Cybernetics, 45(4),* 716–727. https://doi.org/10.1109/TCYB.2014.2334692
11. Wang, Y., Yin, D. Q., Yang, S., Sun, G. (2019). Global and local surrogate-assisted differential evolution for expensive constrained optimization problems with inequality constraints. *IEEE Transactions on Cybernetics, 49(5),* 1642–1656. https://doi.org/10.1109/TCYB.2018.2809430
12. Takahama, T., Sakai, S. (2006). Constrained optimization by the ε constrained differential evolution with gradient-based mutation and feasible elites. *2006 IEEE International Conference on Evolutionary Computation*, pp. 1–8. Vancouver, BC, Canada.
13. Runarsson, T. P., Yao, X. (2000). Stochastic ranking for constrained evolutionary optimization. *IEEE Transactions on Evolutionary Computation, 4(3),* 284–294. https://doi.org/10.1109/4235.873238
14. Deb, K. (2000). An efficient constraint handling method for genetic algorithms. *Computer Methods in Applied Mechanics and Engineering, 186(2),* 311–338. https://doi.org/10.1016/S0045-7825(99)00389-8
15. Takahama, T., Sakai, S. (2013). Efficient constrained optimization by the ε constrained differential evolution with rough approximation using kernel regression. *2013 IEEE Congress on Evolutionary Computation*, pp. 1334–1341.
16. Wang, B. C., Li, H. X., Li, J. P., Wang, Y. (2019). Composite differential evolution for constrained evolutionary optimization. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 49(7),* 1482–1495.
17. Zhang, C., Qin, A. K., Shen, W., Gao, L., Tan, K. C. et al. (2022). ε-Constrained differential evolution using an adaptive ε-level control method. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 52(2),* 769–785. https://doi.org/10.1109/TSMC.2020.3010120
18. Stanovov, V., Akhmedova, S., Semenkin, E. (2019). Selective pressure in constrained differential evolution. *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pp. 83–84. Prague Czech Republic, ACM.
19. Ni, Z. H., Cheng, L., Zhang, C. M. (2019). ε Constrained multi-mutant rank-based differential evolution algorithm and its application in multipath repression. *2019 IEEE 15th International Conference on Control and Automation (ICCA)*, pp. 893–898. Edinburgh, UK.
20. Duc Nguyen, M., Nguyen Hai, H., Al-Ansari, N., Amiri, M., Ly, H. B. et al. (2022). Hybridization of differential evolution and adaptive-networkbased fuzzy inference system in estimation of compression coefficient of plastic clay soil. *Computer Modeling in Engineering & Sciences, 130(1),* 149–166. https://doi.org/10.32604/cmes.2022.017355
21. Yi, W., Gao, L., Pei, Z., Lu, J., Chen, Y. (2021). ε Constrained differential evolution using halfspace partition for optimization problems. *Journal of Intelligent Manufacturing, 32(1),* 157–178. https://doi.org/10.1007/s10845-020-01565-2
22. Liang, J. J., Runarsson, T. P., Mezura-Montes, E., Clerc, M., Suganthan, P. N. et al. (2006). Problem definitions and evaluation criteria for the CEC 2006 special session on constrained real-parameter optimization. https://cw.fel.cvut.cz/b181/_media/courses/a0m33eoa/cviceni/2006_problem_definitions_and_evaluation_criteria_for_the_cec_2006_special_session_on_constraint_real-parameter_optimization.pdf
23. Zhang, C., Lin, Q., Gao, L., Li, X. (2015). Backtracking search algorithm with three constraint handling methods for constrained optimization problems. *Expert Systems with Applications, 42(21),* 7831–7845. https://doi.org/10.1016/j.eswa.2015.05.050

24. Mallipeddi, R., Suganthan, P. N. (2010). Problem definitions and evaluation criteria for the CEC 2010 competition on constrained real-parameter optimization. http://www.al-roomi.org/multimedia/CEC_Database/CEC2010/RealParameterOptimization/CEC2010_RealParameterOptimization_TechnicalReport.pdf
25. Tessema, B., Yen, G. G. (2009). An adaptive penalty formulation for constrained evolutionary optimization. *IEEE Transactions on Systems Man and Cybernetics–Part A: Systems and Humans, 39(3),* 565–578. https://doi.org/10.1109/TSMCA.2009.2013333
26. Zavala, A. E. M., Aguirre, A. H., Diharce, E. R. V. (2009). Continuous constrained optimization with dynamic tolerance using the COPSO algorithm. In: *Constraint-handling in evolutionary optimization*, pp. 1–23. Berlin, Heidelberg: Springer.
27. Zhang, W., Yen, G. G., He, Z. (2014). Constrained optimization via artificial immune system. *IEEE Transactions on Cybernetics, 44(2),* 185–198. https://doi.org/10.1109/TCYB.2013.2250956
28. Takahama, T., Sakai, S. (2010). Constrained optimization by the ε constrained differential evolution with an archive and gradient-based mutation. *IEEE Congress on Evolutionary Computation*, pp. 1–9.
29. Li, S., Gong, W., Hu, C., Yan, X., Wang, L. et al. (2021). Adaptive constraint differential evolution for optimal power flow. *Energy, 235(8),* 121362. https://doi.org/10.1016/j.energy.2021.121362
30. Biswas, P. P., Suganthan, P. N., Mallipeddi, R., Amaratunga, G. A. J. (2018). Optimal power flow solutions using differential evolution algorithm integrated with effective constraint handling techniques. *Engineering Applications of Artificial Intelligence, 68(1),* 81–100. https://doi.org/10.1016/j.engappai.2017.10.019
31. Liu, Z. Z., Wang, Y., Wang, B. C. (2021). Indicator-based constrained multiobjective evolutionary algorithms. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 51(9),* 5414–5426. https://doi.org/10.1109/TSMC.2019.2954491
32. Wang, B. C., Li, H. X., Zhang, Q., Wang, Y. (2021). Decomposition-based multiobjective optimization for constrained evolutionary optimization. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 51(1),* 574–587. https://doi.org/10.1109/TSMC.2018.2876335
33. Wen, L., Gao, L., Li, X., Li, H. (2022). A new genetic algorithm based evolutionary neural architecture search for image classification. *Swarm and Evolutionary Computation, 75(5),* 101191. https://doi.org/10.1016/j.swevo.2022.101191