2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

An Enhanced Non-dominated Sorting Based Fruit Fly Optimization Algorithm for Solving Environmental Economic Dispatch Problem

Xiaolong Zheng, Ling Wang and Shengyao Wang

***Abstract*—A fruit fly optimization algorithm based on the enhanced non-dominated sorting (ESFOA) is proposed to solve the environmental economic dispatch (EED) problem. To measure the difference between two non-dominated solutions, the concept of the enhanced non-dominance is defined, and the degrees of dominance and non-dominance are presented. To enhance the parallel search ability, multiple fruit flies groups are used to perform evolutionary search in the ESFOA. In the vision-based search process, the best fruit fly is determined according to the enhanced non-dominance value. To guarantee the feasibility of the new solutions, an effective heuristic mechanism to handle constraints is adopted to repair the infeasible solutions. Meanwhile, an external archive is used to store the non-dominated solutions. The influence of parameter setting is investigated based on the Taguchi method of design of experiment, and a suitable parameter setting is suggested. Finally, numerical tests are carried out by using the IEEE 30-bus benchmark. The comparisons to some existing methods by using the technique for order preference by similarity to ideal solution (TOPSIS) demonstrate the effectiveness of the proposed algorithm.**

I. Introduction

THE economic dispatch (ED) problem is to schedule the generating unit outputs to meet the load demand at a minimum total fuel cost while satisfying the system constraints. With the increasing concern of environmental pollution, the environmental economic dispatch (EED) problem considering both the fuel cost and the emission of pollutants simultaneously is becoming more and more desirable in the operations of power system [1-9]. As a constrained multi-objective optimization problem, at the same time, the EED problem has been proved to be NP-hard. Hence, the EED problem is an important real-world engineering optimization problem with both practical and academic significance.

During the past two decades, many approaches to solve the EED have been proposed in the literature. The early researchers usually formulated the EED problem as a single objective optimization problem [1-5]. Using traditional single objective methods, only one satisfactory solution can be obtained. So, multiple runs with different weights are needed to produce more solutions to help making operating decisions. In [1], the emission was treated as a constraint with a permissible limit. In [2-3], the fuel cost and emission objectives were linearly combined as a weighted sum, and a newly proposed charged system search (CSS) algorithm was applied to solve the transformed EED problem [3]. In [4], conic scalarization method was adopted to turn the two objectives into a single one. In [5-6], a set of the non-dominated solutions could be obtained by varying the weights, and a biogeography-based optimization (BBO) was applied to solve the EED problem. Setting the weight as 0, 0.5 and 1, it obtained the corresponding best cost, compromise and emission solutions, respectively [6]. However, these methods could offer little information about the tradeoff relationship of the different objectives.

Recently, some Pareto-dominance based multi-objective genetic algorithms (GA) have been proposed, including non-dominated sorting genetic algorithm (NSGA) [7], strength Pareto evolutionary algorithm (SPEA) [8], niched Pareto genetic algorithm (NPGA) [9] and NSGA-II [10]. In addition, several swarm based algorithms have been proposed to solve the EED problem. The relevant results in this filed include the following. Wu [11] proposed a multi-objective differential evolution (MODE) algorithm and designed a crowding entropy strategy to preserve the diversity of the Pareto non-dominated solutions. Lu [12] adopted a self-adaptive second mutation operator and proposed an enhanced MODE. Abido [13] proposed a multi-objective particle swarm optimization (MOPSO) approach. Gong et al. [14] proposed a hybrid multi-objective algorithm based on PSO and DE (PSODE). Zhang et al. [15] proposed a bare-bones multi-objective particle swarm optimization (BB-MPSO), where a mutation operator was developed with the action range varying over time and the particle diversity based global leader particle updating approach. Pandit et al. [16] proposed an improved bacterial foraging algorithm (IBFA) with a parameter automation strategy, which adopted a crossover operation to improve the computational efficiency. In [17], an improved scatter search (ISS) with new combination scheme was presented. In [18], Jadhav and Roy proposed a gbest guided artificial bee colony algorithm (GABC). For the above mentioned approaches, crowding distance and entropy are used to handle multiple objectives.

As a recently developed swarm evolutionary optimization algorithm, fruit fly optimization algorithm (FOA) [19] is inspired by the foraging behaviors of real fruit flies. The FOA is easy to implement and it has few parameters that need adjusting. Due to the merits, the FOA has already been successfully applied to several continuous optimization problems, including financial distress [19], web auction logistics service [20], general regression neural network

X.L. Zheng is with Department of Automation, Tsinghua University, Beijing, China (e-mail: zhengxl11@mails.tsinghua.edu.cn).
L. Wang is with Department of Automation, Tsinghua University, Beijing, China (phone: 0086-010-62794267; e-mail: wangling@tsinghua.edu.cn).
S.Y. Wang is with Department of Automation, Tsinghua University, Beijing, China (e-mail: wangshengyao@tsinghua.org.cn).
This work was financially supported by the National Key Basic Research and Development Program of China (No. 2013CB329503), the National Science Foundation of China (No. 61174189), and the Doctoral Program Foundation of Institutions of Higher Education of China (No. 20130002110057).

optimization [21], and proportional integral derivative controller tuning [22]. Essentially, the EED is a complex continuous optimization problem with multiple objectives, which is difficult to be solved by traditional methods. The FOA is a swarm based intelligent algorithm with a parallel search framework, which could be a powerful solver due to its prominent results in solving other problems. In this paper, an enhanced non-dominated sorting based FOA (ESFOA) with multiple fruit fly groups is proposed to solve the EED. The concept of the enhanced non-dominance is defined to measure the difference of two non-dominated solutions quantitatively. It helps to sort solutions according to the non-dominance values. The ESFOA has few parameters, whose influence on the performance is investigated by using the design of experiment test. The performances of the ESFOA are tested with the well-known IEEE 30-bus benchmark, and the effectiveness of the ESFOA is demonstrated by the comparisons to some existing algorithms using the TOPSIS.

The remaining of the paper is organized as follows: The problem is formulated in Section II. Section III provides the concept of the enhanced non-dominance and the sorting algorithm. The proposed ESFOA is presented in Section IV in details. The testing results are provided in Section V. Finally, we end the paper with some conclusions and future work in Section VI.

II. Problem Formulation

The typical EED problem is to minimize two objectives simultaneously, i.e. fuel cost and emission, while satisfying some inequality and equality constraints. The mathematical model can be formulated as follows [11].

A. Objective Functions

Minimization of fuel cost:

$$F(P_G)=\sum_{i=1}^{N}(a_i+b_iP_{Gi}+c_iP_{Gi}^{\ 2}) \tag{1}$$

Minimization of emission:

$$E(P_G)=\sum_{i=1}^{N}\{10^{-2}(\alpha_i+\beta_iP_{Gi}+\gamma_iP_{Gi}^{\ 2})+\xi_i\exp(\lambda_iP_{Gi})\} \tag{2}$$

where N is the number of generators, a_i, b_i and c_i are the cost coefficients of the i-th generator, P_{Gi} is the real power output of the i-th generator, and α_i, β_i, γ_i, ξ_i and λ_i are coefficients of the i-th generator emission characteristics, and P_G is the real power output vector of the generators defined as $P_G=[P_{G1}, P_{G2},\ldots, P_{GN}]^T$.

B. Problem Constraints

Generation capacity constraints:

$$P_{Gi}^{\min}\le P_{Gi}\le P_{Gi}^{\max}, i=1,2,\ldots,N \tag{3}$$

where P_{Gi}^{min} and P_{Gi}^{max} are the minimum and maximum of the real power outputs of the i-th generator, respectively.

Power balance constraints:

$$\sum_{i=1}^{N}P_{Gi}-P_D-P_L=0 \tag{4}$$

where P_D is the total system demand and P_L is the real power loss in transmission calculated as follows:

$$P_L=\sum_{i=1}^{N}\sum_{j=1}^{N}P_{Gi}B_{ij}P_{Gj}+\sum_{i=1}^{N}P_{Gi}B_{0i}+B_{00} \tag{5}$$

III. Enhanced Non-dominated Sorting Algorithm

A. Enhanced Non-dominance

It might not be perfect to solve the multi-objective optimization problem using the basic definition of Pareto dominance [23]. The reason mainly lies in two folds: first, Pareto dominance does not make a difference between two solutions if neither is dominated; and second, Pareto dominance does not measure the degree quantitatively by which one solution dominates the other. Hence, the concept of the enhanced non-dominance is presented. With such a concept, it can compare different non-dominated solutions and prioritize between any two solutions. Different from the crowding distance, the enhanced non-dominance emphasizes the optimization of the objective value instead of the uniformity of the distribution of solutions.

Without loss of generality, suppose that a minimization continuous problem has m objective functions f_i (i=1,2,...,m), and the solution space is denoted as $P\subseteq R^N$, where N is the dimensionality.

Definition 1 Enhanced i-dominance by a solution.

Solutions u, $v\in P$, solution u enhanced i-dominates solution v, denoted as $u\,\Gamma_i\,v$, iff $f_i(u)\le f_i(v)$. Moreover, a monotonically non-decreasing mapping $\mu^{edom}\rightarrow[0,1]$ is defined to measure the degree by which u i-dominates v:

$$\mu^{edom}(u\Gamma_i v)\equiv\mu^{edom}(f_i(v)-f_i(u))=\frac{f_i(v)-f_i(u)}{f_{i\max}-f_{i\min}} \tag{6}$$

where f_{imin} and f_{imax} are the minimum and maximum of objective function i (i=1,2,…,m) in the current solution set, respectively. If solution u i-dominates solution v, then $\mu^{edom}(v\,\Gamma_i\,u)=0$.

Definition 2 Enhanced dominance by a solution.

Solutions u, $v\in P$, solution u enhanced dominates solution v, denoted as $u\Gamma v$, iff $\forall\ i\in\{1,2,\ldots,m\}$, $u\ \Gamma_i\ v$ and $\exists\, i\in\{1,2,\ldots,m\}, f_i(u)<f_i(v)$. Furthermore, the degree by which solution u enhanced dominates solution v is defined as follows:

$$\mu^{edom}(u\Gamma v)=\sum_{j=1}^{m}\frac{f_j(u)-f_j(v)}{f_{j\max}-f_{j\min}} \tag{7}$$

Definition 3 Enhanced non-dominance by a solution.

Solutions u, $v\in P$, solution u enhanced non-dominates solution v, denoted as $u\not\Gamma v$ or $v\not\Gamma u$, iff $\exists\, i, j\in\{1,2,\ldots,m\}$, $u\,\Gamma_i\,v$ and $v\,\Gamma_j\,u$. Furthermore, the degree by which solution u enhanced non-dominates solution v is defined as:

$$\mu^{edom}(u\not\Gamma v)=\sum_{\substack{i=1\& \\ u\Gamma_i v}}^{m}\frac{f_i(v)-f_i(u)}{f_{i\max}-f_{i\min}} \tag{8}$$

Definition 4 Enhanced dominance in a population.

Given a population S, solution $u \in P$ enhanced dominates in S, denoted as $u \, \Gamma \, S$, iff $\forall \, v \in S$, $v \neq u$, $u \Gamma v$. Furthermore, the degree by which solution u enhanced dominates in population S is defined as follows:

$$\mu^{edom}(u\Gamma S) = \bigcup_{\substack{v \in S \\ v \neq u}} \mu^{edom}(u\Gamma v) = \sum_{v \in S \& v \neq u} \sum_{i=1}^{m} \frac{f_i(v) - f_i(u)}{f_{i\max} - f_{i\min}} \quad (9)$$

Definition 5 Enhanced non-dominance in a population.

Given a population S, solution $u \in P$ enhanced non-dominates in S, denoted as $u \not\Gamma S$, iff $\exists \, v \in S$, $v \neq u$, $u \not\Gamma v$. Furthermore, the degree by which solution u enhanced non-dominates in S is defined as folllows:

$$\mu^{edom}(u\Gamma S) = \bigcup_{\substack{v \in S \\ v \neq u}} \mu^{edom}(u\Gamma v) = \sum_{v \in S \& v \neq u} \sum_{i=1}^{m} \frac{f_i(v) - f_i(u)}{f_{i\max} - f_{i\min}} \quad (10)$$

where $S_1=\{v \in S | \, u \Gamma v \}$ and $S_2=\{v \in S | \, u \not\Gamma v \}$.

B. Enhanced Non-dominated Sorting

Once the enhanced non-dominance values of all solutions in the population are calculated, the solutions can be sorted according to the values in an ascending order.

Property 1 If solutions u, v meet $u \, \Gamma \, v$, then $ES(u)>ES(v)$, where $ES(u)$ and $ES(v)$ are the enhanced non-dominance values of u, v in the population S, respectively.

Proof: As u, v meet $u \; \Gamma \; v$,, then $\forall \;\; i \in \{1,2,\ldots,m\}$, $f_i(u) \leq f_i(v)$ and $\exists j \in \{1,2,\ldots,m\}$, $f_j(u)<f_j(v)$. Thus, $\mu^{edom}(u \, \Gamma_i \, x) \geq \mu^{edom}(v \, \Gamma_i \, x)$. $\forall \, x \in P$, $x \neq u$, v. Since $\mu^{edom}(v \not\Gamma u)=0$, so $ES(u)=\mu^{edom}(u \;\Gamma_i\; v)+ \sum_{x \in P, x \neq u,v} \sum_{i=1}^{m} \mu^{edom}(u\Gamma_i x) > \mu^{edom}(v \;\Gamma_i\; u)+ \sum_{x \in P, x \neq u,v} \sum_{i=1}^{m} \mu^{edom}(v\Gamma_i x) = ES(v)$.

According to Property 1, the enhanced non-dominance value of a non-dominated solution must be larger than that of a dominated one. So, it guarantees that the non-dominated solutions are better than the dominated ones.

IV. The Proposed ESFOA

A. Fruit fly optimization Algorithm (FOA)

FOA is a newly developed swarm-based optimization algorithm, which simulates the foraging behaviors of fruit fly swarms. The evolution process of the FOA consists of two major procedures: smell-based search and vision-based search. The procedure of the basic FOA is as follows:

Step 1. Initialization: set the parameters and randomly generate the fruit fly group.

Step 2. Smell-based search process: randomly generate several fruit flies around the fruit fly group to construct a population.

Step 3. Evaluation: evaluate the population to obtain the smell concentration values (fitness value) of each fruit fly.

Step 4. Vision-based search process: find the best fruit fly with the maximum smell concentration value, and then let the fruit fly group fly towards the best one.

Step 5. Stopping criterion: end the algorithm if it reaches the maximum number of evaluations; otherwise, go back to Step 2.

B. Encoding scheme

In this paper, the real power output of each generator is regarded as the encoded element. So, a solution is represented by a vector composed of N elements. For example, to optimize the IEEE 30-bus 6-generator test system, each fruit fly is represented by a 6-bit real coded string, i.e.

$$\bar{P}_i = (P_{i,1} \; P_{i,2} \; P_{i,3} \; P_{i,4} \; P_{i,5} \; P_{i,6}), \; i=1,2,\ldots,NP$$

where NP is the size of the population.

C. Initialization

Random strategy is adopted in the initialization. In the ESFOA, NP fruit fly groups are used to emphasize the parallel search. All the groups are generated randomly. To be specific, each element of a fruit fly group is randomly generated within the feasible real power output range:

$$P_{i,j}=P_{Gminj}+rand\times(P_{Gmaxj}-P_{Gminj}) \quad (11)$$

where i is the i-th group, j is the j-th generator, $rand \in U(0,1)$, and U denotes uniform distribution.

Note that, actually the fruit fly group is a central location of the fruit fly swarm, which can be regarded as a special solution when implementing the FOA.

D. Smell-based search and vision-based search

In the smell-based search, S fruit flies are generated around each fruit fly group to construct a sub-population. To be specific, each element of the new fruit fly is generated as follows:

$$P_{i,j}=P_{i,j}+\theta \times rand \quad (12)$$

where θ is the scope of the smell-based search.

In the vision-based search, S generated fruit flies in each sub-population are sorted by using the enhanced non-dominated sorting algorithm. Then, each fruit fly group is replaced with the corresponding best fruit fly.

E. Constraints Handling

The ESFOA adopts a heuristic constraint handling mechanism [8] to repair the infeasible solutions during the evolution process. The element of a solution is modified as follows to satisfy the generation capacity constraints:

$$X_j = \begin{cases} P_{G\min j}, if \; X_j < P_{G\min j} \\ P_{G\max j}, if \; X_j > P_{G\max j} \end{cases} \quad (13)$$

Then, the following procedure is used to repair the infeasible solutions to satisfy the power balance constraints:

Step 1: Calculate the violation of the real power balance constraint P_{Diff}:

$$P_{Diff} = P_D + P_L - \sum_{j=1}^{N} P_{Gj} \quad (14)$$

Step 2: If $P_{Diff}=0$, then end the procedure. Otherwise, if $P_{Diff}>P_{Diff}VioLimt$, then go to Step 3; else, go to Step 5.

Step 3: Calculate the average violation of real power balance constraint $P_{AvgDiff}$:

$$P_{AvgDiff} = (P_D + P_L - \sum_{j=1}^{N} P_{Gj}) / N \quad (15)$$

Step 4: Adjust the real power output of all the generators as Eq. (16). If the generation capacity constraint is violated, the solution will be modified according to Eq. (13). Then, go back to Step 1.

$$P_{Gj}=P_{Gj}+P_{AvgDiff}\ ,j=1,2,\ldots,N \quad (16)$$

Step 5: Set the iteration number of fine adjusting as $l_{fin}=0$.

Step 6: Calculate P_{Diff}. Randomly choose a generator r, and adjust the power output $P_{Gr}=P_{Gr}+P_{Diff}$. If the generation capacity constraint is violated, the solution will also be modified according to Eq. (13)

Step 7: Let $l_{fin}=l_{fin}+1$. If $l_{fin}< l_{finmax}$, then go to Step 6; otherwise, end the procedure.

F. External archive

To obtain a good Pareto-optimal solution set, it is important to retain the non-dominated solutions during the search process. Generally, a non-dominated solution in the current population may not be necessarily non-dominated by all the historical solutions. To guarantee that the final solutions are non-dominated with respect to all the generated solutions, an external archive is used to store the non-dominated solutions explored during the search.

Here, an external archive without capacity limitation is adopted. At each generation, if the non-dominated solution in the current population is dominated by a member of the archive, then it will be rejected. Besides, if the trial solution dominates some members of the archive, the dominated members of the archive will be removed and the trial solution will enter the archive.

G. The proposed ESFOA

The procedure of the ESFOA is illustrated as follows:

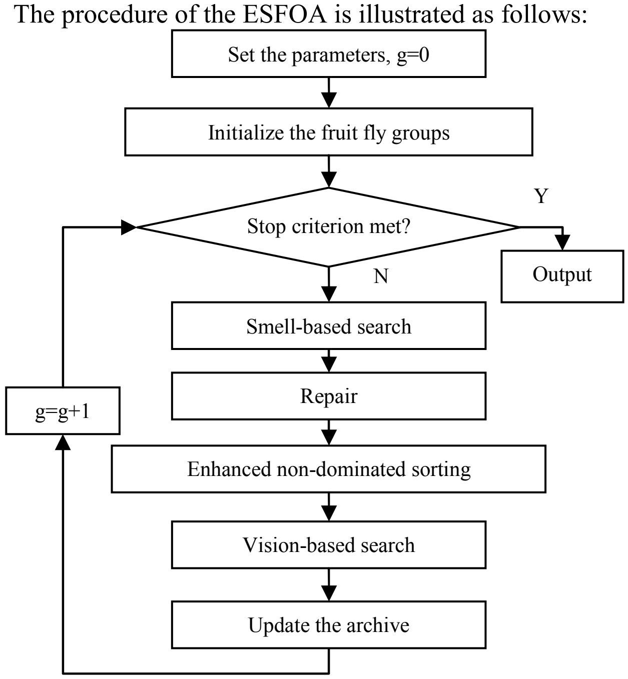


Fig. 1. The flowchart of ESFOA.

V. Experiments and Results

The performance of the ESFOA is tested with the standard IEEE 30-bus 6-generator power system, which has been widely used in literature. The total system demand amounts 2.834 p.u, and the transmission loss is considered. The system parameters including fuel cost and emission coefficients are listed in Table I, and the coefficients used in Eq. (5) are shown as Eqs. (17)-(19).

TABLE I

GENERATOR COST AND EMISSION COEFFICIENTS

	P_{G1}	P_{G2}	P_{G3}	P_{G4}	P_{G5}	P_{G6}
a	10	10	20	10	20	10
b	200	150	180	100	180	150
c	100	120	40	60	40	100
α	4.091	2.543	4.258	5.326	4.258	6.131
β	-5.543	-6.047	-5.094	-3.550	-5.094	-5.555
γ	6.490	5.638	4.586	3.380	4.586	5.151
ξ	2.0e-4	5.0e-4	1.0e-6	2.0e-3	1.0e-6	1.0e-5
λ	2.857	3.333	8.000	2.000	8.000	6.667
P_{Gmax}	1.5	1.5	1.5	1.5	1.5	1.5
P_{Gmin}	0.05	0.05	0.05	0.05	0.05	0.05

$$B=\begin{bmatrix} 0.1382 & -0.0299 & 0.0044 & -0.0022 & -0.0010 & -0.0008 \\ -0.0299 & 0.0487 & -0.0025 & 0.004 & 0.0016 & 0.0041 \\ 0.0044 & -0.0025 & 0.0182 & -0.0070 & -0.0066 & -0.0066 \\ -0.0022 & 0.0004 & -0.0070 & 0.0137 & 0.0050 & 0.0033 \\ -0.0010 & 0.0016 & -0.0066 & 0.0050 & 0.0109 & 0.0005 \\ -0.0008 & 0.0041 & -0.0066 & 0.0033 & 0.0005 & 0.0244 \end{bmatrix} \quad (17)$$

$$B_0=[-0.0107\ 0.006\ -0.0017\ 0.0009\ 0.0002\ 0.003] \quad (18)$$

$$B_{00}=9.8573\times10^{-4} \quad (19)$$

A. Parameter Setting of ESFOA

The ESFOA contains three key parameters: population size (*NP*), the number of generated neighbors (*SN*) and the scope of the smell-based search (θ). To investigate the influence of these parameters, the Taguchi method of design of experiment (DOE) [24] is applied. Different combinations of the values are listed in Table II. In addition, the maximum number of iteration is set as $l_{finmax}=30$ for constraints handling procedure, which is enough to guarantee the feasibility of the solution. Experiment is done with Windows XP, Inter Core2 Duo T5750 CPU, and the program is coded using C++.

TABLE II

COMBINATIONS OF PARAMETER VALUES.

	Factor level			
Parameters	1	2	3	4
NP	20	50	80	100
SN	1	3	5	10
θ	0.01	0.05	0.1	0.15

TABLE III

ORTHOGONAL ARRAY AND ARV VALUE

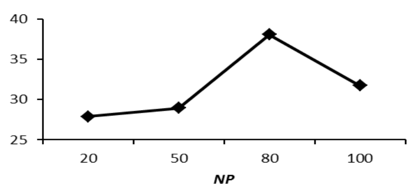
Experiment	Factors			ARV(%)
number	*NP*	*SN*	θ	
1	1	1	1	22.13
2	1	2	2	35.67
3	1	3	3	32.34
4	1	4	4	21.42
5	2	1	2	36.33
6	2	2	1	34.87
7	2	3	4	22.03
8	2	4	3	22.56
9	3	1	3	33.56
10	3	2	4	32.15
11	3	3	1	42.69
12	3	4	2	43.88
13	4	1	4	21.09
14	4	2	3	25.64
15	4	3	2	44.35
16	4	4	1	35.91

For each parameter combination, the ESFOA is run 20 times independently (set the maximum evaluation times as 100,000). The non-dominated solutions among all the solutions obtained by each run are collected as the reference set RS. The more solutions in RS obtained, the better the combination is. Thus, the average response variable (ARV) value is the average percentage of the non-dominated solutions for each combination in RS. According to the number of parameters and the number of factor levels, we choose the orthogonal array $L_{16}(4^3)$. The orthogonal array and the obtained ARV values are listed in Table III.

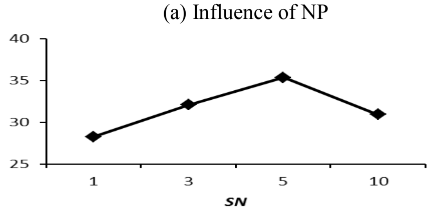
Using the statistical analysis tool Minitab, the significance rank of each parameter can be analyzed. Thus, he trend of each factor level is shown in Table IV and Fig 2, respectively.

TABLE IV
RESPONSE VALUE

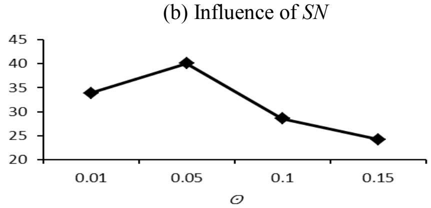
Level	*NP*	*SN*	θ
1	27.89	28.28	33.90
2	28.95	32.08	40.06
3	38.07	35.35	28.53
4	31.75	30.94	24.17
Delta	10.18	7.0	15.88
Rank	2	3	1



(a) Influence of NP



(b) Influence of *SN*



(c) Influence of θ

Fig. 2. Factor trend level of ESFOA

From Table IV and Fig 2, it can be seen that θ is the most significant parameter. A small value of θ is helpful for local exploitation. But a too small value may limit the search step, and waste the evaluation times. As for *NP*, with a fixed total evaluation times, a small size may lead to poor exploration so as to cause premature convergence, while a large size means a small number of evolution generations leading to insufficient deep search. As for *SN*, it can be seen from Fig 2 that it has a bit influence on the algorithm, where medium values are preferable. Thus, according to the results of the DOE test, we recommend to set parameters as *NP*=80, *SN*=5, and θ =0.05, which will be used for the following tests.

B. Results and Comparisons

The proposed ESFOA obtains not a single optimal solution, but a set of Pareto optimal solutions. If a decision maker only cares about one certain objective, like fuel cost or emission, the extreme point among the obtained Pareto front will be meaningful to provide information about the best cost or the best emission. In practical application, however, it is needed to select a solution from a set of the obtained Pareto front to tradeoff multiple objectives. Such a selected solution is called best compromise solution. For the EED problem, the best cost solution, the best emission solution and the best compromise solution are all used for comparisons [7-9, 11, 13].

Since the judgment of a decision-maker may not be precise, fuzzy set theory can be adopted to identify the candidate Pareto optimal solutions [11]. To compare the results, the same fuzzy mechanism as that in [7-9, 11, 13] is adopted to identify the best compromise solution. The satisfactory degree of the *i*-th solution $\bar{P}_i$ for the *j*-th objective function f_j is defined as the following membership function $\mu_{i,j}$:

$$\mu_{i,j} = \begin{cases} 1 & f_j\left(\bar{P}_i\right) \le f_{j\min} \\ \dfrac{f_{j\max} - f_j(\bar{P}_i)}{f_{j\max} - f_{\min}} & f_{j\min} < f_j\left(\bar{P}_i\right) < f_{j\max} \\ 0 & f_j\left(\bar{P}_i\right) \ge f_{j\max} \end{cases} \tag{20}$$

For the *i*-th solution, the normalized membership function μ_i is calculated as follows:

$$\mu_i = \sum_{j=1}^{m} \mu_{i,j} \Big/ \sum_{k=1}^{NQ} \sum_{j=1}^{m} \mu_{k,j} \tag{21}$$

where *NQ* is the number of the Pareto optimal solutions and *m* is the number of the objective functions. The best compromise solution is the one with the maximum μ.

During recent years, multi-objective optimization methods are commonly compared using the performance metric such as hyper-volume, two-set coverage, spacing etc. However, researchers in the field of power systems usually provide the best cost solution, the best emission solution and the best compromise solution for comparisons. Since the existing results of the EED problem are mainly from the field of power systems, we also use the best cost solution, the best emission solution and the best compromise solution for fair comparisons as done in [7-9, 11, 13]. It obtains the results by running the algorithm 20 times independently. The ESFOA is compared with some existing typical approaches, including NSGA [7], SPEA [8], NPGA [9], MODE [11], and MOPSO [13]. The comparative results of different approaches are listed in Table V-VII. Note that the results of the existing method are taken from the literature directly.

TABLE V
BEST COST SOLUTION

	MODE	**NPGA**	**SPEA**	**MOPSO**	**NSGA**	**ESFOA**
P_{G1}	0.1332	0.1425	0.1279	0.1207	0.1447	0.1163
P_{G2}	0.2727	0.2693	0.3163	0.3131	0.3066	0.2890
P_{G3}	0.6018	0.5908	0.5803	0.5907	0.5493	0.5778
P_{G4}	0.9747	0.9944	0.9580	0.9769	0.9894	1.0080
P_{G5}	0.5146	0.5315	0.5258	0.5155	0.5244	0.5111
P_{G6}	0.3617	0.3392	0.3589	0.3504	0.3542	0.3575
Emiss	0.2195	0.2207	0.2176	0.2193	0.2191	0.2217
Cost	606.13	608.06	607.86	607.79	607.98	**606.03**

TABLE VI
BEST EMISSION SOLUTION

	MODE	NPGA	SPEA	MOPSO	NSGA	ESFOA
P_{G1}	0.3927	0.4064	0.4145	0.4101	0.3929	0.3984
P_{G2}	0.4625	0.4876	0.4450	0.4594	0.3937	0.4433
P_{G3}	0.5631	0.5251	0.5799	0.5511	0.5815	0.5464
P_{G4}	0.4031	0.4085	0.3847	0.3919	0.4316	0.4277
P_{G5}	0.5676	0.5386	0.5348	0.5413	0.5445	0.5309
P_{G6}	0.4783	0.4992	0.5051	0.5111	0.5192	0.5215
Emiss	**0.1942**	0.1943	0.1943	0.1942	0.1947	**0.1942**
Cost	642.85	644.23	644.77	644.74	638.98	**641.95**

TABLE VII
BEST COMPROMISE SOLUTION

	MODE	NPGA	SPEA	MOPSO	NSGA	ESFOA
P_{G1}	0.2355	0.2976	0.2752	0.2367	0.2935	0.2980
P_{G2}	0.3489	0.3956	0.3752	0.3616	0.3645	0.3563
P_{G3}	0.5700	0.5673	0.5796	0.5887	0.5833	0.5339
P_{G4}	0.7252	0.6928	0.6770	0.7041	0.6763	0.6869
P_{G5}	0.5536	0.5201	0.5283	0.5635	0.5383	0.5584
P_{G6}	0.4261	0.3904	0.4282	0.4087	0.4076	0.4281
Emiss	0.2026	0.2004	0.2001	0.2021	0.2002	0.2002
Cost	613.27	617.79	617.57	615.00	617.80	617.21

As shown in Tables II and III, the ESFOA can obtain the best cost solution with minimum cost 606.03 $/hr and emission 0.2217 ton/hr as well as the best emission solution with minimum emission 0.1942 ton/hr and cost 641.95 $/hr. Although the MODE also can obtain a solution with minimum emission 0.1942 ton/hr, the cost is 642.85 $/hr which is larger than that of the ESFOA. That is, the solution obtained by the MODE is dominated by the one obtained by the ESFOA. As shown in Table IV, the best compromise solution obtained by the ESFOA is non-dominated by all the solutions obtained by other approaches; and among all the solutions, the cost of the solution by the ESFOA is the smallest. Since the enhanced non-dominated sorting emphasizes the optimization of the objective values rather than the crowding distance, the extreme points obtained by the ESFOA are better than those of the existing methods. Due to the preference of the enhanced non-dominance, the Pareto front obtained by the ESFOA is not so uniformly distributed as those by other crowding distance based algorithms. The Pareto fronts obtained by the MODE, NSGA, NPGA, SPEA, MOPSO and ESDFOA are illustrated in Fig. 3-8, respectively.

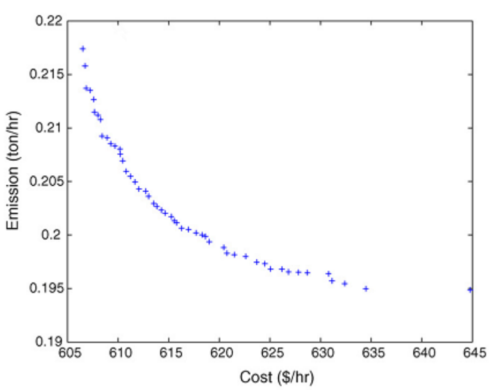


Fig. 3. Pareto front by NSGA [7]

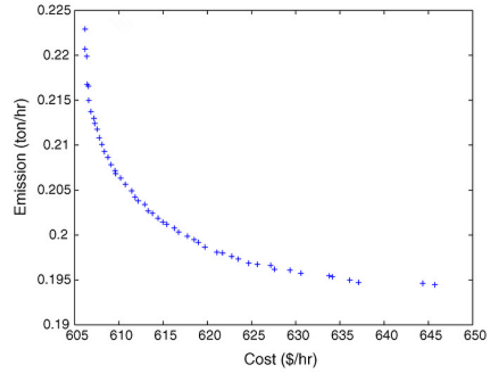


Fig. 4. Pareto front by NPGA [8]

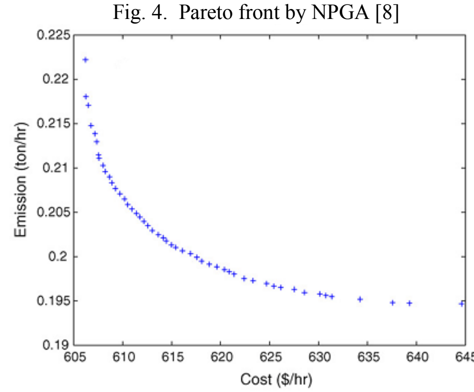


Fig. 5. Pareto front by SPEA [9]

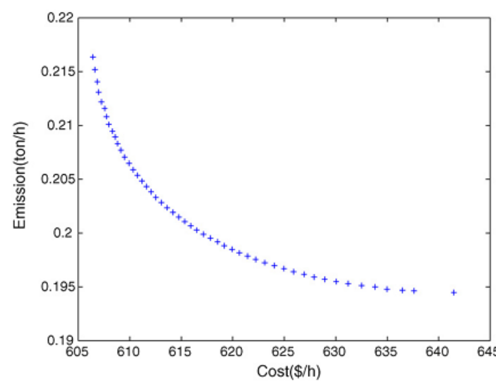


Fig. 6. Pareto front by MODE [11]

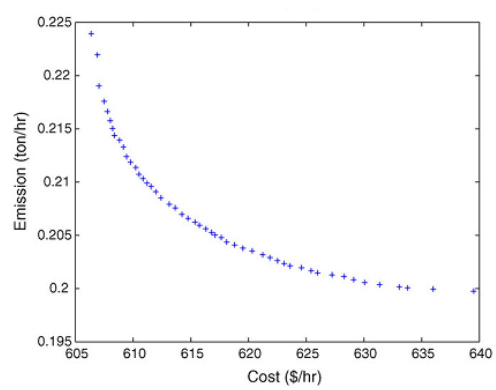


Fig. 7. Pareto front by MOPSO [13]

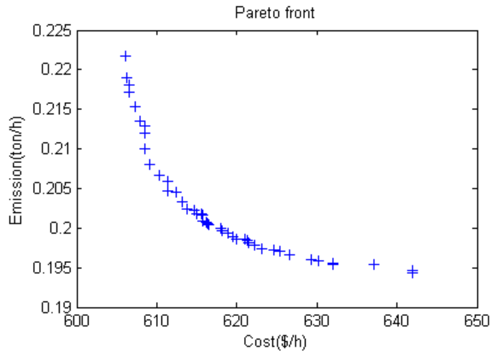


Fig. 8. Pareto front by ESFOA

As shown in Fig. 8, multiple non-dominated solutions with a well distribution can be obtained by the ESFOA. Although it cannot be concluded that the resulted front by the ESFOA is significantly better than others, the ESFOA can be a new approach to solve the EED problem which is easy to handle multiple objectives without using the crowding distance.

C. Ranking of results with TOPSIS

TOPSIS is one of the widely used multiple criteria decision methods [25], which provides a cardinal ranking of the candidate solutions making full use of the criteria information. To further demonstrate the effectiveness of the proposed ESFOA, the methods are ranked with the TOPSIS.

The core idea of TOPSIS is that the final alternative should be close to the ideal solution and away from the negative-ideal solution as much as possible. The procedure can be described as Fig. 8 [26]. The weight matrix indicates the preference to the objectives in TOPSIS. The weight matrix W_1=(0.01, 0.99) shows the least weight to emission, while W_2=(0.99, 0.01) shows the most weight to emission compared with the cost criteria. To use TOPSIS to rank the methods, the weight matrix W_1 for the best cost solution case, W_2 for the best emission case, and W_3=(0.5, 0.5) for the best compromise case are adopted, respectively. The results of TOPSIS for the three cases are listed in Table VIII, IX, X.

From Table VIII-X, it can be seen that the ESFOA ranks the first in all the three cases, which indicates that the ESFOA is effective in solving the EED problem.

TABLE VIII
RANK FOR THE BEST COST CASE

	F		D^+	D^-	R	Rank
MODE	0.2195	606.13	7.5E-5	0.0013	0.0554	2
NPGA	0.2207	608.06	1.6E-3	1.9E-5	0.9864	6
SPEA	0.2176	607.86	0.0012	1.5E-4	0.8882	4
MOPSO	0.2193	607.79	0.0012	1.9E-4	0.8636	3
NSGA	0.2191	607.98	0.0013	7.2E-5	0.9475	5
ESFOA	0.2217	606.03	7.6E-5	0.0014	0.0534	**1**

TABLE IX
RANK FOR THE BEST EMISSION CASE

	F		D^+	D^-	R	Rank
MODE	0.1942	642.85	2.5E-5	0.0010	0.0231	2
NPGA	0.1943	644.23	0.0002	0.0008	0.2020	4
SPEA	0.1943	644.77	0.0002	0.0008	0.2025	5
MOPSO	0.1942	644.74	3.7E-5	0.0010	0.0340	3
NSGA	0.1947	638.98	0.0010	3.7E-5	0.9659	6
ESFOA	0.1942	641.95	1.9E-5	0.0010	0.0178	**1**

Step 1: Normalize the decision matrix $F_{n\times m}$. The normalized value f_{ij} is calculated as:

$$f_{ij} = y_{ij} \Big/ \sqrt{\sum_{j=1}^{m} y_{ij}^2}\ ,\ i=1,2,\ldots,n \tag{22}$$

where n is the number of alternatives, m is the number of objectives or criteria.

Step 2: Calculate the weighted normalized decision matrix $Z_{n\times m}$.

$$z_{ij} = w_j \times f_{ij}\ ,\ i=1,2,\ldots,n, j=1,2,\ldots,m \tag{23}$$

where w_j is the weight of the j-th objective, and $\sum w_j = 1$.

Step 3: Determine the ideal and negative-ideal solutions according to $Z_{n\times m}$.

$S^+=\{s_1^+, s_2^+,\ldots, s_m^+\}=\{(\min z_{ij}|i=1,2,\ldots,n), j=1,2,\ldots,m\}$

$S^-=\{s_1^-, s_2^-,\ldots, s_m^-\}=\{(\max z_{ij}|i=1,2,\ldots,n), j=1,2,\ldots,m\}$

Step 4: Calculate the separation distances from the ideal solution (D_i^+, $i=1,2,\ldots,n$) and negative-ideal solution (D_i^-, $i=1,2,\ldots,n$) respectively.

$$D_i^+=\sqrt{\sum_j (s_j^+ - z_{ij})^2} \tag{24}$$

$$D_i^-=\sqrt{\sum_j (s_j^- - z_{ij})^2} \tag{25}$$

Step 5: Calculate the relative distance close to the ideal solution and away from the negative-ideal solution.

$$R_i=\frac{D_i^+}{D_i^+ + D_i^-} \tag{26}$$

Step 6: Rank the candidate solutions with an ascending order of R_i (i=1,2…n).

Fig. 8. Procedure of TOPSIS

TABLE X
RANK FOR THE BEST COMPROMISE CASE

	F		D^+	D^-	R	Rank
MODE	0.2026	613.27	0.0025	0.0015	0.6287	5
NPGA	0.2004	617.79	0.0015	0.0022	0.4060	4
SPEA	0.2001	617.57	0.0014	0.0025	0.3591	2
MOPSO	0.2021	615.00	0.0021	0.0011	0.6663	6
NSGA	0.2002	617.80	0.0015	0.0024	0.3814	3
ESFOA	0.2002	617.21	0.0013	0.0024	0.3485	**1**

VI. CONCLUSIONS

This paper deals with the EED problem in power system with an enhanced non-dominated sorting based fruit fly optimization algorithm. The main contributions of this work are as follows: define the concept of the enhanced non-dominance; present an enhanced non-dominated sorting algorithm; and develop the ESFOA to solve the EED problem. The numerical results with the IEEE 30-bus test system and the comparisons to the existing methods demonstrate the effectiveness of the ESFOA. This work not only enriches the application fields of the FOA, but also provides a new solver for the EED problem. Since the enhanced non-dominance sorting is easy but effective to handle multiple objectives, the future wok could focus on applying it to other multi-objective engineering optimization problems and generalizing the study to the combinatorial optimization problems like the production scheduling problems.

REFERENCES

[1] G. P. Granelli, M. Montagna, G. L. Pasini, and P. Marannino, “Emission constrained dynamic dispatch,” *Electric Power System*, vol. 24, pp. 56–64, 1992.

[2] R. Yokoyama, S. H. Bae, T. Morita, and H. Sasaki, “Multi-objective generation dispatch based on probability security criteria,” *IEEE Transaction on Power System*, vol. 3(1), pp. 317–324, 1988.

[3] S. Ozyon, H. Termutas, B. Durmus, and G. Kuvat, “Charged system search algorithm for emission constrained economic power dispatch problem,” *Energy*, vol. 46(1), pp. 420–430, Oct. 2012.

[4] C. Yasar, S. Ozyon, “Solution to scalarized environmental economic power dispatch problem by using genetic algorithm,” *Electrical Power and Energy System*, vol. 38, pp. 54-62, 2012.

[5] J. S. Dhillon, S. C. Parti, and D. P. Kothari, “Stochastic economic emission load dispatch,” *Electric Power Systems Research*, vol. 26, pp. 186–197, 1993.

[6] S. Rajasomashekar, P. Aravindhababu, “Biogeography based optimization technique for best compromise solution of economic emission dispatch,” *Swarm and Evolutionary Computation*, vol. 7, pp. 47–57, Dec. 2012

[7] M. A. Abido, “A novel multi-objective evolutionary algorithm for environmental/economic power dispatch,” *Electric Power Systems Research*, vol. 65, pp. 71–81, 2003.

[8] M. A. Abido, “Environmental/economic power dispatch using multi-objective evolutionary algorithms,” *IEEE Transaction Power System*, vol. 18, pp. 1529–1537, 2003.

[9] M. A. Abido, “A niched Pareto genetic algorithm for multi-objective environmental economic dispatch,” *Electric Power Energy System*, vol. 25(2), pp. 97–105, 2003.

[10] M. Basu, “Dynamic economic emission dispatch using non-dominated sorting genetic algorithm-II,” *Electric Power Energy Systems Research*, vol. 30(2), pp. 140–210, 2008.

[11] L. H. Wu, and Y. N. Wang, “Environmental/economic power dispatch problem using multi-objective differential evolution algorithm,” *Electric Power Systems Research*, vol. 80, pp. 1171–1181, 2010.

[12] Y. L. Lu, and J. Z. Zhou, “Environmental/economic dispatch problem of power system by using an enhanced multi-objective differential evolution algorithm,” *Energy Conversion and Management*, vol. 52(2), pp. 1175-1183, 2011.

[13] M. A. Abido, “Multi-objective Particle Swarm Optimization for Environmental Economic Dispatch Problem,” *Electric Power Systems Research*, vol. 79, pp. 1105–1113, 2009.

[14] D. W. Gong, Y. Zhang, and C. L. Qi, “Environmental/economic power dispatch using a hybrid multi-objective optimization algorithm,” *Electrical Power and Energy Systems*, vol. 32, pp. 607-614, 2010.

[15] Y. Zhang, D. W. Gong, Z. H. Ding, “A Bare-bones Multi-objective Particle Swarm Optimization Algorithm For Environmental/Economic Dispatch,” *Information Sciences*, vol. 192(1), pp. 213-227, 2012.

[16] N. Pandit, A. Tripathi, S. Tapaswi, and M. Pandit, “An improved bacterial foraging algorithm for combined static/dynamic environmental economic dispatch,” *Applied Soft Computing*, vol. 12, pp. 3500-3513, 2012.

[17] M. A. C. Silva, C. E. Klein, V. C. Mariani, L. S. Coelho, “Multiobjective scatter search approach with new combination scheme applied to solve environmental/economic dispatch problem,” *Energy*, vol. 53, pp. 14–21, 2013.

[18] H. T. Jadhav, R. Roy, “Gbest guided artificial bee colony algorithm for environmental/economic dispatch considering wind power,” *Expert Systems with Applications*, vol. 40(16), pp. 6385–6399, 2013.

[19] W. T. Pan, “A new fruit fly optimization algorithm: taking the financial distress model as an example,” *Knowledge-Based Systems*, vol. 26, pp. 69–74, 2012.

[20] S. M. Lin, “Analysis of service satisfaction in web auction logistics service using a combination of fruit fly optimization algorithm and general regression neural network,” *Neural Computing & Applications* vol. 22, pp.783-791, 2012.

[21] W. T. Pan, “Using modified fruit fly optimization algorithm to perform the function test and case studies,” *Connection Science,* vol. 25(2-3), pp. 151-160, 2013.

[22] W. Sheng, Y. Bao, “Fruit fly optimization algorithm based fractional order fuzzy-PID controller for electronic throttle,” *Nonlinear Dynamics*, vol. 73(1-2), pp. 611-619, 2013.

[23] D. Kundu, K. Suresh, and S. Ghosh, “Multi-objective optimization with artificial weed colonies,” *Information Science*, vol. 181(12), pp. 2441–24, 2011.

[24] D. C. Montgomery, 2005. Design and analysis of experiments, John Wiley &Sons, Arizona.

[25] M. Behzadiana, S. K. Otaghsara, M. Yazdani, J. Ignatius, “A state-of the-art survey of TOPSIS applications,” *Expert Systems with Applications*, vol. 39, pp. 13051–13069, 2012.

[26] S. Opricovic, G. Tzeng, “Compromise solution by MCDM methods: A comparative analysis of VIKOR and TOPSIS,” *European Journal of Operational Research*, vol. 156, pp. 445–455, 2004.