A Novel Multi-Objective Optimization Algorithm Based on Differential Evolution and NSGA-II

Fuqing Zhao[1], Liu Huan[1], Yi Zhang[2],Weimin Ma[3], and Chuck Zhang [4]
[1]School of Computer and Communication Technology, Lanzhou University of Technology Lanzhou 730050, China
[2]School of Mechnical Engineering, Xijin University Xi'an710123, China
[3]School of Economics and Management, Tongji University, Shanghai 200092, China
[4]H. Milton Stewart School of Industrial & Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA
Fzhao2000@hotmail.com; 799151253@qq.com; 1049440546@qq.com; mawm@tongji.edu.cn chuck.zhang@isye.gatech.edu

***Abstract*—NSGA-II is a well known, fast sorting and elite multi-objective genetic algorithm. The local exploitation ability of NSGA-II is relatively limited by the parameters of crossover and mutation. DE has shown powerful search abilities for continuous optimization. In this paper, an enhanced NSGA-II based on differential evolution and L-near distance (DP-NSGA-II/EDA) is proposed. To improve the diversity and convergence of Pareto optimal solutions by NSGA-II algorithm, DP-NSGA-II/EDA produces two populations by different approaches. One is from NSGA-II itself, the other is from differential evolution (DE). Through the competition between two populations, the superior individuals will be selected to construct new offspring population. Meanwhile, a new distance strategy called L-near distance is introduced to NSGA-II to maintain the diversity of the population. To validate the proposed algorithm, it is compared with the original NSGA-II, SPEA2 and MOEA/D-DE through several numerical benchmark problems. Results show the effectiveness of the proposed approach.**

Keywords—NSGA-II; dual-population; L - near distance; differential evolution; Pareto-optimal front

I. INTRODUCTION

A variety of optimization problems in the application require to optimize two or more conflicting objectives simultaneously, which are called multi-objective optimization problems (MOPs) With the advantage of a set of Pareto optimal solutions, evolutionary algorithms (EAs) have been found to be very successful in solving MOPs.

In the early 1990s, the first generation of MOEAs which were characterized by the use of selection mechanisms based on Pareto ranking and fitness sharing to maintain diversity were proposed. The representative algorithms includes the multi-objective genetic algorithm (MOGA) [1], the niched Pareto genetic algorithm (NPGA) [2] and the non-dominated sorting genetic algorithm (NSGA) [3]. The second generation of MOEAs using the elitism strategy had been presented. The major contributions includes the strength Pareto evolutionary algorithm (SPEA) [4] and the improved version of NSGA (NSGA-II) [5]. In recent years,. Many other search techniques have also been applied to solving MOPs, such as the simulated annealing algorithms [6], the particle swarm based algorithms [7], the ant colony based algorithms [8], the shuffled complex evolution algorithm [9]. In addition, Dual Population GA (DPGA) helps to provide additional population diversity to the main population by means of crossbreeding between the main population and reserve population. This helps to solve the problem of premature convergence and helps in early convergence of the algorithm [10].

In the past decade, NSGA-II is one of the most popular evolutionary multi-objective optimization(EMO) algorithms. At each generation of NSGA-II, it firstly uses non-dominated sorting to divide the hybrid population of parents and offspring into several non-domination levels. Starting from F1, one non-domination level is selected at a time to construct a new population, until its size equals or exceeds the limit. The exceeded solutions in the last acceptable non-domination level will be eliminated according to crowding distance. In this case, the selection strategy of NSGA-II implies its convergence first and diversity second behavior. With the advantage of NSGA-II, NSGA-II is very successful in solving MOPs. Therefore, NSGA-II is widely used in various fields and has a very important significance in solving practical problems. Bandyopadhyay et al. [11] introduced a mutation algorithm to embed in the NSGA-II, which caused the modification of the parts of the original NSGA-II. Du et al. [12] introduced JADE to NSGA-II aiming at refining the search area to maintain the population diversity.

Differential evolution algorithm (DE) is an optimization algorithm based on the theory of swarm intelligence [13].In this research, a hybrid differential evolution and NSGA-II based on L-near distance (DP-NSGA-II/EDA) is proposed. In DP-NSGA-II/EDA, two populations are produced by DE and NSGA-II. Through the competition between two populations, the superior individuals will be selected to construct new offspring population. Meanwhile, we develop L-near distance to maintain the diversity of the population. Finally, the results of the proposed algorithm have been compared with those of the original NSGA-II, SPEA2 and MOEA/D-DE. The experimental results show that the effectiveness of the proposed approach.

The remainder of the paper is organized as follows: Section 2, the main loop of DP-NSGA-II/EDA, with a particularly detailed description. Section 3 describes the experimental results. Section 4 makes a conclusion.

This work was financially supported by the National Natural Science Foundation of China under grant numbers 61663023. It was also supported by the Key Research Programs of Science and Technology Commission Foundation of Gansu Province (2017GS10817), Wenzhou public welfare science and technology project project(G20170016), General and Special Program of the Postdoctoral Science Foundation of China, the Science Foundation for Distinguished Youth Scholars of Lanzhou University of Technology under grant numbers 2012M521802, 2013T60889, and J201405, respectively.

978-1-5386-1482-2/18/$31.00 ©2018 IEEE

II. DP-NSGA-II/EDA

Many traditional multi-objective optimization algorithms employ only one crossover and mutation from beginning to end, which is easy to fall into local optimum. Another algorithm takes the self-adaptive strategy to avoid falling into local optimum. However, human intervention will increase and the stability is not strong. In different environments, the algorithm produces offspring populations through different crossover and mutation, which enables the populations to have better adaptability. It is not easy to fall into local optimum and better in converging to the true Pareto-optimal front. NSGA-II is one of the most outstanding multi-objective evolutionary algorithms, which are widely used in various fields and has a very important significance in solving practical problems. DE has shown powerful search abilities for continuous optimization. This paper combines NSGA-II and DE, which uses the simulated binary crossover and DE crossover to generate two offspring populations.

A. Crossover and mutation operator

The dual population is a vital step of DP-NSGA-II/EDA. In this paper, we define this dual population as generating two offspring through different crossover operation and two offspring have the same parent population. As a consequence, the offspring inherits the advantages from the same parent population, but each has its own unique advantage. Therefore, the search process is expected to strike a balance between global and local search. This paper uses the simulated binary crossover and the DE operator for offspring generation. Differential evolution (DE) has shown powerful search abilities for continuous optimization. Due to its advantage, DE facilitates the design of dual populations for exploiting guidance information towards the optima. Therefore, one offspring is generated by DE crossover and polynomial mutation[40]. The other is generated by simulated binary crossover and polynomial mutation.

Offspring1:

DE crossover :

While $i < N$ (N is the size of P_t)

$x^i, x^{r1}, x^{r2} \in P_t$

for $j < M$ (M is the size of x)

$$u_{ij} = \begin{cases} x_j^i + F\left(x_j^{r1} - x_j^{r2}\right), & if\ rand < CR \quad or \quad j = j_{rand} \\ x_j^i, & otherwise \end{cases}$$

$j = j + 1$

$i = i + 1$

$p_1 = p_1 \cup u_i$

Polynomial mutation :

While $t < N$ (N is the size of P_t)

$x_i, x_j \in P_t$

$$x_t = \begin{cases} u_j + \sigma_j\left(x_i - x_j\right) & if \quad rand < p_m \\ u_j & otherwise \end{cases}$$

$t = t + 1$

$p_1 = p_1 \cup x_t$

Offspring2:

Simulated Binary crossover :

While $k < N/2$ (N is the size of P_t)

$x_i, x_j \in P_t$

$c_1 = 0.5\left[\left(x_i + x_j\right) - \beta\left|x_i - x_j\right|\right]$

$c_2 = 0.5\left[\left(x_i + x_j\right) + \beta\left|x_i - x_j\right|\right]$

$k = k + 1$

$p_2 = p_2 \cup c_1 \cup c_2$

Polynomial mutation :

While $t < N$ (N is the size of P_t)

$x_i, x_j \in P_t$

$$x_t = \begin{cases} u_j + \sigma_j\left(x_i - x_j\right) & if \quad rand < p_m \\ u_j & otherwise \end{cases}$$

$t = t + 1$

$p_2 = p_2 \cup x_t$

B. L-near Distance and Dynamic Selection

To get an estimate of the density of solutions surrounding a particular solution in the population, we calculate the L-near distance of individuals. DP-NSGA-II/EDA replaces crowded-comparison with L-near distance. The L-near distance of individual is required to change dynamically in the selection process.

- L-near Distance

L-near distance is calculated as the sum of distance values corresponding to each point and its nearest neighbors. L is defined as the number of nearest neighbors and decided by the probability.

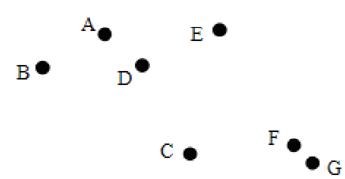

Fig.2. L-near distance calculation

The value of L is decided by probability. And the value of L is 1 or 2. If the probability is more than 0.2, L is selected to be 2. Otherwise, L is selected to be 1. As shown in Fig.3, L is selected to be 2. The L-near distance of individual is as follows. $D_A = |AB| + |AD|$, $D_B = |AB| + |BD|$, L, $D_G = |FG| + |CG|$.

- Dynamic Selection

The value of L is 1 or 2, is chosen according to the probability. The value of L has an 80 percent chance of being 2. On the contrary, the value of L has a 20 percent chance of being 1. As shown in Fig.4, L-near distance is used to eliminate individuals dynamically. The algorithm generates randomly a number between zero and one. If the number is more than 0.2, the L is 2 and the algorithm eliminates the individual A according to 2-near distance. If the number is no more than 0.2, the L is 1 and the algorithm eliminates the individual G according to 1-near distance.

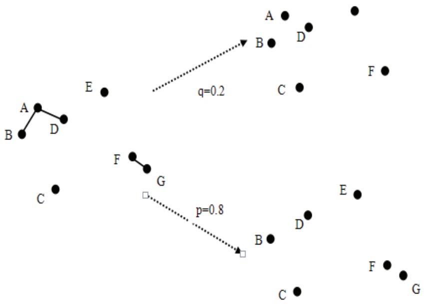

Fig.3. Eliminating dominated individual dynamically

C. Basic idea and DP-NSGA-II/EDA framework

DP-NSGA-II/EDA retains the basic framework of NSGA-II. To improve convergence of the NSGA-II, DP-NSGA-II/EDA proposes the strategy of dual-population evolution. The simulated binary crossover and DE crossover are selected to generate two offspring populations. The two new offspring populations are compared with each other to form non-dominated solutions of the offspring population, which enables the population to have better adaptability. In addition, we develop L-near distance to maintain the diversity of the population. Specific steps of DP-NSGA-II/EDA are as follows.

Step 1 Initialize the population POP_0, the population POP_0 is sorted according to non-domination.

Step 2 The binary tournament selection, DE crossover and Polynomial mutation are used to create offspring p_1^t. Simulated binary crossover and Polynomial mutation are used to create offspring p_2^t.

Step 3 The two offspring populations p_1^t and p_2^t are merged into one population P_t. The population P_t is sorted according to non-domination. Then, the new population Q_t, which is of size N, is selected from the population P_t.

$P = p_1^t \cup p_2^t$

$P_t = fast-non-do\min ated-sort(P_t)$

$Q_t = \varnothing$ and $i = 1$

While $i \le N$

$\quad Q_t = Q_t \cup P_t^i \qquad (P_t^i$ is the ith of $P_t)$

$\quad i = i + 1$

Step 4 A combined population $R_t = POP_t \cup Q_t$ is formed. The population is of size $2N$. Then, the population is sorted according to non-domination. The population is divided into dominated set $F_1, F_2, \ldots, F_l$. The new population POP_{t+1} is selected from F_1 to F_l based on the L-near distance sorting. The population POP_{t+1} is of size N.

$R_t = P_t \cup Q_t$

$F = fast-non-do\min ated-sort(R_t)$

$P_{t+1} = \varnothing$ and $i = 1$

until $|P_{t+1}| + |F_i| \le N$

$\quad$ L-near-distance-assignment(F_i)

$\quad P_{t+1} = P_{t+1} \cup F_i$

$\quad i = i + 1$

Sort(F_i, p_n)

$P_{t+1} = P_{t+1} \cup F_i[1 : (N - |P_{t+1}|)]$

$Q_{t+1} =$ make-new-pop(P_{t+1})

$t = t + 1$

Step 5 If $gen = gen_{\max}$, the loop is over Otherwise, $gen = gen + 1$, $t = t + 1$, go to Step2

III. Computational Results and Comparisons

A. performance metrics

There are two goals in a multi-objective optimization: convergence to the Pareto-optimal set and maintenance of diversity in solutions of the Pareto-optimal set. This paper adopts five performance metrics to evaluate algorithm. IGD [14] and HV [15] are the more popular comprehensive performance metrics. Both IGD and HV metrics can measure the convergence and diversity of population simultaneously.SP (Spacing) is an evaluation of uniformity.(Maximum spread) is a metrics of spread[16]. It is used to measure the extent of coverage that the non-dominated solution covers the true Pareto front. IGD (Inverted Generational Distance) is one of the comprehensive evaluations. HV (Hypervolume) is used to calculate the volume which is covered by domination individuals in the domain of objectives.

B. Parameters setting

We evaluate the performance of the proposed DP-NSGA-II/EDA on a set of test problems, which include SCH [16], KUR [17], BK1, FF1, MOP2, MOP7 [18] as well as series of ZDT [19]. For each of test problems, it needs to be calculated the mean and variance of the experimental data which experiments were repeated 20 times.

When each algorithm runs the test problems, overall population size becomes 100 and evolution generations of all algorithms are as follows.

- The test problems (ZDT1 , ZDT2 , ZDT3 , BK1 , FF1 , MOP2): Evolution generations are 100.
- The test problems (SCH , KUR , ZDT6 , MOP7): Evolution generations are 200.

When HV is calculated, it is required to set the reference point. The reference point about SCH is (1, 17). The reference point about KUR is (-14, 1). The reference point about BK1 is

(50, 50). The reference point about FF1, MOP2 and series of ZDT is (1, 1). The reference point about MOP7 and series of DTLZ is (1, 1, 1). Table II showsthe detailed parameter settings. Table 1 shows the specific parameter settings of the above test problems. n is the number of the decision variable.

C. Results and comparison with other algorithms

Table 2-3 shows the performance comparisons of DP-NSGA-II/EDA, NSGA-II, SPEA2 and MOEA/D-DE, regarding the four metrics, respectively. We find that DP-NSGA-II/EDA shows a clearly better performance than NSGA-II, SPEA2 and MOEA/D-DE. Especially it obviously achieves better metric values in SP, IGD and HV. MOEA/D-DE shows a clearly better performance than others. It is worth noting that although the performance of SPEA2 is not satisfied on most test instances, it obtains better metric values in SP. By contrast, NSGA-II shows a little poor performance in SP. Therefore, SPEA2 and DP-NSGA-II/EDA has a good ability to maintain the population diversity. DP-NSGA-II/EDA shows significantly better performance than both of its baseline algorithms. These observations demonstrate the effectiveness of DP-NSGA-II/EDA for balancing convergence and diversity.

Table 1 Parameter settings of the EMO algorithms.

MOEAs	Parameter settings
NSGA-II	$p_c = 0.9$, $p_m = 1/n$, $u_c = 15$, $u_m = 20$
SPEA2	$p_c = 0.9$, $p_m = 1/n$, $u_c = 15$, $u_m = 20$
DP-NSGA-II/EDA	$p_c = 0.9$, $p_m = 1/n$, $u_c = 15$, $u_m = 20$, $CR = 1.0$, $F = 0.5$
MOEA/D-DE	$p_m = 1/n$, $u_m = 20$, $CR = 1.0$, $F = 0.5$ $\delta = 0.9$, $T = 20$, $n_r = 2$

Table 2 Performance comparisons on SCH, KUR, BK1, FF1, MOP2

Problem	Algorithm	SP	MS	IGD	HV
SCH	SPEA2	1.32E-02	1.00E+00	2.23E-03	2.13E+01
		8.47E-04	0.00E+00	1.38E-04	6.02E-03
	NSGA-II	5.26E-02	1.00E+00	2.54E-03	2.12E+01
		3.69E-04	0.00E+00	1.07E-04	5.92E-03
	MOEA/D-DE	4.30E-02	1.00E+00	**2.22E-03**	2.12E+01
		4.42E-03	0.00E+00	**9.16E-05**	5.09E-03
	DP-NSGA-II/EDA	**1.28E-02**	**1.00E+00**	2.28E-03	**2.13E+01**
		1.88E-03	**0.00E+00**	1.09E-04	**1.80E-03**
KUR	SPEA2	5.34E-02 ±	9.99E-01	1.49E-03	3.70E+01
		2.40E-03	3.65E-05	1.63E-04	2.33E-02
	NSGA-II	7.54E-02 ±	1.00E+00	1.81E-03	3.69E+01
		1.98E-02	1.28E-04	1.87E-04	2.46E-02
	MOEA/D-DE	8.24E-02	1.00E+00	1.60E-03	3.70E+01
		1.85E-02	4.65E-05	9.85E-05	1.68E-02
	DP-NSGA-II/EDA	**4.00E-02**	1.00E+00	**1.34E-03**	**3.71E+01**
		2.58E-02	7.11E-05	**1.16E-04**	**1.13E-02**
BK1	SPEA2	1.35E-01	9.96E-01	2.14E-02	2.07E+03
		1.51E-02	3.06E-04	1.84E-03	4.16E-01
	NSGA-II	3.59E-01	9.98E-01	9.72E-03	2.07E+03
		1.88E-02	7.33E-04	1.99E-04	7.69E-01
	MOEA/D-DE	3.01E-01	**9.98E-01**	8.38E-03	2.07E+03
		4.04E-02	**5.23E-05**	2.31E-03	7.97E-01
	DP-NSGA-II/EDA	**1.33E-01**	9.98E-01	**5.60E-03**	**2.07E+03**
		2.23E-02	2.40E-04	**5.60E-04**	**3.41E-01**
FF1	SPEA2	2.95E-03	1.00E+00	1.48E-04	6.22E-03
		4.81E-04	3.65E-05	7.49E-06	1.82E-04
	NSGA-II	8.27E-03	1.00E+00	2.03E-04	6.12E-02
		5.81E-04	1.27E-05	1.84E-05	1.81E-04
	MOEA/D-DE	6.18E-03	**1.00E+00**	1.75E-04	6.18E-02
		7.44E-04	**2.23E-07**	2.67E-05	2.23E-04
	DP-NSGA-II/EDA	**2.77E-03**	1.00E+00	**1.43E-04**	**6.22E-02**
		4.16E-04	3.78E-05	**8.95E-06**	**2.21E-04**
MOP2	SPEA2	**2.11E-03**	1.00E+00	1.20E-03	3.36E-01
		2.09E-04	2.41E-04	1.09E-05	2.35E-04
	NSGA-II	6.27E-03	**1.00E+00**	1.18E-04	3.33E-02
		3.08E-04	**0.00E+00**	2.89E-05	2.65E-04
	MOEA/D-DE	5.05E-03	**1.00E+00**	1.20E-03	3.34E-01
		3.62E-04	**0.00E+00**	5.67E-05	5.01E-04
	DP-NSGA-II/EDA	2.26E-03	**1.00E+00**	**1.18E-04**	**3.36E-01**
		5.27E-04	**0.00E+00**	**1.34E-05**	**2.38E-04**

Table 3 Performance comparisons on MOP7, ZDT1, ZDT2, ZDT3, ZDT6

Problem	Algorithm	SP	MS	IGD	HV
MOP7	SPEA2	1.22E-02	9.96E-01	8.48E-04	**8.71E-01**
		9.36E-04	2.57E-04	2.01E-04	**6.52E-04**
	NSGA-II	1.38E-02	9.93E-01	9.50E-04	8.58E-01
		2.20E-03	5.74E-03	5.38E-04	2.86E-03
	MOEA/D-DE	1.24E-02	**9.97E-01**	8.45E-04	8.54E-01
		2.22E-03	**3.14E-04**	4.32E-04	6.67E-03
	DP-NSGA-II/EDA	**4.31E-03**	9.95E-01	8.21E-04	8.70E-01
		7.60E-04	9.61E-04	2.10E-04	7.38E-04
ZDT1	SPEA2	2.53E-03	9.99E-01	3.56E-04	3.65E+00
		1.85E-04	1.12E-03	2.03E-05	2.67E-03
	NSGA-II	5.41E-03	1.00E+00	1.94E-04	3.66E+00
		5.97E-04	2.94E-04	3.29E-05	5.23E-04
	MOEA/D-DE	5.66E-03	9.99E-01	1.50E-04	3.66E+00
		9.36E-04	6.22E-04	2.65E-05	2.97E-04
	DP-NSGA-II/EDA	**2.09E-03**	1.00E+00	**1.43E-04**	**3.66E+00**
		2.23E-04	1.10E-07	**8.21E-06**	**9.70E-05**
ZDT2	SPEA2	2.40E-03	9.99E-01	5.24E-05	3.33E+00
		1.51E-04	6.85E-04	1.37E-06	3.10E-03
	NSGA-II	5.62E-03	9.99E-01	5.14E-05	3.32E+00
		4.22E-04	1.99E-03	4.36E-06	2.10E-03
	MOEA/D-DE	5.56E-03	9.97E-01	4.58E-05	3.33E+00
		8.85E-04	1.91E-03	3.68E-06	2.88E-04
	DP-NSGA-II/EDA	**2.00E-03**	**1.00E+00**	**4.51E-05**	**3.33E+00**
		1.12E-04	**3.68E-05**	**1.62E-06**	**1.79E-05**
ZDT3	SPEA2	3.32E-03	9.99E-01	3.64E-04	4.81E+00
		2.10E-04	4.42E-04	2.67E-05	1.53E-01
	NSGA-II	7.11E-03	9.93E-01	**3.31E-04**	4.80E+00
		7.96E-04	3.75E-02	**2.14E-05**	6.70E-02
	MOEA/D-DE	6.86E-03	1.00E+00	3.48E-04	4.81E+00
		9.00E-04	4.31E-05	1.57E-05	1.12E-01
	DP-NSGA-II/EDA	**2.78E-03**	**1.00E+00**	3.43E-04	**4.82E+00**
		2.98E-04	**8.03E-05**	1.68E-05	**5.32E-05**
ZDT6	SPEA2	2.86E-03	9.94E-01	8.11E-04	3.02E+00
		3.61E-04	1.05E-03	1.18E-04	3.82E-03
	NSGA-II	4.98E-03	9.97E-01	5.64E-04	3.03E+00
		4.23E-04	2.31E-05	1.24E-05	6.67E-04
	MOEA/D-DE	4.59E-03	9.98E-01	5.10E-04	3.03E+00
		3.84E-03	7.85E-05	1.92E-05	1.39E-03
	DP-NSGA-II/EDA	**1.74E-03**	**9.99E-01**	**4.55E-04**	**3.04E+00**
		2.48E-04	**9.92E-05**	**1.50E-05**	**3.15E-04**

IV. Conclusions

This paper presents a hybrid differential evolution and NSGA-II based on L-near distance (DP-NSGA-II/EDA), which combines NSGA-II and DE. DE mutation strategy is embedded in DP-NSGA-II/EDA algorithm to enhance the ability of local search. DP-NSGA-II/EDA produces two offspring populations by different crossover and mutation. Through the competition between two offspring populations, the next of advantage group is selected. Besides, we develop L-near distance to maintain the diversity of the population. DP-NSGA-II/EDA has a better adaptability, which is not caught in local optimum through the dual-population evolutionary, as well as having a good effect in terms of convergence. In order to verify the performance of DP-NSGA-II/EDA, fifteen functions are used in the experiments. To further testify the performance of DP-NSGA-II/EDA, DP-NSGA-II/EDA is also compared with other three algorithms including NSGA-II, MOEA/D-DE and SPEA2. Empirical results demonstrate that DP-NSGA-II/EDA outperforms three other algorithms: NSGA-II, MOEA/D-DE and SPEA2 in terms of finding a diverse set of solutions and in converging near the true Pareto-optimal set.

However, DP-NSGA-II/EDA reflects certain deficiencies in terms of the broad measure. We also notice that the complexity time of DP-NSGA-II/EDA is more than NSGA-II.

Acknowledgment

This work was financially supported by the National Natural Science Foundation of China under grant numbers 61663023. It was also supported by the Key Research Programs of Science and Technology Commission Foundation of Gansu Province (2017GS10817), Wenzhou public welfare science and technology project (G20170016), General and Special Program of the Postdoctoral Science Foundation of China, the Science Foundation for Distinguished Youth Scholars of Lanzhou University of Technology under grant numbers 2012M521802, 2013T60889, and J201405, respectively.

REFERENCE

1 Fonseca, C.M., and Fleming, P.J.: 'Genetic Algorithms for Multiobjective Optimization: FormulationDiscussion and Generalization', in Editor (Ed.)^(Eds.): 'Book Genetic Algorithms for Multiobjective Optimization: FormulationDiscussion and Generalization' (1993, edn.), pp. 416-423

2 Horn, J., Nafpliotis, N., and Goldberg, D.E.: 'A Niched Pareto Genetic Algorithm for Multiobjective Optimization' (1994. 1994)

3 Srinivas, N., and Deb, K.: 'Muiltiobjective Optimization Using Nondominated Sorting in Genetic Algorithms' (MIT Press, 1994. 1994)

4 Zitzler, E., and Thiele, L.: 'Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach', IEEE Transactions on Evolutionary Computation, 1999, 3, (4), pp. 257-271

5 Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T.: 'A fast and elitist multiobjective genetic algorithm: NSGA-II', IEEE Transactions on Evolutionary Computation, 2002, 6, (2), pp. 182-197

6 Li, H., and Landa-Silva, D.: 'An adaptive evolutionary multi-objective approach based on simulated annealing', Evolutionary Computation, 2014, 19, (4), pp. 561-595

7 Zhao, F., Tang, J., Wang, J., and Jonrinaldi: 'An improved particle swarm optimization with decline disturbance index (DDPSO) for multi-objective job-shop scheduling problem', Computers & Operations Research, 2014, 45, (45), pp. 38-50

8 Rada-Vilela, J., Chica, M., Cordón, Ó., and Damas, S.: 'A comparative study of Multi-Objective Ant Colony Optimization algorithms for the Time and Space Assembly Line Balancing Problem', Applied Soft Computing Journal, 2013, 13, (11), pp. 4370-4382

9 Zhao, F., Zhang, J., Zhang, C., and Wang, J.: 'An improved shuffled complex evolution algorithm with sequence mapping mechanism for job shop scheduling problems', Expert Syst. Appl., 2015, 42, (8), pp. 3953-3966

10 Umbarkar, A.J., Joshi, M.S., and Hong, W.C.: 'Multithreaded Parallel Dual Population Genetic Algorithm (MPDPGA) for unconstrained function optimizations on multi-core system' (Elsevier Science Inc., 2014. 2014)

11 Bandyopadhyay, S., and Bhattacharya, R.: 'Solving a tri-objective supply chain problem with modified NSGA-II algorithm', Journal of Manufacturing Systems, 2014, 33, (1), pp. 41-50

12 Du, W., Leung, S.Y.S., and Kwong, C.K.: 'Time series forecasting by neural networks: A knee point-based multiobjective evolutionary algorithm approach', Expert Syst. Appl., 2014, 41, (18), pp. 8049-8061

13 Awad, N.H., Ali, M.Z., and Suganthan, P.N.: 'Ensemble sinusoidal differential covariance matrix adaptation with Euclidean neighborhood for solving CEC2017 benchmark problems', in Editor (Ed.)^(Eds.): 'Book Ensemble sinusoidal differential covariance matrix adaptation with Euclidean neighborhood for solving CEC2017 benchmark problems' (2017, edn.), pp. 372-379

14 Pasandideh, S.H.R., Niaki, S.T.A., and Asadi, K.: 'Bi-objective optimization of a multi-product multi-period three-echelon supply chain problem under uncertain environments: NSGA-II and NRGA', Information Sciences, 2015, 292, (C), pp. 57-74

15 Knowles, J.: 'A Tutorial on the Performance Assessment of Stochastic Multiobjective Optimizers; Third International Conference on Evolutionary Multi-Criterion Optimization (EMO 2005)', 2005

16 Rajagopalan, R., Mohan, C.K., Mehrotra, K.G., and Varshney, P.K.: 'EMOCA: An Evolutionary Multi-Objective Crowding Algorithm', Journal of Intelligent Systems, 2008, 17, (1-3), pp. 107-124

17 Qi, Y., Hou, Z., Yin, M., Sun, H., and Huang, J.: 'An immune multi-objective optimization algorithm with differential evolution inspired recombination', Appl. Soft. Comput., 2015, 29, (C), pp. 395-410

18 Liu, H.L., Gu, F., and Zhang, Q.: 'Decomposition of a Multiobjective Optimization Problem Into a Number of Simple Multiobjective Subproblems', IEEE Transactions on Evolutionary Computation, 2014, 18, (3), pp. 450-455

19 Maashi, M., Özcan, E., and Kendall, G.: 'A multi-objective hyper-heuristic based on choice function', Expert Syst. Appl., 2014, 41, (9), pp. 4475-4493