A Many-Objective Squirrel Hybrid Optimization Algorithm: MaSHOA

Zhuoran Liu, Fanhao Zhang, Xinyuan Wang, Qidong Zhao, Changsheng Zhang, and Bin Zhang(✉)

Northeastern University, Shenyang 110819, People's Republic of China
zhangchangsheng@mail.neu.edu.cn, paper820@sohu.com

Abstract. Many-objective optimization problems (MaOP) are important to the field of computing intelligence which leads to more requirements for the evolutionary many-objective Algorithms (EMaOA). Meanwhile, we consider that the evolution process also has some influence on the performance of results. And we present a many-objective squirrel hybrid optimization algorithm (MaSHOA) which takes an effective squirrel search algorithm (SSA) as the evolution framework and a reference-point-based many-objective evolutionary algorithm (NSGA-III) as the EMaOA framework. This paper applies the scalarizing evaluation to make sure the solution quality among the neighborhood and takes the reference point association achievement as the reference-point-based part. Taking iterations into account, we design a joint fitness function. For both the evolution and selection operations, a joint fitness function is applied to sort solutions to guide others and select them respectively. Besides, the distance penalization is introduced to prevent the local convergence. About useless reference points, this paper proposes an adjustable reference points strategy. The simulation experiment of the proposed algorithm is carried on different test problems with 3 to 15 objectives. Compared with other classic EMaOAs, the means, variances, box plots and parallel coordinate plots of the obtained results are utilized to analyze the convergence and diversity. And this proposed algorithm has good performance on solving MaOPs.

Keywords: Many-objective optimization · Squirrel search algorithm · Adjustable reference points strategy

1 Introduction

In recent years, with the development of computer technology and the prosperity of artificial intelligence, computing intelligence based on computer technology has developed rapidly [1]. As an important branch of computational intelligence, applying evolutionary computation to solve multi-objective optimization problems has become a research hotspot in the field of computational intelligence [2]. However, the calculations required for complicated projects no longer consider only one single indicator but consider multiple indicators that are mutually constrained. Optimization problems with two or more objectives are often referred to as Multi-objective Optimization Problems (MOPs) [3].

H. Song and D. Jiang (Eds.): SIMUtools 2020, LNICST 369, pp. 442–459, 2021.
https://doi.org/10.1007/978-3-030-72792-5_36

And we take the MOPs with four or more objectives into consideration as a special optimization problem known as the Many-objective Optimization Problem (MaOP) [4]. Algorithms based on Pareto domination are widely accepted in solving multi-objective optimization problems [3, 5]. But when it comes to MaOPs, the non-dominated solution set obtained by the traditional many-objective optimization algorithms based on Pareto domination is not of great quality. In recent years, with many relevant strategies proposed, difficulties [4, 6] in solving MaOPs are also exposed:

1) Most of the solutions in the population of evolutionary algorithms are non-dominated, which causes that the ascendancy of two different solutions becomes ambiguous.
2) The exponential growth of the number of non-dominated solutions is a huge challenge to the processing power of the algorithm.
3) Visualization of high-dimensional solutions becomes difficult. It's tough for decision makers to understand the distribution of the solution and how to evaluate it.

Based on the above reasons, some Evolutionary Many-objective Algorithms (EMaOA) using special environment selection strategies have attracted widespread attention in MaOP research due to their advantages of fast solution speed and wide application range [7]. Used to solve ultra-multi-objective optimization problems. These algorithms can be broadly classified as follows:

1) Algorithms based on Pareto-dominated [6–8] are proposed. NSGA-III [9, 10] is a typical example of a dominated many-objective algorithm that improves the ranking method, which also continues the attempt to use reference points in MONSGA-II [11]. Besides, θ-DEA [12] is a typical algorithm for improving Pareto domination rules.
2) Decomposition-based EMaOA [7, 13] decomposes MaOP into multiple single-objective subproblems which cover the decision space, and solves the subproblems independently, the algorithm uses the optimal solutions of all subproblems to fit the pareto front (PF), like MOEA/D [14]. And some algorithms presented some new concepts into the solving process. RVEA [15] introduces a scalar method called angle-penalized distance (APD) to evaluate the convergence and diversity of candidate solutions.
3) Select the subset with the best index value in the population [7, 16]. Performance indicators can usually evaluate the convergence and diversity of the population at the same time. For example, HypE [17] uses HV indicator, MOMBI-II [18] uses R2 indicator, and MaOEA/IGD [19] uses IGD indicator.

The method of evaluating many-objective optimization algorithms is to analyze their convergence and diversity. Convergence refers to finding a set of solutions closed to the true Pareto front. Meanwhile, diversity means finding a set of solutions that should be sufficient to represent the entire range of Pareto front. Algorithms are usually evaluated by these two types of indicators.

In addition, some scholars are committed to applying some classic and efficient single-objective optimization algorithms to more objects [5, 8, 9, 20–22], such as NSGA-III with genetic algorithm (GA) [23]. Among them, the squirrel search algorithm (SSA) [24], as a single-objective optimization algorithm [25], has the characteristics of strong robustness and fast convergence. To a certain extent, it avoids the dimensional catastrophe problem that exists in many-objective optimization problems.

Based on the many-objective optimization framework of NSGA-III and the evolution framework of SSA, this paper presents a many-objective optimization hybrid squirrel search algorithm (MaSHOA). The contributions are outlined as following:

- This algorithm proposes scalarizing evaluation to make sure the convergence of population members in the neighborhood. And the reference point association achievement presents the effect that associated reference points produce on the candidate solutions.
- The influence of the number of current generations on optimization focus is introduced to design a joint fitness function which combines scalarizing evaluation and reference point association achievement.
- In the evolutionary process, squirrels move forward the direction of best ones rated by the joint fitness function. And also, the distance penalization is applied in the winter detection to prevent the local convergence.
- For selection operator, the members of feasible solution set are selected following the sorting obtained by this joint fitness function.
- This algorithm also presents an adjustable reference points strategy to change some reference points without associated solutions into some solutions by considering the distance between solutions and reference points.

Finally, this paper designs a simulation experiment to compare the presented algorithm with NSGA-III and MOEA/D on the DTLZ test problems [26]. The inverse generational distance (IGD) metric [19, 27] is applied to analyze the results by numbers and box plots, and also the parallel coordinate plots of PF are presented to visualize the performance of algorithms. It turns out that the proposed algorithm has good performance on convergence and diversity.

2 Design of MaSHOA

2.1 Basic Concept of MaSHOA

This paper proposes a Many-Objective Optimization Hybrid Squirrel Search Algorithm (MaSHOA), which utilizes NSGA-III for reference of many-objective optimization framework and integrates SSA into it. The basic framework of MaSHOA is similar to the original NSGA-III, the selection and mutation operator is modified with some strategies.

The algorithm execution process is shown in Algorithm 1. Before starting the process, the reference points set is calculated. First, a population is initialized randomly named P. Before the stop criteria are achieved, the proposed algorithm runs the following actions.

Normalization of population members, the association operation and the niche-preservation operation according to the original NSGA-III is executed first (line 4–6).

And the number P_j of population members that are associated with the certain reference point that associates the individual is obtained for each member.

Based on the above results, the reference point association achievement and scalarizing evaluation is calculated (line 8). And after the number of current generations is affiliated, a joint fitness function is created according to Reference Point Association Achievement and Scalarizing Evaluation (line 9).

Algorithm 1. Procedure for MaSHOA

Input:
Population number N, test problem, parameter definition
Output
Feasible solution set
Begin
1. Generate reference points;
2. Initialized population randomly P;
3. **do while** stop criteria==false
4. Normalization of population members;
5. Association operation;
6. Niche-preservation operation;
7. Calculate the number of population members that are associated with the certain reference point that associates the member p_j;
8. Generate Scalarizing Evaluation and Reference Point Association Achievement according to Eq. 1& Eq. 3;
9. Calculate joint fitness function according to Section 2.2;
10. Squirrels evolutionary to new population set according to Section 2.3;
11. Archive current population as set S;
12. Merge two populations P and S into M;
13. Non-dominated sorting;
14. Choose individuals by the level of non-dominated sorting in turn until the current level L makes the size of population more than N, still choose all the members of level L into the current population P (size > N);
15. Select N of the current population members by the sorting of the joint fitness function;
16. Exchange reference points by an adjustable strategy according to Section 2.4;
17. External archive current population as set P;
18. **end while**
19. **return** feasible solution set;
End

Then during population evolution, the original squirrel search algorithm is designed to solve single-objective optimization problems. For the MOPs, SSA is modified in this algorithm. For the migration operator, squirrels are evaluated by a joint fitness function to sort (line 10). Besides, the distance penalization is introduced into winter detection to avoid solutions from local convergence. And then the external archive is established to store the current population as S (line 11). And the current and previous population are merged to be applied to the next selection operator to maintain the elite information as M (line 12).

Next, the non-dominated sorting is applied to select current members in non-dominated level order into population P until members with the current level L are selected to make the size of P more than N. It is changed from the original process that the members with the current level L are also added into P (line 14).

For the selection of parent and offspring individuals, the original framework utilizes the number of population members that are associated with each reference point which associates each individual as the selection condition after non-dominated sorting. Then the joint fitness function is applied to sort and choose members into the current population (line 15).

In the original framework, the reference points without population members will be deleted. Besides, an adjustable reference points strategy is utilized to update reference points with none associated members which could be deleted in the original framework (line 16). And finally, the solution set is stored as P which is the output at the last generation (line 19).

2.2 A Joint Fitness Function

This section shows how the joint fitness function is comprised of Scalarizing Evaluation, Reference Point Association Achievement and the number of current generations.

Besides, in the original selection operator after the non-dominated sorting, only the P_j could be considered to assist the following selection operators. The joint fitness function is applied to sort and choose the candidate solutions. The solutions with the better joint fitness value could be selected into the new population first.

Scalarizing Evaluation. As we know, one important characteristic of MaOPs is that the number of objectives is large. Thus, the selection pressure would become lower if only the non-dominated levels are regarded as the measurement of convergence. Therefore, the proposed algorithm applies a method to represent the convergence during the selection opera-tion. After normalization, the gap between candidate solutions and extreme points in each dimension is calculated to be the parameter for measure the convergence of the solution set.

As the good performance that it has, the achievement scalarizing function (ASF) is utilized for reference with the measurement of the convergence. Based on ASF, MaSHOA proposes the Scalarizing Evaluation (SE) to make sure the convergence. It calculates the maximum difference between the value on each dimension of objective vectors with the preference vector w_i and the best value on each dimension as shown in Eq. 1.

$$SE(x) = maxF\left(x, z_i^{min}\right) = max_{i=1}^{m}\left(w_i \cdot f_i(x) - z_i^{min}\right) \tag{1}$$

$$w_i = \frac{f_i(x)}{\sum_{j=1}^{m} f_j(x)} \tag{2}$$

Where m is the number of objectives, $f_i(x)$ is the value of objective vector on the ith dimension, z_i^{min} is the value of extreme point on the ith dimension, w_i is the preference vector which is calculated as Eq. 2. We can see that the smaller value of $SE(x)$ is, the closer objective vector is near to the best value, the better convergence it is.

Reference Point Association Achievement. The relationship between reference points and population solutions is shown as the Reference Point Association Achievement (RPAA). It contains two parts to represent the diversity of solutions. One is the distance between current solution and the reference vector formed by the reference point associated with it, and another is the number of solutions associated with the reference point of the current solutions P_j. Thus, the distance (Eq. 4) between the current candidate solution and reference vector r_j and the scalarized current P_j (Eq. 5) is represented as RPAA shown in Eq. 3.

$$RPAA(x_i) = distance(x_i, r_j) \cdot \frac{P_j}{\overline{P}} \tag{3}$$

$$distance(x_i, r_j) = |x_i| \times \sin(x_i, r_j) = |x_i| \times \frac{|x_i \times r_j|}{|x_i| \cdot |r_j|} \tag{4}$$

$$\overline{P} = \frac{\sum_{j=1}^{s} P_j}{s} \tag{5}$$

Where x_i represents the current candidate solution vector, r_j is the reference vector, P_j represents the number of solutions associated with the reference point of the current solution. We can see that the smaller the value of RPAA is, the better the diversity of solutions becomes.

Joint Fitness Function. It is believed that the feasible solution should focus on the convergence as much as possible in the early stage of the process, that is, making solutions forward better. And in the late stage of the process, the diversity of solution set is becoming more im-portant, that is, making distribution of the solution more spread and even. Therefore, the value of iterations is significant to the quality of solutions. The calculation con-siders the number of current generations as a variable. Combining SE and RPAA, the joint fitness function (JF) is presented as Eq. 6.

$$JF(x) = {}^{1}\!/_{g} \times SE(x) + g \times RPAA(x) \tag{6}$$

Where g is the number of current generations. We can see that the smaller g is, the more important the convergence is, and the larger g is, the more important the diversity is.

2.3 Many-Objective Optimization Squirrels Evolution

The population p evolves and mutates through the method of this section into population S. In this part, the joint fitness function is treated as the sorting reference. And Algorithm 2 shows the procedure of the many-objective optimization squirrel evolution.

According to the rank, the first n1($n1 = N/50$) squirrels are the best squirrels which are considered to be on the hickory trees. And the following n2 ($n2 = 3N/50$) squirrels are the second-best squirrels which are considered to be on the acorn nuts trees. And the last n3 ($n3 = N - n1 - n2$) squirrels are normal squirrels which are considered to be on the normal trees.

As the living habits of squirrels, when there are no natural enemies of squirrels, squirrels begin to migrate. This paper sets the probability P_e of natural enemies existing as 0.1. The probability of squirrel migration is based on P_e. Besides, whether squirrels migrate is decided with the random number. A random number is generated between 0 and 1 in every condition. And if this number is more than 0.1, squirrels do the migration. The distance constant SC of squirrel moving is set as [0.5, 1.11] due to experience, and moving distance of every time is decided randomly in this range.

Algorithm 2. Procedure for many-objective optimization squirrel evolution

Input:
Squirrels population
Output
New squirrel population
Begin
1. evaluate fitness value according to Section2.2;
2. sorting squirrels with the fitness value;
3. According to the rank, first n1 squirrels are the best squirrels, the following n2 squirrels are the second-best squirrels, and all the others are normal squirrels.;
4. Random number of the range of [0,1] as r1, r2, r3;
5. **While** r1 > P_e (for second-best squirrels)
6. **For** n = 1 **to** n2.;
7. squirrel migration according to Eq. 7;
8. **While** r2 > P_e (for normal squirrels)
9. For n = 1 to n4;
10. squirrel migration according to Eq. 8;
11. **While** r3 > P_e (for normal squirrels)
9. For n = 1 to n5;
10. squirrel migration according to Eq. 9;
11. Calculate distance penalization constant according to Eq. 10;
12. **While** (season != winter)
13. Normal squirrels levy flight according to Eq. 11;
End

Squirrels on acorn nuts trees are moving to one of the directions of hickory trees randomly according to Eq. 7.

$$ST'_{2nd} = ST_{2nd} + (ST_{best} - ST_{2nd}) \times SC \tag{7}$$

Where ST_{2nd} represents the current location of the second-best squirrel, ST'_{2nd} represents the new location of the moving second-best squirrel, $(ST_{best} - ST_{2nd})$ is the distance between the best squirrel and the current moving second-best squirrel.

Squirrels on the normal trees are moving to one of the directions of acorn nuts trees randomly according to Eq. 8. Some of normal squirrels have never been on the acorn nuts trees (the number of them is n4).

$$ST'_n = ST_n + (ST_{2nd} - ST_n) \times SC \tag{8}$$

Where ST_n represents the current location of the normal squirrel, ST'_n represents the new location of the moving normal squirrel, $(ST_{2nd} - ST_n)$ is the distance between the second-best squirrel and the current moving normal squirrel.

Besides, some of normal squirrels were on acorn nuts trees in the past generations (the number of them is n5 = n3 − n4). Thus, they are moving to one of the directions of hickory trees randomly according to Eq. 9.

$$ST'_n = ST_n + (ST_{best} - ST_n) \times SC \tag{9}$$

Where ST_n represents the current location of the normal squirrel, $ST_n^{'}$ represents the new location of the moving normal squirrel, $(ST_{best} - ST_n)$ is the distance between the second-best squirrel and the current moving normal squirrel.

Meanwhile, the winter detection is applied to prevent the algorithm from local convergence. The difference between the best squirrels of the current generation and the last generation is considered as the distance penalization constant. The minimum distance between best squirrels of the current and last generation is calculated to be the distance penalization constant (DPC) as Eq. 10. WDC is applied to evaluate the similarity of these two generations and judge whether the process is going into the local convergence.

$$DPC = min_{i,j=1}^{n1,n1'} \sqrt{\left(ST_i^g - ST_j^{g-1}\right)^2} \tag{10}$$

Where g is the current generation, n1 is the number of best squirrels in this generation, n1$'$ is the number of best squirrels in last generation, ST_i^g is the vector of ith best squirrel in this generation, and ST_j^{g-1} is the vector of jth best squirrel in last generation. So DPC can represent the minimum distance between the best squirrels from two generations.

And a threshold value is fixed to compare with DPC to judge whether it is the end of winter (see Eq. 11). If the DPC is less than the fixed threshold, it is considered as the situation of local convergence.

$$WD^{min} = \frac{10E^{-6}}{(365)^{\frac{g}{g_m/2.5}}} \tag{11}$$

Where g is the number of the current generation, g_m is the maximum number of generations. So WD^{min} is the threshold of winter detection.

In addition, the best and second-best squirrels of the current generation should be preserved, and the normal squirrels should mutate using Levy flight according to Eq. 12.

$$ST_n^{'} = ST_n + (ST_{max} - ST_{min})_2 \times Levy \tag{12}$$

$$Levy = \frac{0.01 \times r_a \times \sigma}{|r_b|^{\frac{1}{\beta}}} \tag{13}$$

$$\sigma = \left(\frac{\Gamma(1+\beta) \times \sin(\frac{\pi\beta}{2})}{\Gamma\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\left(\frac{\beta-1}{2}\right)}}\right)^{\frac{1}{\beta}} \tag{14}$$

$$\Gamma(x) = (x-1)! \tag{15}$$

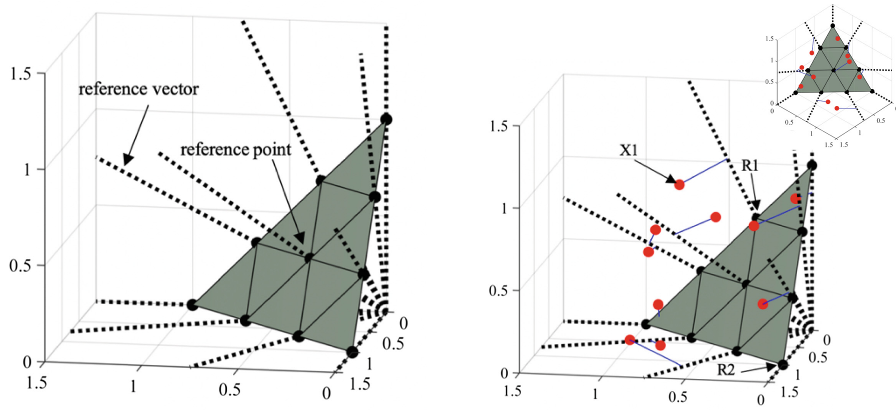
Where $ST_n^{'}$ is the new location of moving normal squirrels, ST_{min} and ST_{max} are the squirrel location of the best and worst fitness value respectively. And Levy is shown in Eq. 13.

After the winter detection, the new squirrel population S is generated completely which can be utilized to merge with the previous population.

2.4 Adjustable Reference Points Strategy

In the original reference points set method, the reference point associated without any solutions will be deleted. And in this section, an adjustable reference point strategy is proposed to update the reference point.

After the reference point associated without any solutions is detected, the solution with the farthest distance between others and itself is chosen to be the new reference point based on the feasible solution set as shown in Fig. 1. The original reference points and their corresponding reference vectors are shown in Fig. 1(a). As is shown in Fig. 1(b), there are no solutions associated with R2, so x1 with the maximum distance is chosen to adjust R2 to X1. X1 becomes the new reference point. This strategy could make sure the diversity while making solution set evolve to the advantageous direction, instead of the original even reference point set. Meanwhile, in every adjustment process, the one reference point cannot be associated with more than one new reference point. This means the solution associated with the reference point with the previous new reference point could not be the new reference point. This constrain can prevent the solving process from evolutionary convergence forward one certain direction. And After the previous X1 in Fig. 1(b) is becoming the new reference point, we are assuming that the new solution with the maximum distance from its reference vector should be selected to be the new reference. But the new solution is also associated with the R1 (same as Fig. 1(b)), so this solution could not be employed to be the new reference set. In this case, the other solution with the second farthest distance should be selected to be the new reference point.



a. Reference points and vectors settings   b. An adjustable reference point strategy

Fig. 1. Based on the reference points set, an adjustable reference point strategy is designed shown in this figure.

3 Simulative Results Analysis

To analyze the performance of the proposed algorithm, this section designs the comparative simulation experiment. Based on the previous work, we choose NSGA-III and MOEA/D to contrast the MaSHOA. And all these three EMaOAs are simulated at five classic many-objective optimization test problems (DTLZ 7, DTLZi, i = 1 – 4) [26] as shown in Table 1. And the selected algorithms are tested on the 3-objective to 15-objective DTLZ test problems.

Both NSGA-III and MOEA/D apply the genetic algorithm as the mutation strategy. Table 2 shows the parameters of mutation operators used in the NSGA-III and MOEA/D.

Table 1. Test problems

Name	Dimensions of solutions	Feature of PF
DTLZ1	4 + N	Linear multi-modal
DTLZ2	9 + N	Concave
DTLZ3	9 + N	Concave multi-modal
DTLZ4	9 + N	Concave non uniform
DTLZ7	19 + N	Mixed, Disconnected multi-modal

Table 2. Parameters of variation operators

Parameters	NSGA-III	MOEA/D
SBX probability [28] p_c	1	1
Polynomial mutating probability [29] p_m	1/n	1/n
Crossover distribution index η_c	30	20
Mutation distribution index η_m	20	20

The number of reference points depends on both the simple-lattice design factor [10] D and the number M of objectives. D represents the number of points which are distributed evenly on one borderline of the solution space. We can calculate the number R of reference points or directions according to Eq. 16.

$$R = \binom{D + M - 1}{D} \tag{16}$$

When $M \geq 8$, the two layers of reference points are employed to generate the reference points. And the reference points are divided into the inside layer and the boundary layer. Table 3 presents the number of objectives, divisions, reference points

Table 3. Number of reference points/directions and population sizes used in this experiment

No. of objectives (M)	No. of divisions (D)	No. of reference p/d (R)	MaSHOA popsize (N)	NSGA-III popsize (N')	MOEA/D popsize (N")
3	12	91	91	92	91
5	6	210	210	212	210
8	(3,2)	156	156	156	156
10	(3,2)	275	275	276	275
15	(2,1)	135	135	136	135

and population size of all the algorithms used in the experiments. To ensure the accuracy of the experiment results, these simulative experiments are carried out 30 times on each test problem respectively.

3.1 Analysis on IGD Metric

In general, the solution set of MaOPs is composed of many solutions, which leads to the difficulty of the evaluation of solutions of different algorithms. So, the evaluation method on EMaOAs is complicated to transfer the solution set to a mode easy to evaluate. The inverse generational distance (IGD) is widely applied to evaluate the convergence and the diversity of PF. The convergence describes the gap between the approximate PF and the ideal PF. And the diversity means that the solution set could represent the whole page of the ideal PF.

In this experiment, we take the IGD metric as the evaluation indicator. The IGD metric represents the average distance between each solution of the reference set and the solution closest to it of approximate PF. The IGD metric is defined as Eq. 17. Z_i is the ideal solution set, P is the approximate solution set obtained by all the EMaOAs, z_i and x_j is the solution of the ideal and approximate set respectively.

$$\mathrm{IGD}(\mathrm{P}, Z_i) = \frac{1}{|Z_i|}\sum\nolimits_{i=1}^{Z_i} \min_{\substack{P \\ j=1}} d\left(z_i, x_j\right) \tag{17}$$

We can see that the smaller the IGD value, the closer the approximate solution to the real PF, the better the performance of the algorithm. Meanwhile, the lower IGD value could represent that there exist solutions around each solution of the ideal PF, which is the meaning of the diversity.

Table 4 shows all the means and variances (shown in the first and second line) of IGD values of all the results on the five DTLZ test problems and the maximum generations of each situation of different objectives.

On the DTLZ1 problems, MaSHOA has the best means on the 3-, 8-, 10-, 15-objective problems, and the best variances on the 5-, 8-, 15- objective problems. And the other two algorithms do not have better performance on all DTLZ1 problems. A similar measurement is made for DTLZ2 problems, MaSHOA has the best results on the 8-, 10-,

Table 4. The means and variances of IGD obtained by simulation results on DLTZ

Problems	No. of objectives	MaxGen	NSGA-III	MOEA/D	MaSHOA
DLTZ1	3	30000	**5.742E−4** 6.847E−7	0.0408 0.00305	2.626E–4 **1.26E–11**
	5	30000	6.532E−4 1.124E−8	0.184 0.0356	**5.32E–4** **3.02E–11**
	8	50000	0.0447 2.910E−4	0.292 0.0641	**0.01765** **5.782E–5**
	10	50000	0.0199 **1.510E−4**	0.116 0.0072	**0.0196** 1.612E-4
	15	50000	0.0114 3.059E−5	0.0288 0.00125	**0.00105** **4.952E–9**
DLTZ2	3	30000	**5.877E−4** **4.50E−13**	6.507E-4 1.22E-11	6.735E-4 2.825E-11
	5	30000	**0.00170** **1.88E−12**	0.00238 8.95E-10	0.00175 1.453E-10
	8	50000	0.0102 6.606E−6	0.00924 3.933E−7	**0.00823** **1.85E-12**
	10	50000	0.00816 8.742E−7	0.00698 3.206E-7	**0.00561** **4.22E-11**
	15	50000	0.00421 2.047E−8	0.00363 6.538E-8	**0.00306** **2.44E-13**
DLTZ3	3	30000	0.00403 9.444E−5	0.187 0.0416	**6.765E-4** **7.27E-11**
	5	30000	0.00181 8.798E−9	0.421 0.209	**0.00172** **5.51E-11**
	8	50000	0.107 0.00233	0.597 0.210	**0.0410** **8.399E-4**
	10	50000	**0.0418** 0.00102	0.3178 0.147	0.0455 **1.28E-4**
	15	50000	0.01432 8.19E-5	0.0216 0.00233	**0.00307** **5.04E-11**
DLTZ4	3	30000	0.00522 1.024E-5	**0.00147** **1.355E-6**	0.00369 9.082E-6
	5	30000	**0.00170** **2.22E-12**	0.00367 2.653E-8	0.00176 1.954E-10
	8	50000	0.00851 4.021E-7	0.0109 3.39E-7	**0.00823** **2.947E-12**

(*continued*)

Table 4. (*continued*)

Problems	No. of objectives	MaxGen	NSGA-III	MOEA/D	MaSHOA
	10	50000	**0.00557** **5.19E-14**	0.00793 1.021E-7	0.00559 2.589E-11
	15	50000	0.00311 1.920E-9	0.00374 6.156E-9	**0.003062** **7.68E-14**
DLTZ7	3	30000	0.0322 3.720E-5	0.0473 1.539E-4	**0.0300** **3.02E-10**
	5	30000	0.104 9.885E-5	0.0870 7.121E-6	**0.0736** **3.59E-10**
	8	50000	0.265 2.410E-4	0.103 4.313E-5	**0.0961** **2.960E-8**
	10	50000	0.349 6.335E-4	0.124 6.965E-5	**0.110** **2.487E-8**
	15	50000	0.573 1.304E-4	0.216 3.599E-4	**0.146** **1.766E-7**

15-objective problems, while NSGA-III has good results on the 3-, 5- objective problems which have a little gap with MaSHOA. We can see that the proposed algorithm has better performance on a higher dimension of DTLZ2 problems. And for DTLZ3 problems, MaSHOA has the best results on the 3-, 5-, 8-, 10-, 15-objective problems except that NSGA-III has better means of the 10-objective problems. For problem DTLZ4, MaSHOA has the best performance on the 8-, 15-objective problems, and it still gets good results in other problems. MaSHOA has the best performance on the 3-, 5-, 8-, 10-, 15-objective DTLZ7 problems. From Table 4, MaSHOA has better stabilization and solving performance on these DTLZ problems than other algorithms.

Figure 2 shows the box plots of all the algorithms on 8-, 10- and 15-objective DTLZ problems. The box plots present minimum, maximum, median, first quartile and third quartile (sometimes outliers) of repetitious experiment results. The stability of results can be directly observed through the box plots.

For the 8-objective DTLZ problems (in Fig. 2a–e), MaSHOA has smaller boxes with lower values than the other two algorithms. MOEA/D has good results on the DTLZ2 and DTLZ7, and NSGA-III has good results on the DTLZ1 and DTLZ4, but the shape of MaSHOA is much flatter in these five figures and MaSHOA doesn't have outliers like other two algorithms. There is a similar survey made on the 10-objective DTLZ problems (in Fig. 2f–g) that MaSHOA has better performance on all the problems. And also, NSGA-III is good on the DTLZ1 and DTLZ4while MOEA/D is worse than others. For 15-objective DTLZ problems (in Fig. 2k–o), MaSHOA still has better results than others. And both NSGA-III and MOEA/D don't hold the second position on all the five problems.

It is shown that some of MaSHOA boxes are compressed into a line which means the proposed algorithm has great stability rather than others. And the boxes of MaSHOA are

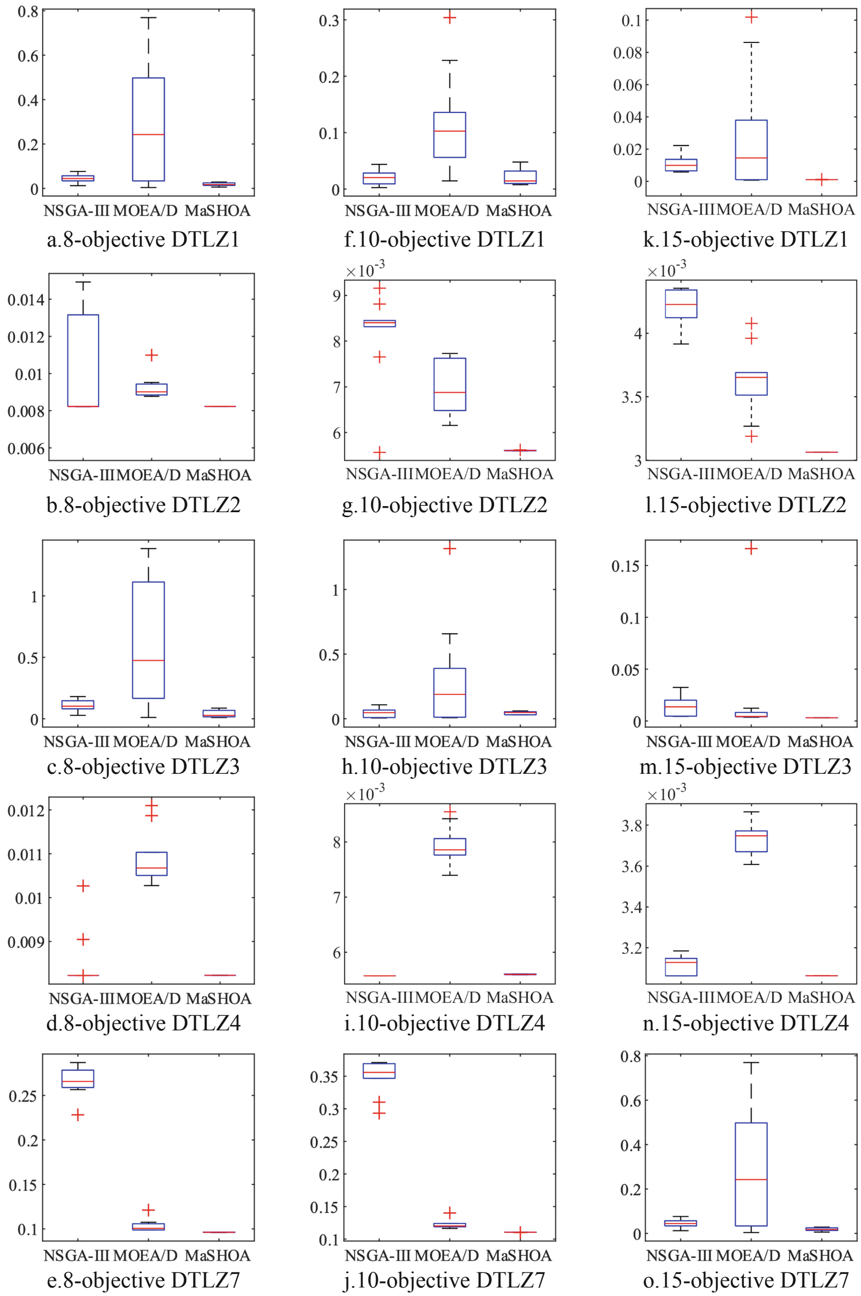


Fig. 2. Box plots of 8-, 10-, 15-objective DTLZ problems.

lower than others on the height in the figures which means the IGD values of MaSHOA are less than others. So MaSHOA has better convergence and diversity.

3.2 Analysis of Parallel Coordinate Plots

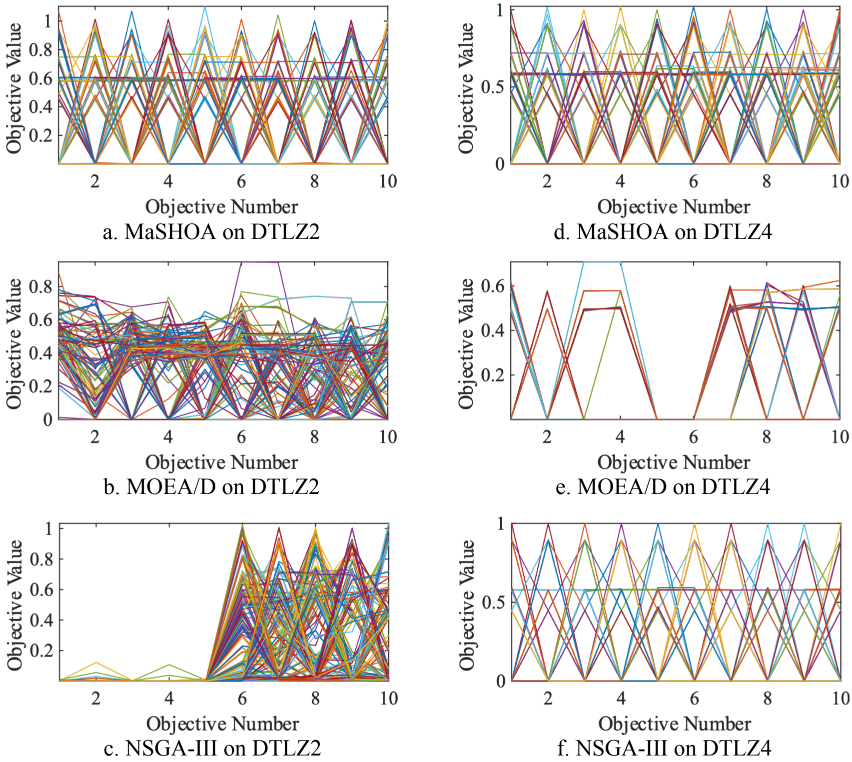


Fig. 3. Parallel coordinate plots of 10-objective problems

This paper utilizes parallel coordinate plots to analyze the fitting precision of obtained PF. The parallel coordinate plot is an effective solution for the difficulty of high-dimensional visualization (discussed in the introduction). The parallel coordinate plot shows all the coordinates of each dimension on the parallel axis and connects them with the polygonal lines.

Figure 3 shows six parallel coordinate plots of results on 10-objective DTLZ2 and DTLZ4 problems. For the problem DTLZ2 and DTLZ4, MaSHOA can show a more integrated fitting precision of PF than the other two algorithms. The results on DTLZ2 and DTLZ4 obtained by MOEA/D are not uniform, and the results on DTLZ2 obtained by NSGA-III are not uniform either. The results on DTLZ4 obtained by NSGA-III are uniform but not good as those of MaSHOA. A similar phenomenon is observed in Fig. 4 which presents six parallel coordinate plots of results on 15-objective DTLZ2 and DTLZ3 problems. The results of MaSHOA are better than others.

And it is shown that the performance of three algorithms is affected by increasing the number of objectives. And MOEA/D and NSGA-III could only present the part of ideal PF while the proposed algorithm could show integrally the PF relatively.

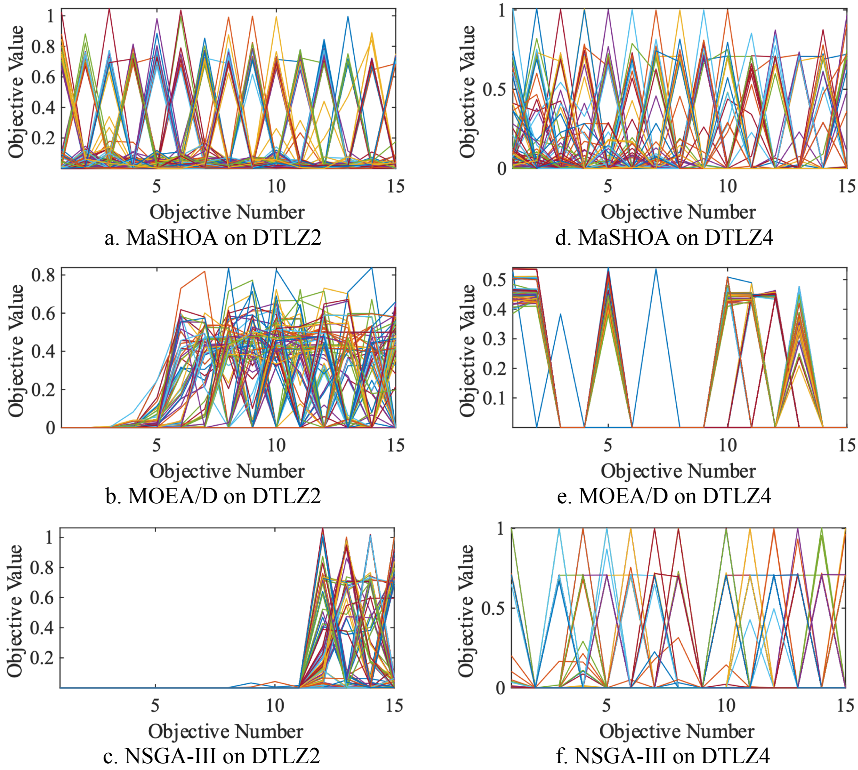


Fig. 4. Parallel coordinate plots of 15-objective problems

On the basis of the simulative experiment analysis, the proposed algorithm MaSHOA has good performance and stability on the DTLZ test set rather than MOEA/D and NSGA-III.

4 Conclusion

This paper presents a many-objective squirrel hybrid optimization algorithm (MaSHOA) which uses the framework of SSA and NSGA-III for reference and improves it on the evolution and selection operators.

A new joint fitness function combines the scalarizing evaluation which evaluates the convergence of the solution among the neighborhood and the reference point association achievement which shows the influence of the associated relationship between reference points and candidate solutions on the quality of solutions and considers the different preference of solving goals from early to late iterations. The new joint fitness function is applied to sort solutions in both evolution and selection operators. The better solutions are treated as the guide of others according to the sorting in the evolutionary process, and the candidate solutions are selected in terms of the sorting results. Besides, the distance penalization is proposed to prevent local convergence during the evolution.

An adjustable reference points strategy is designed to adjust the reference point set.

The simulative experiments of MaSHOA, NSGA-III and MOEA/D are implemented on 3-, 5-, 8-, 10-, 15-objective DTLZ test problems. The IGD metric, a widely applied EMaOA evaluation method, is utilized to evaluate results obtained by running all the algorithms 30 times repeatedly. Through the means, variances and box plots of IGD metrics obtained by all the algorithms are used to compare the stability and performance of algorithms. Moreover, results on 10-, 15-objective problems are visualized by parallel coordinate plots. Taken together, the proposed algorithm, MaSHOA, has good performance on different problems which means it is an effective many-objective optimization algorithm.

References

1. Kibria, M.G., Nguyen, K., Villardi, G.P., Zhao, O., Ishizu, K., Kojima, F.: Big data analytics, machine learning, and artificial intelligence in next-generation wireless networks. IEEE Access **6**, 32328–32338 (2018)
2. Wang, G.G., Cai, X., Cui, Z., Min, G., Chen, J.: High performance computing for cyber physical social systems by using evolutionary multi-objective optimization algorithm. IEEE Trans. Emerg. Top. Comput. **8**, 20–30 (2017)
3. Gong, D., Xu, B., Zhang, Y., Guo, Y., Yang, S.: A similarity-based cooperative co-evolutionary algorithm for dynamic interval multi-objective optimization problems. IEEE Trans. Evol. Comput. **24**, 142–156 (2019)
4. Fleck, M., Troya, J., Kessentini, M., Wimmer, M., Alkhazi, B.: Model transformation modularization as a many-objective optimization problem. IEEE Trans. Softw. Eng. **43**(11), 1009–1032 (2017)
5. Zouache, D., Arby, Y.O., Nouioua, F., et al.: Multi-objective chicken swarm optimization: a novel algorithm for solving multi-objective optimization problems. Comput. Ind. Eng. **129**, 377–391 (2019)
6. Jiang, S., Yang, S.: A strength Pareto evolutionary algorithm based on reference direction for multiobjective and many-objective optimization. IEEE Trans. Evol. Comput. **21**(3), 329–346 (2017)
7. Ishibuchi, H., Sato, H.: Evolutionary many-objective optimization. In: Proceedings of the Genetic and Evolutionary Computation Conference Companion, pp. 614–661, July 2019
8. Luo, J., Liu, Q., Yang, Y., Li, X., Chen, M.R., Cao, W.: An artificial bee colony algorithm for multi-objective optimisation. Appl. Soft Comput. **50**, 235–251 (2017)
9. Cho, J.H., Wang, Y., Chen, R., et al.: A survey on modeling and optimizing multi-objective systems. IEEE Commun. Surv. Tutor. **19**(3), 1867–1901 (2017)
10. Jain, H., Deb, K.: An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: handling constraints and extending to an adaptive approach. IEEE Trans. Evol. Comput. **18**(4), 602–622 (2013)
11. Gong, D., Sun, J., Miao, Z.: A set-based genetic algorithm for interval many-objective optimization problems. IEEE Trans. Evol. Comput. **22**(1), 47–60 (2016)
12. Yuan, Y., Xu, H., Wang, B., Yao, X.: A new dominance relation-based evolutionary algorithm for many-objective optimization. IEEE Trans. Evol. Comput. **20**(1), 16–37 (2015)
13. Zhao, H., Zhang, C., Zhang, B., Duan, P., Yang, Y.: Decomposition-based sub-problem optimal solution updating direction-guided evolutionary many-objective algorithm. Inf. Sci. **448**, 91–111 (2018)
14. Zhang, Q., Li, H.: MOEA/D: a multiobjective evolutionary algorithm based on decomposition. IEEE Trans. Evol. Comput. **11**(6), 712–731 (2007)

15. Cheng, R., Jin, Y., Olhofer, M., et al.: A reference vector guided evolutionary algorithm for many-objective optimization. IEEE Trans. Evol. Comput. **20**(5), 773–791 (2016)
16. Zhu, Y., Liang, J., Chen, J., Ming, Z.: An improved NSGA-III algorithm for feature selection used in intrusion detection. Knowl.-Based Syst. **116**, 74–85 (2017)
17. Shang, K., Ishibuchi, H.: A new hypervolume-based evolutionary algorithm for many-objective optimization. IEEE Trans. Evol. Comput. **24**, 839–852 (2020)
18. Du, Y., Xing, L., Zhang, J., et al.: MOEA based memetic algorithms for multi-objective satellite range scheduling problem. Swarm Evol. Comput. **50**, 100576 (2019)
19. Sun, Y., Yen, G.G., Yi, Z.: IGD indicator-based evolutionary algorithm for many-objective optimization problems. IEEE Trans. Evol. Comput. **23**(2), 173–187 (2018)
20. Liu, Z., Jiang, D., Zhang, C., et al.: A novel fireworks algorithm for the protein-ligand docking on the AutoDock. Mobile Netw. Appl. 1–12 (2019)
21. Mirjalili, S., Jangir, P., Saremi, S.: Multi-objective ant lion optimizer: a multi-objective optimization algorithm for solving engineering problems. Appl. Intell. **46**(1), 79–95 (2017)
22. Mirjalili, S.Z., Mirjalili, S., Saremi, S., et al.: Grasshopper optimization algorithm for multi-objective optimization problems. Appl. Intell. **48**(4), 805–820 (2018)
23. Kumar, M., Guria, C.: The elitist non-dominated sorting genetic algorithm with inheritance (i-NSGA-II) and its jumping gene adaptations for multi-objective optimization. Inf. Sci. **382**, 15–37 (2017)
24. Jain, M., Singh, V., Rani, A.: A novel nature-inspired algorithm for optimization: squirrel search algorithm. Swarm Evol. Comput. **44**, 148–175 (2019)
25. Liu, Z., Zhang, C., Zhao, Q., Zhang, B., Sun, W.: Comparative study of evolutionary algorithms for protein-ligand docking problem on the AutoDock. In: Song, H., Jiang, D. (eds.) SIMUtools. LNICSSITE, vol. 295, pp. 598–607. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-32216-8_58
26. Ishibuchi, H., Setoguchi, Y., Masuda, H., et al.: Performance of decomposition-based many-objective algorithms strongly depends on Pareto front shapes. IEEE Trans. Evol. Comput. **21**(2), 169–190 (2016)
27. Ruiz, A.B., Saborido, R., Luque, M.: A preference-based evolutionary algorithm for multi-objective optimization: the weighting achievement scalarizing function genetic algorithm. J. Global Optim. **62**(1), 101–129 (2014). https://doi.org/10.1007/s10898-014-0214-y
28. Deb, K., Agrawal, R.B.: Simulated binary crossover for continuous search space. Complex Syst. **9**(2), 115–148 (1995)
29. Deb, K.: Multi-objective Optimization Using Evolutionary Algorithms, vol. 16. Wiley, Hoboken (2001)