A Multi-objective Particle Swarm Algorithm Based on a Preference Strategy

Yi Wang(✉), KangShun Li, and Yong Fan

College of Computer Science, Guangdong University of Science and Technology,
Dongguan 523083, China
gust312@163.com

Abstract. To solve the problems of premature convergence and insufficient diversity in high-dimensional multi-objective optimization. A preferred area-based multi-objective particle swarm optimization (PAMOPSO) is proposed. The algorithm first determines the preferred regions through reference points, divides the target space into preferred regions and non-preferred regions, and uses different dominance rules in the different regions to improve the survival pressure of the algorithm in the high-dimensional space. Secondly, we design the selection mode of the global optimal position Gbest in two stages, comprehensively considering the convergence and distribution of the algorithm. Experimental results show that the proposed algorithm has good convergence and diversity compared with other improved multi-objective algorithms.

Keywords: High-dimensional multi-objective optimization · multi-objective particle swarm algorithm · preferred multi-objective algorithm

1 Introduction

In practical optimization problems, decision-makers often have to consider many conflicting goals, namely, multi-objective optimization problems (MOPs). Intelligence optimization algorithms prove to be effective for solving multi-objective optimization problems. Particle swarm optimization (PSO) is a well-known evolutionary algorithm to simulate the interaction during bird group predation. Due to its simple mechanism, few parameters, and fast convergence speed, PSO has been successfully applied to solve MOPs and expanded to a multi-objective particle swarm optimization algorithm. However, MOPSO can easily fall into local optimal. Besides, a single search mode of particle swarm quickly leads to poor convergence accuracy and diversity [1].

With the deepening of the research, the researchers explore the introduction of the particle swarm algorithm into the process of multi-object problem solving, combined with the rapid convergence characteristics of the use of the particle swarm algorithm and the excellent performance in the multi-object optimization problem. Domestic and foreign researchers mainly divided the multi-target particle swarm algorithms into the following groups according to the selection mechanism:

L. Pan et al. (Eds.): BIC-TA 2022, CCIS 1801, pp. 44–53, 2023.
https://doi.org/10.1007/978-981-99-1549-1_4

(1) **Multi-objective particle swarm algorithm dominated by Pareto.** Zhang et al. proposed the CMOPSO algorithm to change the particle speed through the update mechanism based on a learning strategy. The particles in the current population select the winning particle through pairwise competition. This particle guides another particle to perform speed updates, better balancing convergence and population diversity and reduces unnecessary memory overhead and computing complexity [2]. By introducing two-stage strategies, Hu et al. emphasized convergence and distribution in different stages. They maintained the variety of external archives by introducing parallel cell systems to support the selection of more diverse solutions [3]. Lin et al. facing the multi-target particle group algorithm under high target space selection pressure drop, put forward the NMPSO algorithm, and introduced a new fitness evaluation method to overcome the Pareto-dominated ranking difficulty or the limitation of decomposition method, combining the convergence distance and diversity distance to balance the algorithm convergence ability and population diversity ability, and adopted the new speed update way, not research provides another direction [4].

(2) **Multi-objective particle swarm algorithm based on decomposition.** This kind of method is a multi-objective optimization problem into a set of single-objective optimization problems, by solving each subproblem without using Pareto control. Coello et al. use the decomposition method using a global optimal position solution set to update example locations, simultaneously maintaining the diversity of the population [5]. Dai et al. proposed the MPSO/D algorithm to maintain diversity. The target space of the multi-objective optimization problem is divided through the direction vector and incorporates the idea of differential evolution [6].

(3) **Index-based multi-target particle swarm algorithm.** The algorithm is in the process of updating by evaluating the index to guide the search direction of the algorithm. Garcia et al., by introducing the HV index guide algorithm search and convergence, according to the optimal global location and individual historical optimal HV index, the external archive update according to the size of HV index, its low dimensional target space can be a good guide algorithm search, but in high target problem, HV solution is very difficult, the calculation complexity of the algorithm is further improved [7–9].

This study proposes a multi-objective particle swarm optimization algorithm based on a preference strategy to divide the target space into preferred and non-preferred regions and improves the archiving mode and the optimal global location selection mechanism, verifying the algorithm performance on different test sets.

2 Relevant Theoretical Basis

2.1 Particle Swarm Optimization

The particle swarm optimization algorithm is initialized into a group of random particles, in which the optimal solution is found through iteration. During each iteration, the particle updates its speed and position by tracking two extreme values (Pbest, Gbest). the updated formula is [8]:

$$v_t(t+1) = w \times v_t(t) + r_1 \times c_1 \times (pbest_i(t) - x_i(t)) + r_2 \times x_2 \times (gbest(t) - x_i(t)) \tag{1}$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{2}$$

where x$_\mathrm{i}$(t) = (xt1, xt2,…, xtn) is the position information of the i-th particle at the t iteration, n represents the dimension of the decision variable in the particle solution v$_\mathrm{i}$(t) = (vt1, vt2,…, vtn) is the velocity information of the i-th particle in the t-th iteration. Pbest represents the historical optimal position of the i-th particle, the local optimal position at the t generation, while Gbest represents the best location explored in the whole population.

c1 and c2 are the learning factors, c1 is the individual learning factor, c2 is the individual social learning factor, if c1 = 0 means the particle only group experience, its convergence is fast, but easy to fall into local optimal, if c$_2$ = 0, means the particle share no group information, a scale for M group run M are particles, the chance of solution is very small, so generally c$_1$ and c$_2$ is set to the same size. r$_1$ and r$_2$ are the random numbers between [0,1], w is the inertial weight factor when w is larger, the next generation speed is larger, so can expand the global search range, w small, the next generation speed will decrease, thus local search ability enhancement, so can choose larger w value, prevent the algorithm missing the possible optimal solution, into local optimal, and later choose smaller w value, improve the convergence speed of the algorithm.

2.2 Multi-objective Optimization Problem

Due to the complexity of practical problems, our solution is not limited to single-objective optimization but to two or more issues, calling these including two or more objective problems as multi-objective optimization problems. These goals to be optimized include the problems of maximization, minimizing, or both, and to facilitate the solution, we can transform the goal problems into the maximization or minimization solution. This paper discusses minimization. Then the mathematical representation of the minimized multi-objective optimization problem is as follows [10, 11]:

$$\min f(x) = (f_1(x), f_2(x), \ldots, f_m(x)) \tag{3}$$

$$subject \text{ to } \quad x \in S \subset R^n \tag{4}$$

where $x = (x_1, x_2, \ldots, x_\mathrm{n})$ is n dimension decision variables, S is the feasible domain of x, and m represents the number of objective functions. $f_i(x)$ the i-th objective function.

3 The Proposed Algorithm

3.1 Improve the Preference Areas

The MOPSO algorithm follows strict Pareto dominance rules when evaluating the merits of the solutions, but it works against the algorithm convergence in a high-dimensional space. Lopez et al. proposed a new preference dominance relationship, dividing the entire target space into two subspaces, one subspace of the individual is compared according to the Pareto dominance relationship, and the payoff scalar function reaches one subspace. In the proposed algorithm τ value is set in the algorithm initialization stage in advance for the value of the [0,1] interval. Still, for multi-target particle algorithm itself has "precocious" characteristics. If fixed the size of the preferred area is, it is easy to make the algorithm converge to the local optimal, and it may miss other better solutions and not be conducive to the diversity of the solution. Therefore, this paper proposes dynamic value adjustment to ensure that the algorithm can expand the search range in the first and mid stages, safeguard the diversity of solutions, and narrow the search range in the late stage so that the algorithm can converge effectively.

The value of τ decreases gradually with the number of iterations but then linear. If the decrease rate is fast before the algorithm has fully searched the feasible area, the τ value is defined by formula (5):

$$\tau = \tau_{\max} - (\tau_{\max} - \tau_{\min}) * (current_{iter}/\max_{iter})^{c} \tag{5}$$

$\tau_{\max}$ and $\tau_{\min}$ set the maximum and minimum values of the decision maker for the search stage, currentiter is the current number of iterations, maxiter is the maximum number of iterations, and c is the descent speed.

3.2 Gbest Selection Strategy

The Gbest selection strategy of the MOPSO algorithm depends on the density of the particles, but the algorithm only considers maintaining the uniformity of the solution distribution while ignoring strengthening the convergence ability of the algorithm. This paper proposes a two-stage Pbest selection strategy to consider both the algorithm's convergence ability and distribution ability when selecting the Pbest.

In the first stage, according to the position of the particle in the target space and the target space to calculate the corresponding similar distance (Similarity Distance, SD), according to the similar distance set selected from the particles far from the noninferior solution, keep the noninferior solution as the selection set of the second stage, SD solution method as shown in Eq. (6) and Eq. (7):

$$d(x_i, y_j) = \sqrt{\sum_{k=1}^{M} (f_k(x_i) - f_k(y_i))^2} \tag{6}$$

$$SD_i\{d(x_i, y_1), d(x_i, y_2), \ldots, d(x_i, y_t)\} \tag{7}$$

where x_i represents the i-th particle in the population, y_j represents the j-th non-inferior solution (j = 1, 2, …, t), t is the number of non-inferior solutions in the external archive Archive, $f_k(x_i)$ represents the function value of the i-th particle on the k-th target, M is the number of an objective function, SD_i is the collection of stored i-th particle and each particle in the external archive Archive in the target space.

Similar distances represent the distance of the particles to be updated from the particles in an external archive. In the first stage, the algorithm can mainly search for more solutions. The x_i particles should tend to select particles more prominent than the average similarity distance in the external archive, as shown in Eqs. (8) and (9).

$$AVG_{SD_i} = \frac{\sum_{j=1}^{t} SD_{i,j}}{t} \tag{8}$$

$$\sum_{j=1}^{t} SD_{i,j} = d(x_i, y_1) + d(x_i, y_2) + \ldots + d(x_i, y_t) \tag{9}$$

Through the selection of the first stage, the set of non-inferior solutions to be processed in the second stage, such as the formula (10).

$$S = \{y | y \in Archive \ \& \ d(x_i, y) > AVG_{SD_i}\} \tag{10}$$

In the first stage, it considers the diversity of the algorithm solution. To strengthen the convergence ability of the algorithm, the second stage focuses on realizing the uniform distribution of the solution set and selects the individuals with excellent distribution performance in the S set. The entire calculation step of the improved algorithm is as follows: The whole calculation step of the improved algorithm is shown as Algorithm 1:

Algorithm 1 Improves the multi-objective particle swarm algorithm

Step 1: Set the relevant parameters, and initialize the population speed, position, Pbest and Gbest;

Step 2: Calculate the value of the particles on each target function in the population and select the preferred region according to formula (5).

Step 3: evaluate the advantages and disadvantages of particles according to the control rules, and save the non-inferior solution set to the external archive.

Step 4: Select Gbest for each particle according to formula (6-10).

Step 5: Update the position and speed of each particle.

Step 6: execute the variation, evaluate the quality of the particles, update the particle Pbest.

Step 7: Update the external archive and remove particles if the external archive exceeds the storage capacity.

Step 8: Determine whether the maximum number of iterations or the stop condition of the algorithm is reached. If not reached, return to step 4, if reached, the cycle ends, output the solution set in the external archive, and the end of the algorithm.

4 Experimental Results and Analysis

4.1 Evaluation Indicators

Different from single objective optimization, which can compare the size of the target function value to evaluate the algorithm performance, multi-objective optimization needs some performance indicators to evaluate the algorithm effect, mainly from two aspects to evaluate the algorithm, on the one hand, evaluate the algorithm solution set and the real Pareto edge, on the other hand, evaluate the distribution of the algorithm solution set.

Therefore, the performance indicators of multi-objective optimization can be divided into three categories:

(1) The first type of indicator considers the distance between the solution set obtained by the algorithm and the real Pareto solution set, such as the GD index.
(2) The second type of indicator considers the distribution of the solution set obtained by the algorithm in the target space, such as the Spacing index.
(3) The third category is the comprehensive index, which considers the algorithm solution's convergence while considering the algorithm's distribution, such as IGD and HV index.

This paper uses GD, IGD, HV, and distance performance indicators from the reference point to evaluate the algorithm [12–14].

(1) The GD calculation formula is shown in formula (11):

$$GD(P, P^*) = \frac{\sum_{\mu \in P} d(\mu, P^*)}{|P|} \tag{11}$$

where P^* is the set of sampling points uniformly distributed on the optimal edge of real Pareto, P is the Pareto non-dominant solution set solved by the algorithm, d(u, P) represents the minimum value of the distance between all solutions from u to P. from formula (10) can be found that the smaller the GD value, the more the algorithm converges.

(2) The IGD calculation formula is shown in formula (12):

$$IGD(P^*, P) = \frac{\sum_{v \in P} d(v, P^*)}{|P^*|} \tag{12}$$

(3) The HV calculation formula is shown in formula (13):

$$HV = \bigcup_{i} vol_i| \in PF \tag{13}$$

4.2 Algorithm Parameter Setting

To verify the performance of the proposed algorithm PAMOPSO, several commonly used benchmark functions (DTLZ2, DTLZ4, DTLZ6–7) will be used to test the proposed algorithm, and the test results will be compared with the more popular multi-objective optimization algorithms, including MMOPSO, CMOPSO, g-NSGA. In the experiment, the number of decision variables of all test functions is M+9, and M is the number of objective functions.

Comparing the population size of the algorithm, the external archive capacity, and the maximum number of iterations are shown in Table 1. The other parameters of each algorithm are set from the relevant references. The algorithm was run 50 times independently in each case test function, and the performance index was averaged 50 times.

Table 1. Initial conditions

Algorithms	Population size	Archive	Iterations
PAMOPSO	4 Objectives:150	Reference population size	800
MMOPSO	6 Objectives:150	——	800
CMOPSO	8 Objectives:200	——	800
g-NSGAII	10 Objectives:20	——	800

4.3 Experimental Results

Table 2, Table 3, and Table 4, respectively, show the measured values of the three test functions on 4, 6, 8, and 10 targets, which include the average values of GD, IGD, and HV. The best results are marked using the bold font method. Also, the number of PAMOPSO better, worse, and equal to other algorithm test examples is given in the last row of the table.

Table 2. The mean GD values obtained on the DTLZ problem

Test functions	M	PAMOPSO	MMOPSO	CMOPSO	g-NSGAII
DTLZ2	4	3.7738E−02	5.3122E−02+	1.5387E−01+	3.7821E−02=
	6	5.3536E−02	1.3321E−01+	1.5442E−01+	5.3235E−02=
	8	5.9284E−02	1.4647E−01+	1.5598E−01+	5.9343E−02=
	10	6.6561E−02	1.6158E−01+	1.5996E−01+	6.6670E−02=
DTLZ4	4	2.6924E−02	5.1425E−02+	1.0257E−01+	1.4394E−01+
	6	5.0021E−02	1.4825E−01+	1.3869E−01+	1.2707E−01+
	8	5.5324E−02	1.3526E−01+	1.4015E−01+	1.8321E−01+
	10	7.6291E−02	1.4061E−01+	1.2667E−01+	1.2787E−01+
DTLZ6	4	**6.1534E−02**	5.6114E−01+	6.9726E−01+	6.3809E−01+
	6	**6.5981E−02**	7.0722E−01+	6.7222E−01+	6.2658E−01+
	8	**5.4311E−02**	6.2307E−01+	6.8057E−01+	6.2985E−01+
	10	**5.2320E−02**	6.5274E−01+	6.5195E−01+	6.3943E−01+
±/=			12/0/0	12/0/0	8/0/4

Tables 2, 3 and 4 show that the proposed PAMOPSO algorithm performs well on both the DTLZ4 and DTLZ6 test functions. The best convergence is achieved with the other three algorithms, and the performance is comparable to the g-NSGA algorithm on DTLZ2. Overall, this also proves that the PAMOPSO algorithm can be guaranteed

Table 3. The mean IGD values obtained on the DTLZ function

Test functions	M	PAMOPSO	MMOPSO	CMOPSO	g-NSGAII
DTLZ2	4	1.5221E−01	1.3221E−01=	1.0954E−01−	5.7264E−01+
	6	5.3263E−01	3.83485E−01−	1.5987E+00+	1.5961E+00+
	8	7.3661E−01	7.6232E−01=	2.3621E+00+	2.0513E+00+
	10	8.2562E−01	1.0299E+00+	2.4361E+00+	1.1236E+00+
DTLZ4	4	2.2331E−01	1.2103E−01−	1.3954E−01−	1.8324E−01=
	6	4.7395E−01	3.4351E−01−	6.2101E−01+	9.3651E−01+
	8	7.1324E−01	8.2571E−01+	1.3215E+00+	1.9876E+00+
	10	8.0281E−01	1.5631E+00+	1.3947E+00+	1.4764E+00+
DTLZ6	4	1.8540E−02	4.2164E−02+	2.5468E−01+	2.8674E+00+
	6	3.2440E−02	2.6345E−01+	6.2422E+00+	8.5981E+00+
	8	3.1230E−02	4.8451E−01+	9.4966E+00+	8.6568E+00+
	10	3.2931E−02	4.6954E−01+	9.5032E+00+	8.9614E+00+
±/=			7/3/2	10/2/0	11/0/1

Table 4. Average HV values obtained on the DTLZ problem

Test functions	M	PAMOPSO	MMOPSO	CMOPSO	g−NSGAII
DTLZ2	4	5.7643E−01	6.3689E−01−	6.5911E−01−	1.4214E−01+
	6	3.1364E−01	5.3652E−01−	4.5623E−01−	3.2462E−04+
	8	2.9981E−01	2.8511E−01+	0.0000E+00+	3.2141E−05+
	10	1.4695E−01	1.1364E−01+	0.0000E+00+	7.3124E−03+
DTLZ4	4	6.5684E−01	6.6542E−01+	6.2965E−01+	4.6912E−01+
	6	4.6381E−01	6.0369E−01−	1.7361E−01+	3.5243E−02+
	8	4.8954E−02	2.2456E−01−	2.5124E−03+	8.9631E−03+
	10	2.9961E−02	4.1245E−02−	1.3512E−04+	9.7512E−04+
DTLZ6	4	1.5631E−01	1.5212E−01=	1.2365E−01+	0.0000E+00+
	6	1.1023E−01	9.7864E−02+	0.0000E+00+	0.0000E+00+
	8	9.8964E−02	8.8624E−02+	0.0000E+00+	0.0000E+00+
	10	9.7521E−02	8.9657E−02+	0.0000E+00+	0.0000E+00+
±/=			6/5/1	10/2/0	12/0/0

faster in the high-dimensional target space by introducing the reference points and the preferred regions, adopting the new dominance rules according to the preferred regions, and adopting the new Gbest strategy.

5 Conclusion

The particles of traditional multi-target particle clusters all use a single search pattern, resulting in premature population convergence and poor diversity while causing poor particle generalization ability.

In this study, we set up reference points and divided the target space into two subspaces according to the preferred area. In different spaces, the solution evaluation method improved the survival pressure of the population and searched the algorithm for the region of interest of the decision maker. Secondly, the two-stage selection Gbest method is proposed. In the first stage, the algorithm can search for a wider range of solutions, and the second stage ensures that the understanding can be distributed more evenly. Experimental results show that the improved algorithm has excellent performance.

Acknowledgment. This work is supported by the Guangdong Youth Characteristic Innovation Project (2021KQNCX120), the Natural Science Foundation of Guangdong Province of China (2020A1515010784), the Natural Science Project of Guangdong University of Science and Technology (GKY-2021KYYBK-20), the General Project of Science and Technology of Dongguan Social Development (20231800910352), the Natural Science Project of Guangdong University of Science and Technology (XJ2022003501), the University Distinguishing Innovation Project of Guangdong Provincial Department of Education (2021KTSCX149), and the Key Project of Science and Technology of Dongguan Social Development (20211800905512).

References

1. Habib, M., Aljarah, I., Faris, H., Mirjalili, S.: Multi-objective particle swarm optimization: theory, literature review, and application in feature selection for medical diagnosis. In: Mirjalili, S., Faris, H., Aljarah, I. (eds.) Evolutionary Machine Learning Techniques. AIS, pp. 175–201. Springer, Singapore (2020). https://doi.org/10.1007/978-981-32-9990-0_9
2. Zhang, X., Zheng, X., Cheng, R., Qiu, J., Jin, Y.: A competitive mechanism based multi-objective particle swarm optimizer with fast convergence. Inf. Sci. **427**, 63–76 (2018)
3. Hu, W., Yen, G.G., Luo, G.: Many-objective particle swarm optimization using two-stage strategy and parallel cell coordinate system. IEEE Trans. Cybern. **47**(6), 1446–1459 (2016)
4. Lin, Q., et al.: Particle swarm optimization with a balanceable fitness estimation for many-objective optimization problems. IEEE Trans. Evol. Comput. **22**(1), 32–46 (2016)
5. Coello, C.A.C., Pulido, G.T., Lechuga, M.S.: Handling multiple objectives with particle swarm optimization. IEEE Trans. Evol. Comput. **8**(3), 256–279 (2004)
6. Dai, C., Wang, Y., Ye, M.: A new multi-objective particle swarm optimization algorithm based on decomposition. Inf. Sci. **325**, 541–557 (2015)
7. García, I.C., Coello, C.A.C., Arias-Montano, A.: Mopsohv: A new hypervolume-based multi-objective particle swarm optimizer. In: 2014 IEEE Congress on Evolutionary Computation (CEC), pp. 266–273. IEEE (2014)
8. Poli, R., Kennedy, J., Blackwell, T.: Particle swarm optimization: an overview. Swarm Intell. **1**(1), 33–57 (2007). https://doi.org/10.1007/s11721-007-0002-0
9. Deb, K.: Multi-Objective Optimization, pp. 403–449. Search Methodologies. Springer, Boston, MA (2014)
10. Tian, Y., et al.: Evolutionary large-scale multi-objective optimization: a survey. ACM Comput. Surv. (CSUR) **54**(8), 1–34 (2021)

11. Zhang, Y., Wang, S., Phillips, P.: Detection of Alzheimer's disease and mild cognitive impairment based on structural volumetric MR images using 3D-DWT and WTA-KSVM trained by PSOTVAC. Biomed. Signal Process. Control **21**(8), 58–73 (2015)
12. Chen, L., Wang, H., Ma, W.: Two-Stage multi-tasking transform framework for large-scale many-objective optimization problems. Complex Intell. Syst. **7**(3), 1499–1513 (2021)
13. Tian, Y., Zheng, X., Zhang, X.: Efficient large-scale multi-objective optimization based on a competitive swarm optimizer. IEEE Trans. Cybern. **50**, 3696–3708 (2019)
14. Das, S., Mullick, S., Suganthan, P.: Recent advances in differential evolution – an updated survey. Swarm Evol. Comput. **27**, 1–30 (2016)