A Novel Multi-objective Evolutionary Algorithm

Bojin Zheng¹ and Ting Hu²

¹ College of Computer Science, South-Central University For Nationalities,
Wuhan, 430074, China

² Department of Computer Science, Memorial University of Newfoundland, St.
John's, NL, A1B 3X5, Canada
zhengbojin@gmail.com, tingh@cs.mun.ca

Abstract. Evolutionary Algorithms are recognized to be efficient to deal with Multi-objective Optimization Problems (MOPs) which are difficult to be solved with traditional methods. Here a new Multi-objective Optimization Evolutionary Algorithm named DGPS which is compound with Geometrical Pareto Selection Method (GPS), Weighted Sum Method (WSM) and Dynamical Evolutionary Algorithm (DEA) is proposed. Some famous benchmark functions are carried out to test this algorithm's performance and the numerical experiments show that this algorithm runs much faster than SPEA2, NSGAII, HPMOEA and can obtain finer approximate Pareto fronts which include thousands of well-distributed points.

Keywords: Multi-Objective Optimization, Evolutionary Algorithm, Geometrical Pareto Selection, Weighted Sum Method.

1 Introduction

Multi-objective optimization problems (MOPs) are very common in economics, and engineering etc., but they are a class of very difficult problems. In 1984, David Schaffer proposed Vector Evaluated Genetic Algorithm (VEGA) [1,2] to deal with MOPs. Since then many Multi-objective Optimization Evolutionary Algorithms (MOEAs) are proposed, most of them base on Pareto techniques. Since the works of E. Zitzler et al. [3,4,5] in 1999, the importance of elitism strategy in multi-objective search was recognized and supported experimentally.

In general, the current popular MOEAs employ explicit or implicit "archive" [6,7] which is designed to store the non-dominated solutions to implement the elitism. So the schema of current MOEAs could be depicted as follow [8]:

$$\text{MOEA} = \text{Archive} + \text{Generator}$$

The equation means that if we want to design an efficient MOEA, we should pay attention to two aspects: firstly, design a more efficient algorithm to deal with archive, i.e., elitist space, secondly, design a more efficient algorithm in the production of new individuals.

As to the generator, hundreds of Evolutionary Algorithms would be meaningful to us. Among them, Dynamical Evolutionary Algorithm (DEA) [9] would be worth of researching. Zou [10] proposed High Performance MOEA (HPMOEA)

[11] which combines DEA with Ranking method and Niching method to deal with MOPs and reported good results.

As to the archiving algorithm, except the schema of “Ranking-alike operator + Niching-alike operator”, there exists another schema, which is called “the sampling schema”. Geometrical Pareto Selection (GPS) would be such a typical algorithm[12,13].

In this paper, we propose a novel MOEA which combines Geometrical Pareto Selection Method (GPS), Weighted Sum Method (WSM) and Dynamical Evolutionary Algorithm (DEA) together. The experimental results show that this algorithm runs much faster than SPEA2, NSGAII[14], HPMOEA and can obtain finer approximate Pareto fronts which include thousands of well-distributed points.

2 Background

In this section, we present three major components of DGPS to help understand our work. For the detail information please refer to [10],[13].

2.1 Geometrical Pareto Selection

For convenience of discussion, we only discuss how to deal with the two-objective optimization problems.

The method can be depicted as follows:

1. Use some techniques to estimate Pareto front, $F_1 \subset (L_1, U_1), F_2 \subset (L_2, U_2)$. Here F is the objective functions.
2. Choose a constant integer κ , here κ is an angle constant which splits the Pareto front equally. Create an array: Array. And the array, i.e., archive, is used to store the current best solutions (but not non-dominated solutions).
3. Choose a point $A(x, y)$ which is far away from Pareto front.
4. When a new individual $N(x_1, y_1)$ will be inserted into archive, use such rules:
 - Compute its slope α with point $A(x, y)$.

$$\alpha = \frac{y_1 - y}{x_1 - x}$$

- Compute its distance dis with point $A(x, y)$.

$$dis = \sqrt{(y_1 - y)^2 + (x_1 - x)^2}$$

- Compute the location pos in archive.

$$pos = \left\lceil \frac{\alpha - \gamma}{\kappa} \right\rceil, \text{ here } \gamma = \frac{L_2 - y}{L_1 - x}$$

- Modify the archive. If $dis > Array[pos].dis$, then $Array[pos] := N(x_1, y_1)$.

For the above method, it's easy to generalize this method from 2-objective problems to N-objective problems. Namely, we can get the angles respectively by using the projective method. We should get N-1 angles totally and then find the very point to compare with the current point.

2.2 Dynamical Evolutionary Algorithm

DEA firstly is designed as a single objective optimizer. In DEA the individuals is considered as particles and the population as a dynamical system or particle system. So every particle is assigned a momentum and an activity, these two quantities are incorporated to control selections and to drive particles moving and searching all the time and everywhere. In DEA, the iterative step t in DEA like the generation in the traditional GA is called time t .

The momentum P of particle x_i at time t is defined as

$$P(t, x_i) = f(t, x_i) - f(t - 1, x_i)$$

Here f is the value of objective function.

The activity of $a(t, x_i)$ particle x_i is defined as the count that x_i is selected as the parent individual at time t .

We can generally take a weight co-efficient $\lambda \in (0, 1)$ to indicate that which one of the two terms is more significant than the other in selection. Namely

$$slct(t, x_i) = \lambda \sum_{k=0}^t |p(k, x_i)| + (1 - \lambda)a(k, x_i)$$

Based on $slct(t, x_i)$, not fitness, the individuals would be selected to evolve.

2.3 Weighted Sum Method

When searching in the solution space of MOPs, the individuals have to evolving toward multiple directions in common MOEAs. Actually, Evolutionary Algorithms are more efficient while evolving toward only one direction. So the Weighted Sum Method is employed to seek the extreme points, that is, the whole population is divided into several subpopulations, every subpopulation would evolving by a weight setting. Considering that trade-off solutions also are very important, an additional subpopulation is suggested to optimize the mean of all objectives.

While the objective functions are multi-modal, the method should cooperate with other operators to make the subpopulations evolve toward the right extreme points.

3 Introduction to New Algorithm

In this algorithm, GPS is an archiving algorithm, DEA is used as single-objective optimizer, WSM would guide the evolution of population. As to trade-off solutions, they will be generated by the optimizer during the optimization.

The algorithm can be depicted Figure 1:

```

Program DGPS;
1 Initialize Populations and archive
2 While not satisfy the stopping criteria do
3   For i=1 to SubPopNumber
4     Use DEA to optimize the corresponding objective
5     Try to insert every offspring into archive by GPS method
6     If successful in insertion or the offspring is better than its parent
7       then replace its parent
8   End For
9 End While
10 Using Cutoff strategy to clear all dominated solutions
11 Output all non-dominated solutions

```

Fig. 1. Pseudocode of DGPS

Note:

1) The definition of better

The better function is used to compare two individuals, we note the prototype as `bool Better(Indi1, Indi2)`. When solving the single objective problems, DEA uses a simple strategy to compare two individuals. The individual whose fitness is smaller is the better (for minimization). But when solving the multi-objective problems, we modify the strategy as follows:

*If the Indi1's corresponding objective value is smaller
then return true*

*Else if the Indi1's corresponding objective value equals to Indi2's
then if the average of all objectives of Indi1 is smaller
then return true
else return false*

2) Cutoff Strategy

In GPS, some dominated solutions are not eliminated until the iteration stops. So after the iteration, these dominated solutions should be removed by this strategy.

4 Experimental Results and Analysis

We test this algorithm with some famous test-bed functions, the results are very satisfactory. Because HPMOEA employs DEA, but with different archiving algorithm and search strategy, we compare our results with it.

For a fair comparison, we set the same generation number and size of population (the same evaluation times) to both the algorithms.

Problem 1: KUR

$$\begin{aligned}
 \min f_1(X) &= \sum_{i=1}^{n-1} \left(-10 \exp(-0.2 \sqrt{x_i^2 + x_{i+1}^2}) \right) \\
 \min f_2(X) &= \sum_{i=1}^n (|x_i|^{0.8} + 5 \sin x_i^3) \\
 x_i &\in [-5, 5], i = 1, \dots, 3
 \end{aligned} \tag{1}$$

Fig. 2. APF of KUR by DGPS

Fig. 3. APF of KUR by HPMOEA

The Pareto front of KUR is non-convex and disconnected. Because of the discrete nature of the Pareto-optimal regions, optimization algorithms may have difficulties in finding Pareto-optimal solutions in all regions. But both the proposed MOEA and HPMOEA obtain very good results, as shown in Figure 2 and 3.

Problem 2: VNT

$$\begin{aligned}
 \text{Min } f_1(X) &= 0.5(x_1^2 + x_2^2) + \sin(x_1^2 + x_2^2) \\
 \text{Min } f_2(X) &= \frac{(3x_1 - 2x_2 + 4)^2}{8} + \frac{(x_1 - x_2 + 1)^2}{27} + 15 \\
 \text{Min } f_3(X) &= \frac{1}{(x_1^2 + x_2^2 + 1)} - 1.1 \exp(-x_1^2 - x_2^2) \\
 \text{s.t. : } & -3 \leq x_1, x_2 \leq 3
 \end{aligned} \tag{2}$$

This is a difficult problem with high objective dimensions. But this algorithm generated a very fine pictures with obvious features, see Figure 4. From Figure 5, HPMOEA can not obtain the whole Pareto front, and its Pareto front does not include enough points.

Fig. 4. POF of VNT by DGPS

Fig. 5. POF of VNT by HPMOEA

From the pictures, we can see that this algorithm can obtain very fine curves in a single run. Obviously, this will contribute great convenience to decision-maker for the final decision and decrease greatly the decision risk of lack of sampling points. It is remarkable that the sampling points distribute very equally.

In this algorithm, multi-subpopulation strategy make the population evolve towards multiple directions, so it is easier to obtain the extreme points. Though this algorithm get more sampling points, the consumed time still is less than some other MOEAs. We compare this algorithm with HPMOEA and other two famous MOEAs - SPEA2 and NSGA-II which base on PISA[15].

All the algorithms are carried out on a machine with one Intel PIV 2.4G CPU. In the experiments, every algorithm runs 30 times, the sizes of populations are set to 100, the numbers of generation of DGPS and HPMOEA are set to 2000 and the number of SPEA2 and NSGA-II are set to 200. The results are listed in Table 1.

Table 1. Comparison of Consumed Time

Problem	Algorithm	Avg. Num. of points	Average time
ZDT3	DGPS	1420.0	0.837s
	HPMOEA	100	21.840s
	SPEA2	100	73.006s
	NSGAI	100	72.646s
SRN	DGPS	2026.1	0.282s
	HPMOEA	100	29.815s
KUR	DGPS	1381.4	1.440s
	HPMOEA	100	19.761s

From Table 1 we can see that this algorithm runs much faster than HPMOEA and can get more sampling points, so we can conclude that this algorithm is better than HPMOEA. Literature[10] points out that HPMOEA have a better convergent rate than SPEA2, NSGAI, so we can conclude that this algorithm can run faster than SPEA2,NSGAI and get finer approximate Pareto Front.

5 Conclusions and Future Work

In this paper, we proposed a new MOEA. The experimental results show that this algorithm can obtain finer approximate Pareto front with less time. In contrast with the some previous MOEAs, which may spend hours to perform one run for only hundreds of non-dominated solutions, this algorithm can obtain tens of thousands of solutions in a single run under a bearable time limitation.

In this paper, The theoretical analyses such as convergence property, convergence ratio are not introduced here, we will leave it as the future work.

Acknowledgement. The authors gratefully acknowledge the financial support of the National Natural Science Foundation of China under Grant No.60473014 and No.60603008.

References

1. Schaffer, J.: Some Experiments in Machine Learning Using Vector Evaluated Genetic Algorithms. PhD thesis, Vanderbilt University (1984)
2. Schaffer, J.: Multiple objective optimization with vector evaluated genetic algorithms. In: Proceedings of the First International Conference on Genetic Algorithms. (1985) 93–100
3. Zitzler, E., Thiele, L.: Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *Evolutionary Computation, IEEE Transactions on* **3**(4) (1999) 257–271
4. Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M., da Fonseca, V.G.: Performance assessment of multiobjective optimizers: an analysis and review. *Evolutionary Computation, IEEE Transactions on* **7**(2) (2003) 117–132

5. Laumanns, M., Zitzler, E., Thiele, L.: On the effects of archiving, elitism, and density based selection in evolutionary multi-objective optimization. *Evolutionary Multi-Criterion Optimization, Proceedings* **1993** (2001) 181–196
6. Knowles, J., Corne, D.: Properties of an adaptive archiving algorithm for storing nondominated vectors. *Evolutionary Computation, IEEE Transactions on* **7**(2) (2003) 100–116
7. Knowles, J.D., Corne, D.W., Fleischer, M.: Bounded archiving using the lebesgue measure. In: *Evolutionary Computation, 2003. CEC '03. The 2003 Congress on. Volume 4.* (2003) 2490–2497
8. Corne, D., Knowles, J.: Some multiobjective optimizers are better than others. In: *Evolutionary Computation, 2003. CEC '03. The 2003 Congress on. Volume 4.* (2003) 2506–2512
9. Li, Y., Zou, X., Kang, L., Michalewicz, Z.: A new dynamical evolutionary algorithm based on statistical mechanics. *Journal of Computer Science and Technology* **18**(3) (2003) 361–368
10. Zou, X.: *Research On Theory of Dynamical Evolutionary Algorithm and their Applications*. PhD thesis, Wuhan University (2003)
11. Zou, X.F., Kang, L.S.: Fast annealing genetic algorithm for multi-objective optimization problems. *International Journal of Computer Mathematics* **82**(8) (2005) 931–940
12. Zheng, B., Li, Y., Peng, S.: GPS: A geometric comparison-based pareto selection method. In Kang, L., Cai, Z., Yan, X., eds.: *Progress in Intelligence Computation and Applications, International Symposium on Intelligent Computation and its Application, ISICA 2005. Volume 1., Wuhan, China* (2005) 558– 562
13. Zheng, B.: *Researches on Evolutionary Optimization*. PhD thesis, Wuhan University (2006)
14. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: Nsga-ii. *Evolutionary Computation, IEEE Transactions on* **6**(2) (2002) 182–197
15. Bleuler, S., Laumanns, M., Thiele, L., Zitzler, E.: Pisa - a platform and programming language independent interface for search algorithms. In Fonseca, C., ed.: *Evolutionary Multi-Criterion Optimization(EMO 2003), LNCS 2632.* (2003) 494–508