New Model for Multi-Objective Evolutionary Algorithms

Bojin Zheng\textsuperscript{1} and Yuanxiang Li\textsuperscript{2}

\textsuperscript{1} College of Computer Science, South-central University for Nationalities, Wuhan, 430074, China
\textsuperscript{2} State Key Lab. of Software Engineering, Wuhan University, Wuhan, 430072, China
zhengbojin@gmail.com, yxli@whu.edu.cn

Abstract. Multi-Objective Evolutionary Algorithms (MOEAs) have been proved efficient to deal with Multi-objective Optimization Problems (MOPs). Until now tens of MOEAs have been proposed. The unified model would provide a more systematic approach to build new MOEAs. Here a new model is proposed which includes two sub-models based on two classes of different schemas of MOEAs. According to the new model, some representatives algorithms are decomposed and some interesting issues are discussed.

Key words: Multi-objective Optimization, Framework, Evolutionary Algorithm, Unified Model

1 Introduction

Evolutionary Algorithms are an randomized searching approach based on Darwin’s evolutionary theory. They play an important role in many fields such as optimization, control, game strategies, machine learning, and engineering design etc.

In 1984, David Schaffer introduced Vector Evaluated Genetic Algorithm (VEGA)\cite{1,2} to solve Multi-objective Optimization Problems (MOPs). Henceforth, the research on Multi-Objective Evolutionary Algorithms (MOEAs) attracted more and more researchers. Up to now tens of MOEAs have been proposed.

To guide the efforts on MOEAs, some researchers tried to build unified models for popular MOEAs. For examples, Macro Laumanns et al.\cite{3} proposed a unified model for the Pareto-based and elitist MOEAs in 2000. This model can describe most popular famous MOEAs, such as Non-dominated Sorting Genetic Algorithm II (NSGA-II)\cite{4}, Strength Pareto Evolutionary Algorithm and its improvement (SPEA/SPEA2)\cite{5}, Pareto Archived Evolution Strategy (PAES)\cite{6} and so on. \cite{7} expressed the schema of MOEAs which employ archive with such a formula as follows:

\[
\text{MOEA} = \text{Archive} + \text{Generator}
\]

But this formula is quite simple. Recently, more and more MOEAs cannot be accurately described by these models, such as Adaptive Grid Algorithm
In this paper, we propose a new model to describe the advanced MOEAs. In section 2, the model is introduced. And then SPEA[5], AGA[8] and GPS [10, 11] are decomposed in section 3 according to this model. Some interesting issues are discussed in section 4. In section 5, some conclusions are made.

2 Introduction to the New Model

The first MOEA – VEGA – is a non-Pareto algorithm. Subsequently, Goldberg D. E. [13] proposed to use Pareto dominance to compute the fitness of the individuals based on ‘Ranking’ method. Subsequent experiments prove that Pareto dominance based MOEAs are more efficient than non-Pareto MOEAs. Since the work of Zitzler et al.[5], the ‘elitism’ of MOEAs has been recognized: Elitism of MOEAs is especially beneficial in deal with MOPs and the use of elitism can speed up the convergence to the Pareto front. the Pareto based MOEAs with elitism is more efficient than the MOEAs without elitism. To implement the elitism, many MOEAs use a secondary ‘elitist’ population, i.e., the archive, to store the elite individuals. According to MOEAs’ formula, the pseudocode of common elitist MOEAs with archive can be depicted as Figure 1:

1 initialize the population and archive
2 evaluate the population
3 while the termination criterions have not been reached do
4 generate a solution by the generator
5 evaluate the new solution
6 try to update the archive
7 according to the feedback of archive, try to update the population
8 end while

Fig. 1. Pseudocode for Generic MOEAs with Archive

The pseudocode seemly does not mention the generation gap methods. Actually, the generation gap methods can be decomposed into this model, if we see the generation number and the replaced parent individuals as the additional parameters. Moreover, though Single-Objective Evolutionary Algorithms(SOEAs) and MOEAs are very similar, there still are three major differences:

1. Different to single-objective optimization, the generator of MOEAs may crossover some individuals in population with the individuals in population or archive
2. the fitness assignment is more complicated, because it is relative to two operators: fitness evaluation for the archive and fitness evaluation for the population
3. As to the elitism, the SOEAs just keeps only one fittest individual. But in MOEAs, the elitism, commonly, the strategy to update the archive is quite complicated.

In general, the new model can be depicted as Figure 2.

Fig. 2. The Model of Elitism MOEAs with Archive

In Figure 2, updating the archive would retrieve information from the archive, so the link is not drawn in this framework. Moreover, the generator is similar to the generating process of SOEA. Actually, except the selection operators, they both are same. It can be depicted as Figure 3.

Fig. 3. The Framework for the Generator

Secondly, the strategy of updating the archive is different to SOEA and very complicated.

Very many MOEAs employ the ranking-alike operators and the niching-alike operators. In such a schema, ranking-alike operators are firstly employed to clim-
inate the dominated solutions and secondly niching-alike operators are employed to eliminate the crowded solutions. But unfortunately, this kind of MOEAs are not convergent because of fitness deterioration. The schema of ranking-alike and niching-alike MOEAs (RN MOEA) could be depicted as Figure 4.

![Fig. 4. The Ranking-alike and Niching-alike Schema](image)

Actually, except this schema, there exists another schema. We call it 'the sampling schema'. In this schema, the feasible solution space (includes Pareto optimal front) is divided into grids in advance, when new solution is generated and evaluated, the algorithm firstly computes its coordinate in the grids and compare it with the individual(s) in the right coordinate. Whether updating the archive with the new solution or not just depends on the comparison. Obviously, this kind of methods are very different to ranking-alike and niching-alike methods, they use 'local dominance' instead of 'global dominance', and therefore hold lower time complexity. Some of them do not eliminate the dominated solutions from the archive in the main loop of algorithm, so additional operation (eliminating operator) should be employed after the main loop to cut the dominated solutions off from the archive. But if the archive should only store nondominated solutions, the eliminating operator should be integrated into the main loop. As to the diversity of Pareto optimal front, it depends only on the generator, because the span of cells in the grids has been predefined, may adaptively. The schema of sampling MOEAs (SA MOEA) can be depicted as Figure 5.

3 Decomposing Existing MOEAs

In this section, we will use SPEA, AGA and GPS to show how to decompose the existing algorithms.
3.1 SPEA

As to SPEA, it is a classical RN_MOEA. SPEA’s mating selection process is a typical selection operator which uses archive and population. Clustering method actually is an operator to keep diversity, i.e., niching-alike operator. Moreover, the truncate and update function can be seen as the strategies to update the archive and population.

3.2 AGA

Actually, AGA\cite{8} is a typical archiving algorithm. It must combine a generator to become MOEAs. AGA employ a nondominated vectors archive, so it employs $\text{Is\_Dominated}$ function and $\text{Dominates}$ function to eliminate the dominated solutions. Moreover, AGA uses adaptive grids, that is, the boundaries should be extended or reduced. However, in spite of these details, AGA samples the feasible solution space in essence. $\text{Reduce\_Crowding}$ function, $\text{Steady\_State}$ function and $\text{Fill}$ function actually perform the comparison and replacement.

3.3 GPS

GPS is also an archiving algorithm. It employs the $\text{Location}$ operator to retrieve the right solutions which will be used to compare with the new solution, employs $\text{Comparison}$ operator to perform the comparison, and at last employs the $\text{Steady\_State}$ operator to update the archive with the new solution if the new solution is better than the original solution in the archive. Moreover, the archive of GPS may store the dominated solution, only when the main loop ends, an additional operation is used to eliminate the dominated solutions.

4 Some Important Issues

According to the model, there are two kinds of archiving algorithms. Based on different schema, MOEAs would behave differently, and they would have different properties.
4.1 Performance Measure

The convergence property is very important to MOEAs. It would theoretically determine the approximation degree to the true Pareto front. But good diversity would be very helpful of the decision-maker. That is, MOEAs had better converge with diversity. So the performance measure should take both the convergence and diversity into considerations.

AGA has been proved convergent with well-distributed solutions under certain strict conditions. GPS also converges to Ray-Pareto optimal front. In contrast to SA_MOEAs, the RN_MOEAs do not converge. Furthermore, the SA_MOEAs could be improved to converge to true Pareto front under certain conditions.

The complexity of MOEAs would be another aspect of performance measure. If two multi-objective approaches have different archivers, their average performance may differ\cite{7}. Because of ranking method, the time complexity of RN_MOEAs would be greater than or equal to $O(MN)$ where computing one new individual, here $M$ is the number of objectives, $N$ is the size of population(or archive). As to AGA and GPS, the time complexity is $O(M)$. Niching-alike operators often hold a space complexity of $O(N^2)$. But AGA’s is $O(N^M)$, GPS’s is $O(N)$ at the best situation, $O(N^{M-1})$ at the worst situation. Furthermore, we can reduce GPS’s space complexity to $O(N)$ by using binary tree with an additional average time complexity of $O(N \log_2 N)$.

4.2 Cooperation between Generator and Archive

In this model, the generator should cooperate with archive to control the evolving directions. Therefore, considering to deal with difficult objective functions, 'local search' may be used to exploit. As to the archive of SA_MOEAs, 'local dominance' is useful to reduce the time complexity. 'local search' and 'local dominance' are different concepts.

The evolving directions of the population are multi-objective. In one hand, the selection pressure should make the individuals evolving toward the true Pareto front, i.e., depth-first search. In the other hand, the selection pressure should make the individuals spread over the whole Pareto front, i.e., width-first search. How to deal with the conflict between depth-first search and width-first search is still lack of delicate research. As to SA_MOEAs, because of local dominance, the feedback of archive just provide information for depth-first search, less for width-first search.

4.3 Taxonomy

Based the proposed model, we suggest that MOEAs could be categorized into four classes:

1. Non-Pareto MOEAs
   The representative MOEA is VEGA\cite{12}. This algorithm employs multiple sub-populations to optimize every single objective separately. This algorithm
often converge to special points which are not Pareto Optimal points, moreover, the diversity is not taken into consideration.

2. Pareto MOEAs (without Elitism)
The representative MOEAs include Multi-Objective Genetic Algorithm (MOGA) [15], Niched Pareto Genetic Algorithm (NPGA) [16,17] and Non-dominated Sorting Genetic Algorithm (NSGA) [18]. These algorithms employ some strategies to maintain the diversity, but approximation is not good enough.

3. Pareto MOEAs with Elitism
The representative MOEAs include NSGA-II [4], Strength Pareto Evolutionary Algorithm and its improvement (SPEA/SPEA2) [5]. These algorithms employ elitism strategy to maintain good approximation. But these algorithms are not convergent.

4. Convergent MOEAs
Actually, the archiving algorithms determine the convergence property of MOEAs. The representative algorithms include Adaptive Grid Algorithm (AGA) [8], GPS [10,11]. These algorithms should converge/pseudoconverge under certain conditions.

5 Conclusions and Future Work
The proposed model is intended to understand the state-of-the-art MOEAs and provides a more systematic approach to design more efficient and more customized MOEAs for researchers and possible users.

Our model implies that the ranking-alike and niching-alike schema is very different from the sampling schema, though they both may use archive to store elitist solutions.

In contrast to the previous unified models, the new model can describe the state-of-the-art MOEAs more accurately and be more atomic. So it is more convenient to use this model for the analysis of the algorithms.

This model provides us many cues to improve the MOEAs, such as the relationship between ‘local search’ and ‘local dominance’, the relationship between evaluation operation and re-evaluation operator and the relationship between depth-first search and width-search etc. The future work would try to discover more principles and develop new operators based on this model.

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. Laumanns, M., Zitzler, E., Thiele, L.: A unified model for multi-objective evolutionary algorithms with elitism. In: Congress on Evolutionary Computation (CEC 2000), IEEE Press (2000) 46–53
4. 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
5. 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
6. Knowles, J.D., Corne, D.: Approximating the non-dominated front using the pareto archived evolution strategy. Evolutionary Computation 8(2) (2000) 149–172
7. 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
8. Knowles, J., Corne, D.: Properties of an adaptive archiving algorithm for storing non-dominated vectors. Evolutionary Computation, IEEE Transactions on 7(2) (2003) 100–116
9. Haiming, L., Yen, G.G.: Rank-density-based multiobjective genetic algorithm and benchmark test function study. Evolutionary Computation, IEEE Transactions on 7(4) (2003) 325–343
10. 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
11. Zheng, B.: Researches on Evolutionary Optimization. PhD thesis, Wuhan University (2006)
12. Bui, L.T., Deb, K., Abbass, H.A., Essam, D.: Dual guidance in evolutionary multi-objective optimization by localization. In: The 6th International Conference on Simulated Evolution and Learning, Volume 4247 of Lecture Notes in Computer Science., Hefei, China, Springer Verlag, Heidelberg, D-69121, Germany (2006) 384–391
13. David, E.G.: Genetic Algorithms in Search, Optimization and Machine Learning, Reading, Massachusetts (1989)
14. Hanne, T.: On the convergence of multiobjective evolutionary algorithms. European Journal of Operational Research 117 (1999) 553–564
15. Tadahiko, M., Hisao, I.: Moga: Multi-objective genetic algorithms. In: Proceedings of the 2nd IEEE International Conference on Evolutionary Computing, Perth, Australia (1995) 289–294
16. Jeffrey, H., Nicholas, N., David, E.G.: A niched pareto genetic algorithm for multi-objective optimization. In: Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence. Volume 1., Piscataway, New Jersey (1994) 82–87
17. Igor, E.G., Sushil, J.L., Roberto, C.M.: Parallel implementation of niched pareto genetic algorithm code for x-ray plasma spectroscopy. In: Late Breaking Papers at the 2000 Genetic and Evolutionary Computation Conference, Las Vegas, Nevada (2000) 222–227
18. Srinivas, N., Kalyanmoy, D.: Multiobjective optimization using nondominated sorting in genetic algorithms. Evolutionary Computation 2(3) (1994) 221–248