Objective Reduction based on the Least Square Method for Large-dimensional Multi-objective Optimization Problem

Cong Zhou, Jinhua Zheng, Ke Li and Hui Lv
Institute of Information Engineering, Xiangtan University, Hunan, China 411105
E-mail:cleverchou23@yahoo.com.cn; Corresponding author: jhzheng@xtu.edu.cn

Abstract

In the real-world applications, many multi-objective optimization involve a large number of objective, however, existing evolutionary multi-objective optimization algorithms are applied only to a few number of objective. Because of inconvenience in handling large number of objective, researchers start to deal with how to reduce the redundant objectives. In this paper, we firstly introduce some existing algorithms on transforming high-dimensional to low-dimensional, and then propose a new algorithm, namely large dimensionality reduction based on the least square method. This method fits every objective function to a line, and compares the slope differences between each two lines, finally makes certain which one is redundancy and further reduces this one. This experiment shows, on one hand, there are some redundant objective functions in certain large dimensionality multi-objective optimization problems, and the objective space of non-redundant objective function is accordant with the low-dimensional true Pareto front. On other hand, the experiment result with other similar algorithm shows our algorithm is competitive and the efficacy of the procedure is demonstrated.

1. INTRODUCTION

Nowadays, Multi-objective Evolutionary Algorithm (MOEA) has shown an acceptable performance in many real-world problems with their origins in engineering, scientific and industrial areas [1]. However, most of the publications consider problems with two or three objectives, further, it is debatable, if it is worth utilizing MOEA methods to solve a large number of conflicting objectives (such as 10 or more objectives) for finding a representative set of Pareto-optimal solutions, for various practical reasons. First, visualization of a large dimensional front is certainly difficult. Second, an exponentially large number of solutions would be necessary to represent a large dimensional front, thereby making the solution procedures computationally expensive. Generally speaking, the more objectives are, the higher computationally time. Third, it would certainly be tedious for the decision makers to analyze so many solutions, to finally be able to choose a particular region, for picking up solution. These are some of the reasons why MOEA applications have also been confined to a limited number of objectives.

Currently, there are mainly two approaches to deal with many objectives: (1) to propose relaxed forms of Pareto optimality [2]; (2) to reduce the number of objectives of the problem to ease the decision making or the search processes [3, 4, 5, 6, 7, 8, 9].

In this paper, we propose an algorithm to reduce the number of objectives of a given problems by identifying the non-conflicting objectives. The algorithm is based on least square method, where every objective is fitted to a line, then gets the slope, and reduce the redundant objectives. A comparative study shows that the algorithm achieves competitive results with respect to three algorithms algorithm recently proposed [3, 6, 9]. Besides, the proposed algorithm has good candidates to incorporate into a multi-objective optimization algorithm for reducing the non-conflicting objectives.

The remainder of this paper is organized as follows. Section 2 presents three algorithms similar to our approach. In section 3 we describe in detail our algorithm. The validation of these algorithms is presented in Section 4. Finally, in Section 5 we draw some conclusions about the proposed algorithms, as well as some possible paths for future research.

2. RELATED WORK

Deb and Saxena [3, 4, 5] propose a method for reducing the number of objectives based on principal component analysis. The main assumption is that if two objectives are negatively correlated, these objectives are in conflict with each other. The authors analyze the eigenvectors of the correlation matrix in turn. To aggregate more objectives to the set of essential objective, the remainder of the eigenvectors is analyzed until the cumulative contribution of the eigenvalues exceeds a threshold cut (TC).

Brockhoff and Zitzler [6, 7, 8] define two kinds of objective reduction problem and two corresponding algorithms to solve them. If the dominance relation among the vectors doesn't change when an objective is discarded, then that objective is not considered as

978-0-7695-3736-8/09 $25.00 © 2009 IEEE
DOI 10.1109/ICNC.2009.40

redundant. We can find this subset which discard some objective, and replace the original objective set.

Jaimes and Coello [9] bring another method to reduce the objective using feature selection technique. The central part of the algorithm divided into three steps: first divide the objective set into homogeneous neighborhoods around each objective with the size q; then select the most compact neighborhood; finally maintain the certain of the most compact and discard its q neighbors. As Brockhoff and Zitzler's, they also divide the process into two parts.

3. PROPOSED OBJECTIVE REDUCTION ALGORITHM

In this paper, we propose a method to identify the most non-conflicting objective in order to reduce the number of objectives of an optimization problem, where we don't divide the process of objective reduction into two separate algorithms as literature [6, 7, 8, 9]. For the two kinds of objectives reduction problems, using other strategy in this paper, we obtain the needful results. The central part of the method is line fitting distribution trends of every objective by least square method, then analyze the relation between the each two objective slopes, and decide which objective is redundant.

Here, we introduce some symbol. The initial data set is represented in the form of a matrix $\mathrm{Y}=(\mathrm{Y}_1, \mathrm{Y}_2, \ldots \mathrm{Y}_M)^T$, where Y_i is the i-th objective, and it is also the vector, the size is the number of individuals in the population. M is objective dimension. We consider K_i as i-th objective slope and it is obtained as $\mathrm{K}_i = \dfrac{n\sum_{j=1}^{n} x_j y_j - \sum_{j=1}^{n} x_j \sum_{j=1}^{n} y_j}{n\sum_{j=1}^{n} x_j^2 - (\sum_{j=1}^{n} x_j)^2}$ where n is the population size and x is the same vector, namely $(1/n, 2/n, \ldots 1)$, and y is also vector, and it is the value of the i-th objective.

Before using least square method, there are some pretreatments.

(1) The given data must be in the standardized form, which means that the center of the whole data set is zero. This can be achieved by subtracting the mean and dividing standard deviation for each Y_i. After standardizing, the component of Y_i become concentration; Furthermore, if Y_i and Y_j is complete-linear relation, for example $\mathrm{Y}_i = k\mathrm{Y}_j + b$. After this process, we get the same slope, and obviously, the objective Y_i and Y_j are redundant, and their slope difference is always 0.

(2) Sort all of the components of Y_i. For example, there are two vectors, $\mathrm{Y}_i = (1, 2, \ldots, \mathrm{M})$ and $\mathrm{Y}_j = (\mathrm{M}, \mathrm{M}-1, \ldots 1)$. The sorting operator will change the vector Y_i into $\mathrm{Y}_i' = (\mathrm{M}, \mathrm{M}\text{-}1, \ldots 1)$, and their slopes values will be consistent.

Hereto, we can use the least square method to get every objective slope without any scruple. The followed is how to determine the redundant objective.

(3) The objective corresponding to the minimum mean is redundant. Figure 1 shows 4 objective and their slopes. First, we calculate respectively each objective difference values with others. For all of the objectives, we get the value as Table 1.

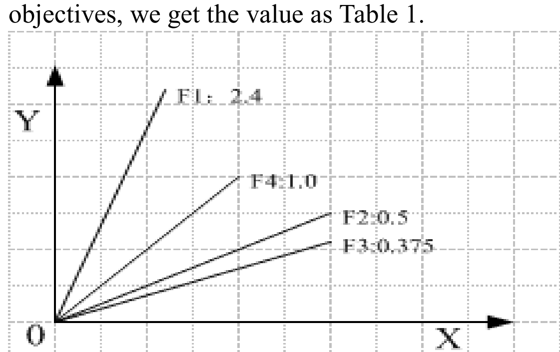

Figure 1: 4 objective and their slopes

Find the minimum value from Table 1, and it is 0.125 between F2 and F3. We determine preliminarily the redundant objective is one of the two objectives F2 and F3. But which objective can be reduced?

For objective F2, we can calculate the average value: $\frac{1.9+0.125+0.5}{3} = 0.842$; the same as above, for F3: $\frac{2.205+0.125+0.625}{3} = 0.985$. We decide the objective with minimum mean is redundancy, in this example, 0.985>0.842, so F2 is redundant objective.

Table 1: 4 objective and the valves of |Fi – Fj|

Fi / Fj	F1	F2	F3	F4
F1	0	1.9	2.025	1.4
F2	1.9	0	0.125	0.5
F3	2.025	0.125	0	0.625
F4	1.4	0.5	0.625	0

(4) Reduce only one objective after these operation processes. That is to say, if we consider F2 is redundancy, omit the objective and then we will initialize the new population with the objectives, F1, F3 and F4 (F2 is not included), and reduce other objectives with the same analysis until all remainder objectives are necessary. Using this strategy, we need not divide the algorithm into two parts as the paper [6, 7, 8, 9].

We will get the essential objectives set from M to $M'(M' \geq 2)$ and their data.

(5) Redundant degree σ. As the paper [6, 7, 8], our method have similar threshold cut σ. In Table 1, if we use σ (σ>0.125), we will think all of the objectives are necessary. In other words, if objective Fi and Fj are redundant, the different value must be less than σ, namely $|\text{slope(Fi)-slope(Fj)}|<\sigma$.

We now ready to present the overall the Least Square Method–NSGAII [12] procedure.

Step1: Set an iteration counter t=0 and σ; and initial set of objectives $R_t = \{1, 1, \ldots 1\}$.

Step2: Initialize a random population for all objectives in the set R_t, run an MOEA (NSGAII), and obtain a population P_t.

Step3: perform a Least Square Method analysis on P_t using R_t to yield a new reduced objectives R_{t+1}. Steps of the method are as follows:

1. Perform the process (1)-(5), which introduced above, and obtain the redundant objective index i, or i is null.
2. if i is not null then $R_t[i]=0$.

Step4: If $R_t = R_{t+1}$ stop and declare the obtained front. Else set t=t+1 and go to **Step 2.**

Thus starting with all M objectives, the above procedure iteratively finds a reduced set of objectives, by analyzing the obtained non-dominated solutions by an MOEA procedure. Furthermore, after analyzing procedure, only one objective may be reduced. Then we reinitialize non-redundant objectives set with all $R_t[i]=1$, where R_t is the vector, and if $R_t[i]=1$, the i-th objective is not considered redundant, otherwise it is redundant .When no further objective reduction is possible, the procedure stops and declares the final set of objectives from M to M' (M'<M) and corresponding non-dominated solution.

4. COMPARISON STUDY

To evaluate the effectiveness of the proposed algorithm (my algorithm, MINE), we compare its results against these obtained by the greedy algorithm proposed by Brockhoff and Zitzler to solve the σ-MOSS (with σ is alterable, BZ algorithm) and Jaimes and Coello's Feature Selection Technique algorithm (algorithm 2, JC algorithm). In this experiment, we employed a variation of the well-known DTLZ5, and DTLZ7 problems defined in [10]. In NSGAII, all parameters as following: generation is 400, population size is 200, and crossover probability and mutation probability are 0.9 and 0.1 respectively.

To evaluation the convergence the obtained Pareto front by using an objective subset we used the inverted generational distance (IGD), which is a variation of a metric proposed in [9]. It is defined by IGD= $\sqrt{\sum_{i=1}^{n} d_i^2} \Big/ n$, where n=$|PF_{true}|$ and d_i is the Euclidian distance between each vector of PF_{true} and the nearest member of PF_{known}. In addition, this metric measures the spread of PF_{known} onto PF_{true}. That is, a non-dominated optimal set, will be penalize in the value of this metric even though its vectors belong to PF_{true}. Lower values are preferred for this metric [13].

Table 2 shows the results for the DTLZ5 (3, M)[11] problem with variable objectives, that is to say, we will get 3 most necessary objectives from the total objectives set, M. Moreover, we regulate the σ (σ is about 2.6) in BZ algorithm so that the number of the non-redundant objective is also 3.

Table2: IGD values with M objective for DTLZ5

	BZ		JC		MINE	
M	Best	Avg	Best	Avg	Best	Avg
5	0.359	0.401	0.379	0.421	0.355	0.401
10	0.216	0.341	0.448	0.619	0.204	0.287
15	0.343	0.367	0.400	0.431	0.411	0.500
20	0.358	0.431	0.313	0.437	0.324	0.519
25	0.455	0.577	0.383	0.588	0375	0.548
30	0.444	0.521	0.466	0.527	0.377	0.486

In the DTLZ5 instance using 6 different objective numbers, the our algorithm has a best performance in Best IGD value aspects, in 4 of the 6 cases, and also in Avg value, in 4 of 6 cases. The other two algorithms are all square. From the Figure 2, the BZ algorithm is worse than others. Because our algorithm reduces only one redundant objective each time, it will run (M-3) times for the DTLZ5 (3, M) problems and running time is not in the ascendant.

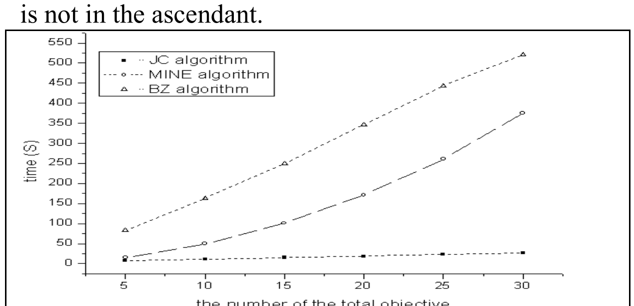

Figure 2: Running times of three algorithms for DTLZ5

Table 3 will shows the essential objectives identified in some case, and we will compare directly the reduced set of objectives.

Table3: Essential objectives

Problem	Reduced set of objective	
	JC and BZ	MINE
DTLZ5(2,5)	1,5	4,5
DTLZ5(3,5)	1,2,5	3,4,5
DTLZ5(2,10)	1,10	9,10
DTLZ5(3,10)	1,9,10	8,9,10

The following is another experiment, DTLZ7. We compare with the result as DTLZ5 problem, and the parameter σ is about 0.95 in BZ algorithm.

JC algorithm has the best performance (IGD value) in the DTLZ7 problem, in 3 of 6 cases. However, our proposed method has the similar results with the BZ algorithm, 2 of 6 and 1 of 6 respectively; they have similar IGD value and the essential objective set.

Table 4: IGD values with M objective for DTLZ7

	BZ		JC		MINE	
M	Best	Avg	Best	Avg	Best	Avg
5	0.344	0.44	0.322	0.552	0.369	0.49
10	14.8	15.7	15.9	16.0	14.2	15.5
15	40.2	42.7	41.5	42.6	41.8	42.7
20	52.9	54.3	22.2	48.2	51.9	54.8
25	68.2	71.1	30.2	42.8	69.7	71.6
30	80.3	82.5	42.1	63.4	35.6	60.4

In the file of running time, the JC algorithm is the best, then our way, the last BZ algorithm. We can distinguish the result from the Figure 3 easily.

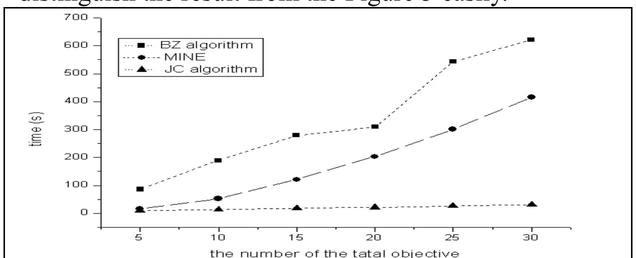

Figure 3: Running times of three algorithms for DTLZ7

There is disputed case. As to the objective functions themselves, some problems have non-redundant objectives [11], for example DTLZ2 and JC algorithm do not consider about this exception. After reducing the "redundant" objectives, some essential objectives centralize at a very small value. In fact, it is unreasonable. However, our proposed algorithm has a control mechanism for preserving the necessary objectives (such as in section 3-(5) σ=0.0005) as well as the BZ algorithm. In other words, before reducing objectives, the method judges whether there are redundant objectives in the original objective set or the new subset. If yes, start or go on reducing the unnecessary objective. Else stop and output the corresponding results.

Anther problem is that the difference between the Avg IGD value and Best IGD value is obvious gap. It shows that the essential objective sets are not unique. Maybe the objective space randomness influences the algorithms. In the three algorithms, BZ algorithm is the best to obtain the steady subset, and then our algorithm, the last is JC algorithm.

Speculative to say that IGD measure is not the best one for objective reduction. The performance measures yielded completely contradictory result, for example, with the increase of the original objective number, the measure valve should be grown bit by bit, however, in some case, IGD decrease.

5. CONCLUSIONS AND FUTURE WORK

This paper presented the algorithm to identify the most redundant objective of a problems so that we can obtain a reduced set objectives that make search and the decision making process easier. The algorithm uses the least square method to fit every objective into a line (in fact we only pay attention to the slope of the line). Then we choose two redundant objectives which correspond to the minimum value in all of the difference between every two slopes. Finally, we consider the one is redundancy, whose mean of the slopes is less than other's. Also we use a parameter to judge whether there are redundant objectives.

The results show that the proposed algorithm is very competitive respect to other three similar algorithms recently proposed. Anther advantage of the proposed algorithm is low complexity, although JC algorithm is the best, and our algorithm is better than BZ method.

Additionally, we proposed the use of the inverted generational distance for measure the quality of the obtained reduced set of objectives. The three algorithms are equally matched.

We also introduce a new strategy not to divide the objective reduction problem into two kinds of algorithm, and we can choose the essential objectives set we need. Hopefully, this study will encourage more such, for devising more reliable and efficient methods of dimensionality reduction and eventually facilitate solution to large dimensional multi objective optimization problem.

From the experimental result using the difference between Best and Avg in the IGD result table, we find all of the three algorithms are unstable. We know JC algorithm is the most unstable, then our algorithm, the last is the BZ algorithm. How to obtain the stable subset will be a part of our further work.

Furthermore, how to measure the objective reduction algorithms and have a better application in the real world would be interesting to study, and they are focus in the following work.

Acknowledgments

Authors acknowledge the supports from the National Natural Science Foundation of China (60773047), the Natural Science Foundation of Hunan Province (05JJ30125) and the Keystone Science Research Project of the Education Office of Hunan Province (06A074).

6. References

[1] C. A. C. Coello, D. A. Vanveldhuizen, and G. Lamont. Evolutionary Algorithms For Solving MultiObjective Problems. Boston, MA: Kluwer Academic Publishers, 2002.

[2] F.di piero. Many Objectives Evolutionary Algorithms

and Applications to Water Resources Engineering. PhD Thesis School of Engineering, Computer Science and Mathematics, UK, August 2006.

[3] D. K. Saxena and K. Deb. Dimensionality Reduction of Objectives and constraints in MultiObjective Optimization Problems, A System Design Perspective. In congress on Evolutionary Computation (CEC 2008) IEEE Press. 3203-3210.

[4] K. Deb and D. K. Saxena. Searching for Pareto optimal Solutions through Dimensionality Reduction for Certain Large Dimensional MultiObjective Optimization Problem. In congtess on Evolutionary Computation (CEC 2006) IEEE Press. 3352-3360.

[5] D. K. Saxena and K. Deb. Non Line Dimensionality Reduction Procedures for Certain large Dimensional MultiObjective optimization problem. In Evolutionary Multi criterion Optimization (EMO 2007) 272-787.

[6] D. Brockhoff, and E. Zitzler.. Dimensionality Reduction in MutiObjective Optimization, the Minimum Objective Subject problem. In Operations Research Proceedings. Springer, 2007, 423-429.

[7] Brockhoff, D., Zitzler, E.: Dimensionality Reduction in Multiobjective Optimization with (Partial) Dominance Structure Preservation: Generalized Minimum Objective Subset Problems. TIK Report 247, ETH Zurich, Zurich, Switzerland (2006)

[8] D. Brockhoff, and E. Zitzler. Are All objectives Necessary? On Dimensionality Reduction in Evolutionary MultiObjective Optimization. In parallel problem solving from Nature, Springer. 2006, 533-542.

[9] A.L.Jaines, C.A.C.Coello and D.Chakraborty. Objective Reduction Using a feature Selection Technique. GECCO 08. 673--680, ACM Press, Atlanta, USA, July 2008,

[10] Deb Kalyanmoy, Lothar Thiele, Marco Laumanns and Eckart Zitzler. 2001a. Scalable Test Problems for Evolutionary Multi-Objective Optimization. TIK-Report No.112, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH) Zurich.

[11] K.Deb and D.K.Saxena. On Finding Pareto-Optimal Solutions Through Dimensionality Reduction for Certain Large- Dimensional Multi-Objective Optimization Problems. KanGAL Report Number 2005011.

[12] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan. A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation, 2002,6(2):182-197.

[13] Miqing Li and Jinhua Zheng. Spread Assessment for Evolutionary Multi-Objective Optimization, Evolutionary Multi-Criterion Optimization, 5th International Conference (EMO 2009), Nantes, France 216-230, 2009