

The Elite Optimality Procedure for Multi-Objective Evolutionary Algorithms

Truong Hong Trinh*

University of Economics, The University of Danang, Vietnam 71 Ngu Hanh Son st, Danang city, Vietnam

*Corresponding author

Abstract—Multi-Objective Evolutionary Algorithms (MOEAs) are likely used to identify non-dominated solutions or Pareto front (the known Pareto front) in multi-objective optimization problems. The quality of Pareto front depends on evolution strategies that are evaluated under performance metrics of generational distance, spacing, and error ratio. In this paper, a procedure of Elite optimality is proposed to transform the known Pareto front (PF_{known}) into the true Pareto front (PF_{true}). The Elite optimality procedure improves the quality of the Pareto fronts that deals with the biggest challenge in the multi-objective evolutionary algorithms.

Keywords—multi-objective imization; multi-objective evolutionary algorithms; pareto front; elite optimality

I. INTRODUCTION

Multi-objective optimization problem becomes increasingly attractive to both practitioners and researchers because most real-world problems contain multiple conflicting objectives that also poses a real challenge for the search of compromised solutions. To deal with the multi-objective optimization problems, there are many approaches from traditional methods to evolutionary methods. The typical traditional methods include weighted-sum [1], ϵ -constraint [2], goal programming [3]. These methods usually consume a relatively larger amount of computational time to find the tradeoff by identifying the non-dominated solutions or Pareto front. Meanwhile, evolutionary methods are likely used to identify Pareto front by using a population based approach with evolution strategies of Multi-Objective Genetic Algorithm (MOGA) [4], Multi-Objective Ant Colony Optimization (MOACO) [5], Multi-Objective Particle Swarm Optimization (MOPSO) [6], and Multi-Objective Differential Evolution (MODE) [7]. Since the quality of Pareto front depends on evolution strategies, the big challenge is how to identify the true Pareto front in the MOEAs. This paper proposes a procedure of Elite optimality to transform the known Pareto front (PF_{known}) into the true Pareto front (PF_{true}).

II. MOEA FRAMEWORK

Multi-objective optimization (MO) is a branch of optimization which mainly deals simultaneously with multiple objectives instead of single objective in traditional optimization problems. Multi-objective optimization can be found in various fields: product and process design, financial portfolio management, or wherever optimal decisions need to be taken in the presence of tradeoffs between two or more conflicting

objectives, such as maximizing profit and minimizing cost, or maximizing return and minimizing risk. In such problems, there is no single solution that is the best for all objectives. The problem is how to identify tradeoffs for the multi-objective problems. A tradeoff reflects the ratio of change in the values of the objective functions concerning the increment of one objective function that occurs when the value of some other objective function decreases [8].

The mathematical model for a multi-objective problem is given as follows:

$$\text{Min } (f_1(x), f_2(x), \dots, f_k(x))$$

Subject to

$$g_j(x) \geq 0, \quad \forall j = 1..m1$$

$$h_l(x) = 0, \quad \forall l = 1..m2$$

$$x \in S$$

In which, $x \in R^n$ is the vector of decision variables. $f_i(x)$ is a function of x , k is the number of the objective functions to be minimized. $g_j(x)$ and $h_l(x)$ are constraint functions of the problem.

Evolutionary algorithm is characterized by a population of solution candidates and evolution process that enables the guidance or combination of existing solutions to generate new population. The flowchart of MOEAs is illustrated in figure 1

In multi-objective optimization problems, the challenge is how to identify non-dominated solutions (Pareto front). The Elitism structure as mentioned in NSGA-II [9] is adopted with *non dominated sort()* the sorting procedure is to identify non-dominated solutions from the population. Elite mechanism is screened to eliminate inferior solutions in Elite archive. As a result, solutions in Elite archive are the best non-dominated solutions found so far that is represented as the known Pareto front.

One of the biggest challenges for most MOEAs is how to select the candidates in Elite archive to help guiding the evolution process for generating the new population. The evolution mechanisms are different in proposed evolutionary algorithms. For instance, multi-objective genetic algorithm (MOGA) and multi-objective differential evolution (MODE) algorithm combine current members in Elite archive (or with external Elites) to generate new population, while multi-

objective ant colony optimization (MOACO) and multi-objective particle swarm optimization (MOPSO) algorithms update the guidance for current members in Elite archive to generate new population.

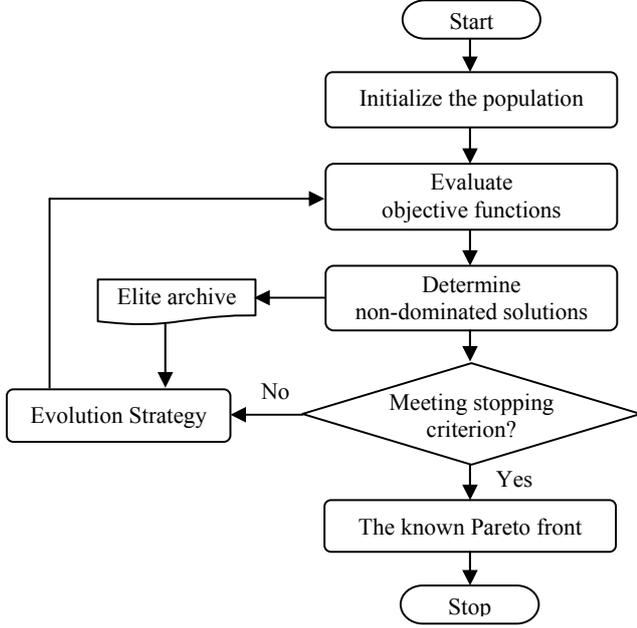


FIGURE I. THE FLOWCHART OF MOEAS

The evolution strategy defines the way of combination or movement of members in Elite archive. Movement strategies for MOPSO algorithm and Mutation strategies for MODE algorithm are explored under the procedure of *select guidance()*, and the procedure of *select mutation()*, respectively. These strategies are discussed as options to fully utilize the Elite archive as guidance for the search of Pareto optimal set that is very critical to improve the quality of Pareto front. In literature, the following performance metrics is often used to measure the quality of Pareto front:

1. Generational Distance (GD): The metric proposed by Van Veldhuizen and Lamont [10] representing in the average how “far” PF_{known} is from PF_{true} and is defined as:

$$GD = \frac{\left(\sum_{i=1}^n d_i^p \right)^{1/p}}{n}$$

Where, n is the number of Elites in PF_{known} , $p = 2$, and d_i is the nearest Euclidean distance (in objective function space) between reference Elite and the optimal solution of PF_{true} . A result of 0 indicates $PF_{true} = PF_{known}$, and other value PF_{known} deviates from PF_{true} .

2. Spacing (SP): Schott [11] introduced this metric to measure the spread of Elites on PF_{known} and is mathematically represented by:

$$SP = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (\bar{d} - d_i)^2}$$

Where, n is the number of vectors in PF_{known} , and $d_i = \min_j \left(|f_1^i(\vec{x}) - f_1^j(\vec{x})| + |f_2^i(\vec{x}) - f_2^j(\vec{x})| \right)$, $i, j = 1..n$, \bar{d} is the mean of all d_i . A value of 0 indicates that all members of PF_{known} are equidistantly spaced.

3. Error Ratio (ER): Van Veldhuizen [12] was proposed this metric to measure the percentage of Elites of PF_{known} that are not belong to PF_{true} , and is mathematically represented by:

$$ER = \frac{\sum_{i=1}^n e_i}{n}$$

Where, n is the number of vectors in PF_{known} and

$$e_i = \begin{cases} 0 & \text{if vector } i, \forall i = 1..n \in PF_{true} \\ 1 & \text{otherwise} \end{cases}$$

$ER = 0$ indicates that every Elite produced by the algorithm in PF_{known} is actually in PF_{true} , and $ER = 1$ indicates that none of the Elites of PF_{known} are in PF_{true} .

Even the evolution strategy affects the quality of the known Pareto front (PF_{known}) in terms of GD, SP and ER. The problem is how to transform Elites on PF_{known} into PF_{true} (the true Pareto front). A procedure is proposed for Elite optimality process including *Elite optimal check ()* and *best distance search ()*.

III. ELITE OPTIMALITY PROCEDURE

Most real world decision problems involve multiple conflicting objectives. In such problems, it requires to identify tradeoffs (Pareto fronts) between multiple objectives, and find Pareto optimal solution nearest the most preferred solution of a decision maker [13]. While the evolution mechanisms and the Movement/Mutation strategies in MOEAs affect the quality of Pareto front in terms of performance metrics (GD, SP and ER), the quality of Pareto front has influence on the effectiveness of decision making.

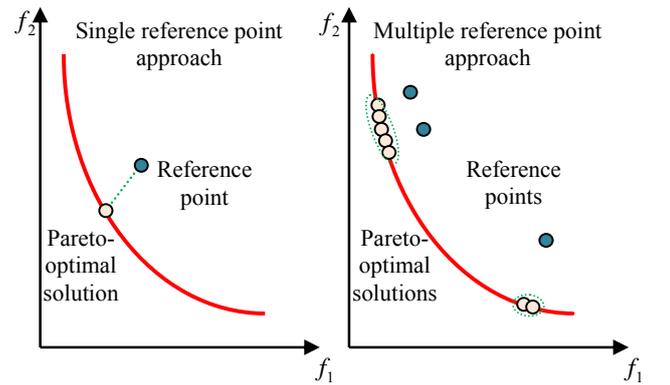


FIGURE II. REFERENCE POINT APPROACHES FOR MULTI-OBJECTIVE OPTIMIZATION

In literature, most MOEAs provide non-dominated solutions known as Pareto front (PF_{known}). The problem is how to transform PF_{known} into PF_{true} (the true Pareto front). By solving

an achievement scalarizing problem, Wierzbicki [14] suggested the reference point approach for achieving a Pareto-optimal solution nearest to a reference point of aspiration level. Meanwhile, multiple reference point approach explores multiple reference points simultaneously rather than one reference point at a time for achieving the preferred regions of Pareto-optimality, the ideas to transform PF_{known} into PF_{true} . Deb and Sundar [15] proposed a modified EMO (Evolutionary Multi-objective Optimization) procedure based on the elitist non-dominated sorting GA (Genetic Algorithm) or NSGA II. The procedures will provide the decision-maker with a set of solutions near the reference points so that a better and a more reliable decision can be made.

In fact, the reference point approaches provide a weak Pareto-optimal solution(s) nearest to the reference point(s). For that reason, this research proposes a procedure of Elite optimality process for identifying the true Pareto front from the reference Elites on PF_{known} . The procedure includes *Elite optimal check* () and *best distance search* ().

The proposed procedure for Elite optimality is described as follows:

- Step 1: Initialize Elites from PF_{known} . For each Elite,
- Step 2: Check Elite optimality by *Elite optimal check* ()
- Step 3: Search the nearest distance to PF_{true}
- If (Elite is optimal) Go to Step 4
- Otherwise, best distance check (), Update Elite, Repeat Step 3
- Step 4: If (all Elites are checked) Archive PF_{true} . Stop. Otherwise, Go to Step 2

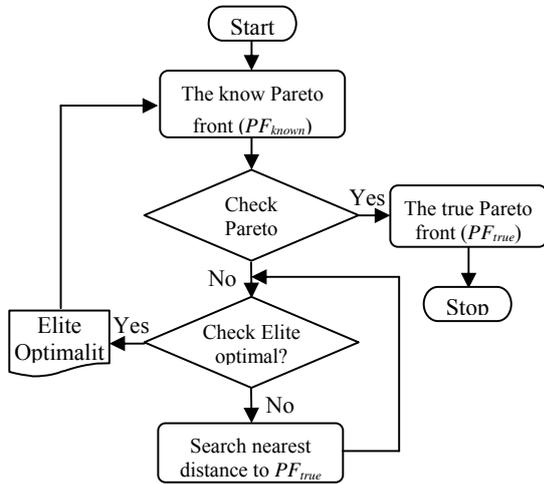


FIGURE III. THE FLOWCHART OF ELITE OPTIMALITY

The algorithm of *Elite optimal check* ():

$$\text{Max } \sum_{i=1}^k \varepsilon_i$$

Subject to

$$f_i(x) + \varepsilon_i = f_i^{**}, \quad \forall i = 1..k$$

$$g_j(x) \geq 0, \quad \forall j = 1..m1$$

$$h_l(x) = 0, \quad \forall l = 1..m2$$

$$x \in S, \quad \varepsilon_i \geq 0, \quad \forall i = 1..k$$

An algorithm of *Elite optimal check* () is developed under a mathematical programming that is used to check whether a particular Elite is optimal or not? An Elite is optimal only if

value of objective function is zero ($\sum_{i=1}^k \varepsilon_i = 0$), and the

reference solution of $(f_1^{**}, f_2^{**}, \dots, f_k^{**})$ is identified as Elite optimal solution (Elite optimality). If value of objective function is infinite and nonzero, then the reference solution $(f_1^{**}, f_2^{**}, \dots, f_k^{**})$ is not the nearest optimal solution. An algorithm of *best distance search* () is used to search the best solution with the nearest distance to the true Pareto front so far.

The algorithm of *best distance search* ():

$$\text{Min } \sqrt{\sum_{i=1}^k \varepsilon_i^2}$$

Subject to

$$\sqrt{\sum_{i=1}^k \varepsilon_i^2} > d_{\min}$$

$$f_i(x) + \varepsilon_i = f_i^*, \quad \forall i = 1..k$$

$$g_j(x) \geq 0, \quad \forall j = 1..m1$$

$$h_l(x) = 0, \quad \forall l = 1..m2$$

$$x \in S, \quad \varepsilon_i \geq 0, \quad \forall i = 1..k$$

The mathematical programming with the minimum of the

objective function $\sqrt{\sum_{i=1}^k \varepsilon_i^2}$ and the constraint of $\sqrt{\sum_{i=1}^k \varepsilon_i^2} > d_{\min}$

is to search the best solution $(f_1^{**}, f_2^{**}, \dots, f_k^{**})$ that is nearest the Elite $(f_1^*, f_2^*, \dots, f_k^*)$. Whenever a solution is found from the algorithm of *best distance search* (), value of the best solution $(f_1^{**}, f_2^{**}, \dots, f_k^{**})$ and the nearest distance d_{\min} are updated as follows:

$$f_i^{**} = f_i^* - \varepsilon_i, \quad \forall i = 1..k$$

$$d_{\min} = \sqrt{\sum_{i=1}^k \varepsilon_i^2}$$

FIGURE IV illustrates the process of Elite optimality, where solution B represents the best solution found so far, but not optimal solution. The search process will be finished only if solution B obtained at C where solution C is an optimal solution corresponding to Elite A (reference Elite). The

distance between A and C is the nearest Euclidean distance from the reference Elites A on PF_{known} to the optimal solution C on PF_{true} .

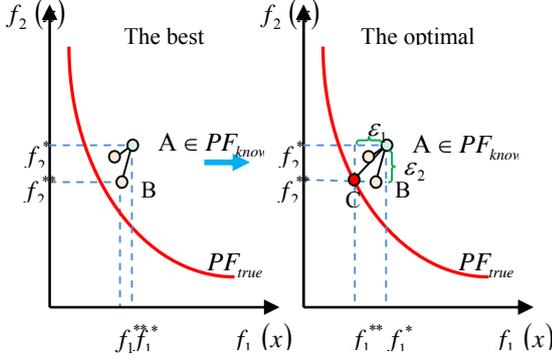


FIGURE IV. THE PROCESS OF ELITE OPTIMALITY

IV. ILLUSTRATION

The multi-objective co-production model is employed for solving the objective tradeoff problem between customer utility (U) and total cost (F) in the hypothetical service system as follows:

Multi-objective co-production model:

$$\text{Max } U = w_s Q - w_L L \quad \{\text{Total Utility}\}$$

$$\text{Min } F = w_K K + w_H H + w_L L \quad \{\text{Total Cost}\}$$

Subject to

$$Q = A \times K^\alpha \times H^\beta \times L^\gamma \quad \{\text{Co-production function}\}$$

$$\forall K, H, L \geq 0$$

The co-production function (Q) is assumed well-defined function with given mean productivity (A) and output elasticity of inputs (α, β, γ). Customers have the same unit cost of customer input (w_L) and unit value of service outcome (w_S), unit cost of firm capital (w_K), unit cost of firm employee (w_H).

There are three decision variables (K, H, L) in the co-producer model. These independent variables are randomly generated in the range of [lower bound, upper bound]. From assigned values for decision variables, values of objective variables (total utility and total cost) are computed through co-production function. FIGURE V illustrates solution representation for the co-producer model.

As mention earlier, decision variables in multi-objective problems are randomly generated in the range of [lower bound, upper bound] = [bL, bU]. These random values for the decision variables may not satisfy constraints in the model. Thus, it requires a procedure to check constraint conditions in the model. In addition, objective values are computed from the objective functions with assigned values of decision variables. These solutions may not be non-dominated solutions. Therefore,

the sorting procedure is used to eliminate dominated solutions at each iteration. In complexity models, the solutions are usually not represented as multiple dimension space directly. For that reason, encoding/decoding procedure is used to express vectors (individual solutions) under form of single dimension vectors, and then these vectors are decoded to represent solutions in the form of multiple dimension space.

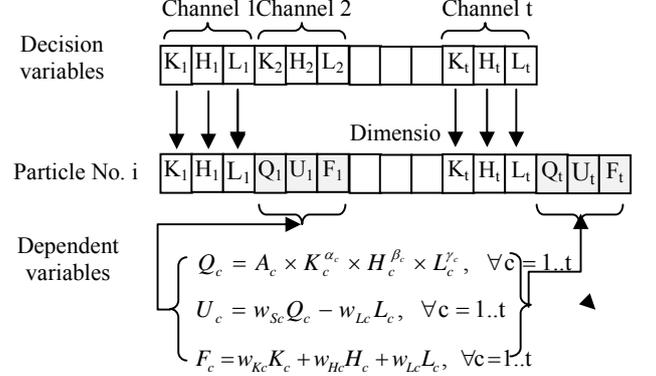


FIGURE V. SOLUTION REPRESENTATION

Multi-objective evolutionary algorithms (MOEAs) attempt to identify the best non-dominated solutions represented as the known Pareto front (PF_{known}). The additional algorithms for Elite optimality process are used to transform non-dominated solutions on PF_{known} into optimal solutions on PF_{true} for the objective tradeoff problem as follows:

Elite optimal check ():

$$\text{Max } (\varepsilon_1 + \varepsilon_2)$$

Subject to

$$(w_s Q - w_L L) - \varepsilon_1 = f_1^{**}$$

$$(w_K K + w_H H + w_L L) + \varepsilon_2 = f_2^{**}$$

$$Q = A \times K^\alpha \times H^\beta \times L^\gamma$$

$$\forall K, H, L \geq 0, \varepsilon_1, \varepsilon_2 \geq 0$$

Where, (K, H, L) and $(\varepsilon_1, \varepsilon_2)$ are variables, (f_1^{**}, f_2^{**}) is value of objective functions in the objective tradeoff problem. For each Elite in PF_{known} , if value of $(\varepsilon_1 + \varepsilon_2)$ is zero, then x^* is the Pareto optimal solution. Otherwise, value of decision variable (x) corresponding to nonzero value of the objective function $(\varepsilon_1 + \varepsilon_2)$ is a Pareto optimal solution.

best distance search ():

$$\text{Min } \sqrt{\varepsilon_1^2 + \varepsilon_2^2}$$

Subject to

$$\sqrt{\varepsilon_1^2 + \varepsilon_2^2} > d_{\min}$$

$$\begin{aligned} (w_S Q - w_L L) - \varepsilon_1 &= f_1^* \\ (w_K K + w_H H + w_L L) + \varepsilon_2 &= f_2^* \\ Q &= A \times K^\alpha \times H^\beta \times L^\gamma \\ \forall K, H, L \geq 0, \varepsilon_1, \varepsilon_2 &\geq 0 \end{aligned}$$

Where, d_{\min} is the feasible solution with the nearest distance from reference solution found so far. When a solution is found from the algorithm of best distance search (), value of the best solution (f_1^{**}, f_2^{**}) and the nearest distance d_{\min} are updated as follows:

$$\begin{aligned} f_1^{**} &= f_1^{**} + \varepsilon_1, & \text{for maximum function} \\ f_2^{**} &= f_2^{**} - \varepsilon_2, & \text{for minimum function} \\ d_{\min} &= \sqrt{\varepsilon_1^2 + \varepsilon_2^2}, & \text{Euclidean distance} \end{aligned}$$

V. CONCLUSION

This paper introduces the MOEA framework that is used for solving the multi-objective optimization problems. From the MOEA framework, the known Pareto front (non-dominated solutions) is identified by the MOEAs, in which the quality of Pareto front depends on evolution strategies that are evaluated under performance metrics of generational distance, spacing, and error ratio. From the known Pareto front (PF_{known}) of the MOEAs, the Elite optimality procedure is used to transform the known Pareto front (PF_{known}) into the true Pareto front (PF_{true}). The algorithms are used to check Elite optimality under Elite optimal check () and Elite movement under guidance of best distance search () to move Elites from the known Pareto front (PF_{known}) to the true Pareto front (PF_{true}). The paper contribute the Elite optimality procedure for the MOEAs to transform the known Pareto front (PF_{known}) into the true Pareto front (PF_{true}), in which the quality of the true Pareto front depends mainly on the known Pareto front (PF_{known}) from the MOEAs.

ACKNOWLEDGMENT

This paper is developed from a part of the author's dissertation at Asian Institute of Technology, Thailand. The author would like to thank Prof. Voratas Kachitvichyanukul. Acknowledgments are also extended to anonymous reviewers for their valuable comments.

REFERENCES

- [1] J. L. Cohon, Multi-objective programming and planning: New York: Academic Press, 1983.
- [2] V. Chankong and Y. Y. Haimes, Multiobjective decision making theory and methodology. New York: Elsevier Science, 1983.
- [3] R. E. Steuer, Multiple criteria optimization: Theory, computation, and application: John Wiley & Son Inc., 1986.
- [4] A. Konak, D. W. Coit, and A. E. Smith, "Multi-objective optimization using genetic algorithms: A tutorial," Reliability Engineering & System Safety, vol. 91, pp. 992-1007, 2006.
- [5] D. Angus and C. Woodward, "Multiple objective ant colony optimisation," Swarm intelligence, vol. 3, pp. 69-85, 2009.
- [6] M. Reyes-Sierra and C. C. Coello, "Multi-objective particle swarm optimizers: A survey of the state-of-the-art," International journal of computational intelligence research, vol. 2, pp. 287-308, 2006.
- [7] W. Wisittipanich and V. Kachitvichyanukul, "Mutation strategies toward Pareto front for multi-objective differential evolution algorithm," International Journal of Operational Research, vol. 19, pp. 315-337, 2014.
- [8] J. Branke, K. Deb, K. Miettinen, and R. Slowinski, Multiobjective Optimization: Interactive and Evolutionary Approach: Springer-Verlag Berlin Heidelberg, 2008.
- [9] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," Evolutionary Computation, vol. 6, pp. 182-197, 2002.
- [10] D. A. Van Veldhuizen and G. B. Lamont, "Multiobjective Evolutionary Algorithm Research: A History and Analysis," in Department of Electrical and Computer Engineering Wright-Patterson AFB, OH: Air Force Institute of Technology, 1998.
- [11] J. R. Schott, "Fault tolerant design using single and multicriteria genetic algorithm optimization," in Department of Aeronautics and Astronautics Cambridge, MA: Massachusetts Institute of Technology, 1995.
- [12] D. A. Van Veldhuizen, "Multiobjective evolutionary algorithms: Classifications, analyzes, and new innovations," in Department of Electrical and Computer Engineering Wright-Patterson AFB, OH: Air Force Institute of Technology, 1999.
- [13] J. B. Yang, "Minimax reference point approach and its application for multiobjective optimisation," European Journal of Operational Research, vol. 126, pp. 541-556, 2000.
- [14] A. P. Wierzbicki, "The use of reference objectives in multiobjective optimisation," in Multiple Criteria Decision Making Theory and Applications, G. Fandel and T. Gal, Eds. Berlin: Springer-Verlag, 1980, pp. (pp. 468-486).
- [15] K. Deb and J. Sundar, "Reference point based multi-objective optimization using evolutionary algorithms," International Journal of Computational Intelligence Research vol. 2, pp. 273-286, 2006.