Multi-objective Evolutionary Algorithms for Coping with Single-Objective Optimization Problems Rationally

– Comparison, Enhancement and Post-optimal Evolution –

Jaekyu Yoo
school of economics
kanazawa university
Kanazawa, Japan
yoo@staff.kanazawa-u.ac.jp

Yoshiaki Shimizu
optimization engineering lab.
toyohashi university of technology
Izumi, Japan
shimizu@me.tut.ac.jp

***Abstract*— To resolve various difficult problems encountered in modern technologies, multi-objective evolutionary algorithms (MOEAs) have been studied aggressively in these decades. Since they can reveal a clear trade-off relation among the conflicting objectives effectively, they are available for multi-objective optimization problem (MOP). To expand such availability more in advance, we previously proposed a novel idea for solving single-objective optimization problem (SOP) using MOEA and applied it to some methods by using scalarization in MOP (scalarized MOP, hereinafter). Actually, such idea was studied in terms of MOEA such as NSGA-II alone. Hence to find out more effective methods, this paper will first compare the performance with other popular MOEAs. Then, regarding the best method (NSGA-II, after all), we reveal a special property of the individual solution in population and develop a new idea on a basis of non-Pareto optimal (best performance) dominance to enhance the solution ability. In numerical experiments, first, we have compared the solution ability among NSGA-II, MODE and MOPSO through a set of benchmarks. Then, in terms of NSGA-II as the best method, scalarized MOPs have been solved to examine the merit of the new idea. They are four bi-objective mechanical design problems and a real-world car structure design. Moreover, the significance of the post-optimal evolution is emphasized especially for the scalarized MOPs. Through those demonstrations, we claim the proposed framework can provide a practical procedure for rational decision making through SOP and/or MOP. discuss some prospects towards the scalable cost accounting.**

Keywords*— Optimization, Multi-Objective Evolutionary Algorithm, NSGA-II, Post-Optimal Evolution, scalarized Multi-Objective Optimization Problem

I. Introduction

To resolve various difficult problems encountered in modern technologies, various optimization methods have been applied successfully in many years. Particularly, in a field known as multi-objective optimization, multi-objective evolutionary algorithms (MOEAs) have been studied aggressively in these decades. Since they are viewed as practical methods and can reveal a clear trade-off relation among the conflicting objectives (multi-objective analysis [1]) they are available for multi-objective optimization problem (MOP) that aims at obtaining a unique solution known as preferentially optimal solution. To expand such availability of MOEAs more in advance, we previously proposed a novel idea for solving single-objective optimization problem (SOP) using MOEA [2] and applied it to the real world optimization problem [3].

Actually, such idea was studied in terms of MOEA such as NSGA-II alone. Hence to find out more effective methods, this paper will compare the previous performance with other popular MOEAs such as MODE (Multi-Objective Differential Evolution) and MOPSO (Multi-Objective Particle Swarm Optimization). In addition, after revealing a special property of individual solution in population, we will develop a new idea to enhance the solution ability for the best method (NSGA-II, after all). Actually, it relies on a basis of non-Pareto optimal (best performance) dominance in the foregoing framework.

To compare the solution ability among MOEAs, in numerical experiments, we will solve a set of benchmark problems. Then, in terms of NSGA-II as the best method, scalarized MOPs have been solved to examine the enhanced merit of the new idea. First, four bi-objective mechanical design problems are concerned by the stiff value function methods (weighing and ε-constraint) and commercial solver LINGO [URL-1]. Then a real-world car structure design problem is solved by an adaptive approach termed MOON^2 (Multi-Objective Optimizer using value function modelled by Neural Network) [4].

Moreover, we engage in the post-optimal evolution that is to be done for re-evaluating the prior optimization result before the final decision. Here, we emphasize its significance especially for the scalarized MOPs since they usually involve uncertain parameters referring to subjective preference of decision maker (DM).

The rest of this paper is organized as follows. In Section 2, problem statement is described clearly. Section 3 explains about the old and new ideas. In Section 4, examining solution ability through comparison with other MOEAs, we move on the bi-objective engineering design problems. Then, significance of the post-optimal evolution is discussed demonstratively. Some conclusions are given in Section 5.

II. Problem Statements

In general, SOP is formulated as [Problem 1].

[Problem 1] min $f(\boldsymbol{x})$ subject to $\boldsymbol{x} \in$ X

where $\boldsymbol{x}$ denotes a decision variable vector, X a feasible region and f a scalar objective function.

Though many mathematical programming methods have been traditionally applied, in modern optimization, they are likely replaced with meta-heuristic methods. This is because the latters can practically cope with various simulation models and expect to obtain a global solution without rigid mathematical conditions even for complicated problems.

Meanwhile, MOP is formulated as follows.

[Problem 2] min $\boldsymbol{F}(\boldsymbol{x}) = \{f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \dots f_N(\boldsymbol{x})\}$

subject to $\boldsymbol{x} \in X$

where $\boldsymbol{F}$ denotes a vector objective function some elements of which conflict with one another. The aim of MOP is to obtain a unique preferentially optimal solution relaying on the DM's preference.

As a widely used approach for MOP, certain scalarized methods have been utilized traditionally to many applications due to their simplicities [5]. The basic idea is to transform appropriately the original MOP into SOP. As its popular form, weighting method is formulated as follows.

[Problem 3] min $f(\boldsymbol{x}) = \sum_i^N w_i f_i(\boldsymbol{x})$ subject to $\boldsymbol{x} \in X$

Another one known as ε -constraint method is given by

[Problem 4] min $f(\boldsymbol{x}) = f_i(\boldsymbol{x})$

subject to $f_j(\boldsymbol{x}) \leqq f_{j\text{-opt}} + \varepsilon_j, \ \forall j \neq i \text{ and } \boldsymbol{x} \in X$

where w_i denotes the weighting coefficient representing the relative importance of i-th objective and ε_j the upper bound compromise from the optimal value $f_{j\text{-opt}}$ for j-th objective.

Though those formulations are understandable, there exist no ways to decide such preference parameters like weights and ε values beforehand correctly. That is an inherent weakness of those approaches. Hence, we call those stiff methods.

To overcome such stiffness, we proposed an adaptive scalarized method named MOON[2]. It tries to identify the value function of DM through a suitable artificial neural network (NN) beforehand. Such NN will be trained using the training data gathered from DM by the AHP-like pair-wise comparisons on his/her preference. Through such elaborate assessment on the value system, we can transform the original MOP into SOP as follows.

[Problem 5] max $V_{NN}(\boldsymbol{F}(\boldsymbol{x}))$ subject to $\boldsymbol{x} \in X$

Here, noting the specific uncertain properties still embedded in MOP, we should note the importance of the post-optimal evolution pointed out already. For this purpose, we can apply the elite induced multi-objective evolutionary algorithm (EI-MOEA [6]) as follows.

(1) Select some elite solutions around the prior solution.

(2) Apply MOEA by incorporating the elite solutions into a set of random initial solutions.

III. ENHANCEMENTS TO SOLVE SOP BY MOEA

Previously, we noted the following propositions.

Proposition 1: Objectives min $f(\boldsymbol{x})$ and max $f(\boldsymbol{x})$ always conflict with each other.

Proposition 2: "[Problem 6] min $\{f(\boldsymbol{x}), -f(\boldsymbol{x})\}$ subject to $\boldsymbol{x} \in X$" is viewed as a bi-objective problem (minus formulation).

Then, due to **Proposition 2**, we proposed a unique procedure to solve SOP or scalarized MOP by MOEA as follows (hereinafter, our approach).

(1) Apply a certain MOEA for the above [Problem 6].

(2) Select the solution with the minimum value of $f(\boldsymbol{x})$ as the optimal solution for the original [Problem 1].

Regarding the approaches associated with this topic, a few ideas are proposed for the constrained SOP. The first one [7] proposed a scheme that transforms the original problem into an unconstrained bi-objective problem by considering a measure of the constraint violations as the second objective. Another one [8] tries to transform the problem into an unconstrained MOP having the original objective function and every constraint as separate objectives. In this case, the constrained SOP is converted into a MOP with N objectives when the number of constraints is N − 1.

It should be noted, in these approaches, trade-off is to be considered between the optimality and the feasibility. Hence, it becomes quite hard to derive some promising feasible solutions efficiently if any particular operations would not be introduced in the algorithm. Accordingly, we cannot directly apply any conventional MOEAs to solve the problem. Moreover, the second approach refers likely to many-objective problem that becomes more difficult to solve since numbers of constraint are large for practical real world applications.

Against these defects, our above idea can cope both with the unconstrained and constrained SOP by just applying the usual MOEA. In other words, we can solve the original SOP even if we do not have any EA and mathematical programming solvers or certain knowledge about their usage.

By the way, it is unnecessary to derive a widely spread distribution of Pareto front in our dealing. It is enough to obtain only several solutions. This is also true for the post-optimal evolution carried out after that. To note these facts, we modified the algorithm of NSGA-II [9] so that the population size will decrease along with the generation. Pseudo code of such algorithm called down-sizing NSGA-II is given as follows.

```
if(gen >α x gener)
{ popsize = β x popsize;
  if(popsize < minpop) popsize = minpop; }
```

where α and β are positive parameters (<1). And *gen*, *gener*, *popsize* and *minpop* represent current generation, its total one, population size and desired final size, respectively.

Since we deployed those ideas and examined their effects only for NSGA-II, it is meaningful to compare the performance with the other popular MOEAs like MODE and MOPSO before considering enhancement. Then, for the best match of our idea among those, we try to enhance the algorithm by introducing a simple operation in terms of the following property,.

Proposition 3: Rank of every individual is always 1 when solving the unconstrained problem of [Problem 6].

From this, for unconstrained problems, it becomes redundant to compute the rank every time. Moreover, we can avoid such a risk that the individuals having smaller objective value $f(\boldsymbol{x})$ are not selected if usual Pareto optimum dominance would be applied. This is because when the ranking is tie, the selection is to be done in terms of another basis such as crowding distance. Hence, it is better to adopt a non-Pareto optimum dominance to enhance the solution ability in the present case. What we call here the best performance dominance (BPD) is defined as follows:

Individual $\boldsymbol{x}^i$ dominates $\boldsymbol{x}^j$ if $f(\boldsymbol{x}^i) < f(\boldsymbol{x}^j)$

As a matter of fact, the above proposition is not satisfied for the constrained problems that are obeyed by the constrained dominance that is defined as follows.

Individual $\boldsymbol{x}^i$ dominates $\boldsymbol{x}^j$,

(1) if $\boldsymbol{x}^i$ is feasible while j infeasible

(2) if $\boldsymbol{x}^i$ has smaller vioration amount than $\boldsymbol{x}^j$ when both are infeasible

(3) due to the Pareto optimum dominance when both are feasible

In this case, we can apply BPD only for the above Case (3). Hence, though the rank of every individual is always 1 in unconstrained problems, there are several ranks during the evolution in constrained ones. Moreover, as a subsidary effect, computational effort is saved by this BPD in any case.

As a summary of this section, we repeat the significance of the proposed idea as follows. Just by an appropriate MOEA, we can cope with a variety of interests in engineering optimization regardless whether it is given as SOP or MOP. In this line, the proposed idea is promising to expand the availability and performance of MOEA greatly.

TABLE I. COMPARISON AMONG MOEAS USING BENCHMARKS

Benchmark	MODE	MOPSO	NSGA-II
Shekel's fox hole: f_{opt} = 0.998004	128.394 [a] 0% (0) [b]	54.3947 68% (21)	0.998003 100% (31)
Schwefel: f_{opt} = 0.0	326.482 0% (0)	84.6800 0% (0)	34.3863 71% (22)
De Jong: f_{opt} = 3905.93	3909.47 100% (31)	3905.96 100% (31)	3905.93 100% (31)
Goldstein & Price: f_{opt} = 3.0	81.8914 0% (0)	4.20729 6% (2)	3.00706 97% (30)
Branin: f_{opt} = 0.397727	2.86889 0% (0)	0.414448 32% (10)	0.474839 97% (30)
Martin & Gaddy: f_{opt} = 0.0	0.702028 0% (0)	2.95844E-3 84% (26)	5.63677E-5 100% (31)
Rosenbrock: f_{opt} = 0.0	3.54863 0% (0)	0.0321421 23% (7)	7.99531E-3 68% (21)
4D-Rosenbrock: f_{opt} = 0.0	83.8490 0% (0)	3.62872 0% (0)	1.18818 3% (1)
Hyper sphere: f_{opt} = 0.0	18.6844 0% (0)	0.930556 0% (0)	7.59519 E-4 100% (31)
Griewangk: f_{opt} = 0.0	0.936496 0% (0)	0.0127924 23% (7)	0.0107789 32% (10)
Total score	10% (31)	34% (104)	77% (238)

[a] Objective value averaged over 31 trials,

[b] % (Success#), where *Success#* = +1 if ($|f_{opt} - f(\boldsymbol{x})| < 0.01*(1.0 + |f_{opt}|)$)

IV. NUMERICAL EXPERIMENT

A. Comparison among MOEAs

Previously, to evaluate the solution ability, we solved ten benchmark problems many of which have multiple peaks of objective functions. Actually, we applied our idea in terms of NSGA-II and compared its performance to several methods, i.e., a direct search known as N-M (Nelder & Mead) and four evolutionary methods such as DE (Differentially Evolution), PSO (Particle Swarm Optimization), GA (Genetic Algorithm) and SA (Simulated Annealing). Then, we revealed the total performance of our procedure is third place a bit behind DE and PSO but satisfactory as a global solver even for SOP.

Here, to find more effective methods, we compare the foregoing performance with the other MOEAs such as MODE and MOPSO. For this purpose, we used the codes named nsga2, MOPSOCD [10] in the R library and transformed R code from MODE.m in Matlab [URL-2] under the default parameter settings. MOPSOCD uses crowding distance computation to ensure even spread of non-dominated solutions while MODE applies the greedy selection using a dominance relation.

Every problem was solved 31 independent runs with different random seeds under the conditions for the population size (*popsz#*) and the generation time (*gener#*) as follows: *popsz#* = min (10*D*, 60) and *gener#* = min(100**popsz*$^{0.7}$, 2000), respectively. Here, *D* denotes the dimension of decision variables. In Table 1, those results are summarized regarding the average objective value and *Success#* defined below the table. We know MODE did not go well with our approach at all and MOPSOCD did not so well. Since NSGA-II is known to be the best match, we continue to use this method for the following evaluations.

B. Effect of Enhancements for Constrained Problems

1) Engineering problems

In this section, we turn our attention to examine the efficiency and applicability of our approach (refer to Propositions 2 & 3) in terms of the four engineering problems studied elsewhere (Refer to Ref.[11] more in detail,). They are all bi-objective constrained minimization problems. Actually, we solved those as the stiff scalarized problems given by [Problem 3] and [Problem 4] under the "minus" formulation. This time, we used the open C code of NSGA-II developed by Deb [URL-3] with the parameter settings such as crossover probability and distribution index: 0.75 & 10, respectively and mutation probability and distribution index: 0.2 & 50, respectively. On the other hand, population size and generation number are set as *popsz#* =10*D* and *gener#* =10*popsz#*, respectively

The first problem formulated below is a two-bar truss design problem minimizing total volume of bars and stress on bar AC subject to three constraints on stress and volume with respect to the variables ($x_1 - x_3$) as shown in Fig.1 (a). Here, two kinds of stiff scalarized problem were solved with weights of (0.5, 0.5) and under ε-constraint like $f_1(\boldsymbol{x}) \leqq 0.2$, respectively.

$$\min\{ f_1(\boldsymbol{x}) = 0.01x_1\sqrt{x_3^2+16} + x_2\sqrt{x_3^2+1},$$

$$f_2(\boldsymbol{x}) = 0.2\sqrt{x_3^2+16}\,/(x_2x_3)\}$$

subject to

$$1 - f_1(\boldsymbol{x}) \geq 0$$

$$1 - f_2(\boldsymbol{x}) \geq 0$$

$$1 - 0.8\sqrt{x_3^2+1}/(x_2x_3) \geq 0$$

$$0 \leq x_1, x_2 \leq 20,\ 1 \leq x_3 \leq 3$$

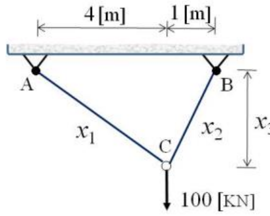

Fig. 1. (a) Scheme of two-bar truss problem

The second problem formulated below is to minimize cost and end deflection for a welded beam subject to four constraints on shear stress, bending stress and buckling load. Meanwhile, variables (x_1 – x_4) are lengths as shown in Fig.1 (b). In this case, weights are (0.00025, 0.99975) and ε-constraint is $f_1(\boldsymbol{x}) \leqq 10$, respectively.

$$\min\{ f_1(\boldsymbol{x}) = 1.105\,x_1\,x_4^2 + 0.048\,x_2\,x_3(14+x_1),$$

$$f_2(\boldsymbol{x}) = 2.1952/(x_2^3\,x_3)\}$$

subject to

$$\tau = \sqrt{\tau'^2 + x_1\,\tau'\,\tau''/\mathrm{R} + \tau''^2} \leq 30000$$

$$\sigma = 5.04\cdot 10^5\,/(x_2^2\,x_3) \leq 13600$$

$$x_3 \geq x_4$$

$$6.4746\cdot 10^4(1-0.3x_2)\,x_2x_3^3 \geq 6000$$

$$0.1 \leq x_1, x_2 \leq 10,\ 0.125 \leq x_3, x_4 \leq 5$$

where

$$\tau' = 6000/(\sqrt{2}x_1\,x_4),\ \ \tau'' = 6000(14+0.5\,x_1)\mathrm{R}/\mathrm{J}$$

$$\mathrm{R} = \sqrt{0.25\{x_1^2 + (x_2+x_4)^2\}},$$

$$\mathrm{J} = \sqrt{2}x_1\,x_4\{x_1^2/12 + (x_2+x_4)^2/4\}$$

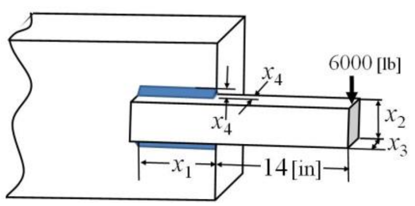

Fig.1. (b) Scheme of welded beam problem

The third problem formulated below is a disc brake design problem minimizing mass of the brake and stopping time subject to five constraints on the minimum distance between radii, the maximum length of brake, pressure, temperature and torque limitations (Fig.1 (c)). Meanwhile variables are inner radius of discs x_1, outer radius of discs x_2, engaging force x_3 and number of friction surfaces x_4. Here, we set weights at (0.8, 0.2) and ε-constraint as $f_1(\boldsymbol{x}) \leqq 1$, respectively.

$$\min\{ f_1(\boldsymbol{x}) = 4.9\cdot 10^{-5}\,(x_2^2 - x_1^2)(x_4+1),$$

$$f_2(\boldsymbol{x}) = = 9.82\cdot 10^6(x_2^2 - x_1^2)/(x_3x_4)/(x_2^3 - x_1^3)\}$$

subject to

$$x_2 - x_1 - 20 \geq 0$$

$$30 - 2.5(x_4+1) \geq 0$$

$$0.4 - x_3\,/\{3.14(x_2^2 - x_1^2)\} \geq 0$$

$$1.0 - 2.22\cdot 10^{-3}\,x_3\,(x_2^3 - x_1^3)/(x_2^2 - x_1^2)^2 \geq 0$$

$$0.0266\,x_3\,x_4\,(x_2^3 - x_1^3)/(x_2^2 - x_1^2) - 900 \geq 0$$

$$55 \leq x_1 \leq 80,\ 75 \leq x_2 \leq 110,$$

$$1000 \leq x_3 \leq 3000,\ 2 \leq x_4 \leq 20$$

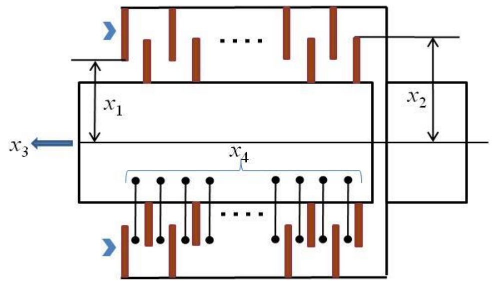

Fig.1. (c) Scheme of disk brake problem

The fourth problem formulated below is a speed reducer design problem for minimizing total volume and stress on the shaft 1 subject to eleven constraints on various stresses, distortion and some design specifications. Among the variables, x_1 – x_3 denote width, module and number of teeth of a toothed gear, respectively and the rests lengths (x_4, x_5) and diameters of shaft (x_6, x_7) as shown in Fig.1 (d). This case sets weights at (0.2, 0.8) and ε-constraint as $f_1(\boldsymbol{x}) \leqq 3000$, respectively.

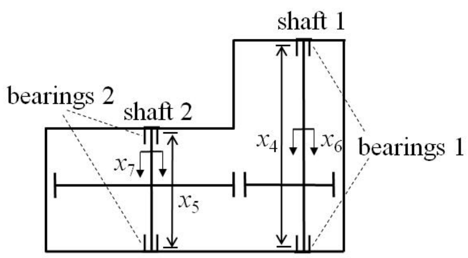

Fig.1. (d) Scheme of speed reducer problem

$$\min \{ f_1(\boldsymbol{x}) = 0.7854\, x_1 x_2^2\, (10 x_3^2/3 + 14.933\, x_3 - 43.0934)$$
$$-1.508\, x_1 (x_6^2 + x_7^2) + 7.447\, (x_6^3 + x_7^3) + 0.7854\, (x_4 x_6^2 + x_5 x_7^2),$$
$$f_2(\boldsymbol{x}) = \sqrt{(745\, x_4/x_2/x_3)^2 + 1.69 \cdot 10^7} / (0.1 x_6^3) \}$$

subject to

$$x_1 x_2^2\, x_3 - 27 \geq 0$$
$$x_1 x_2^2\, x_3^2 - 379.5 \geq 0$$
$$x_2 x_3 x_6^4 - 1.93\, x_4^3 \geq 0$$
$$x_2 x_3 x_7^4 - 1.93\, x_5^3 \geq 0$$
$$40 - x_2 x_3 \geq 0$$
$$12\, x_2 - x_1 \geq 0$$
$$x_1 - 5/x_2 \geq 0$$
$$x_4 - 1.5\, x_6 - 1.9 \geq 0$$
$$x_5 - 1.1 x_7 - 1.9 \geq 0$$
$$1300 - f_2(\boldsymbol{x}) \geq 0$$
$$110 x_7^3 - \sqrt{(745\, x_5/(x_2 x_3))^2 + 1.575 \cdot 10^8} \geq 0$$
$$2.6 \leq x_1 \leq 3.6,\ 0.7 \leq x_2 \leq 0.8, 17 \leq x_3 \leq 28,$$
$$7.3 \leq x_4, x_5 \leq 8.3, 2.9 \leq x_6 \leq 3.9, 5.0 \leq x_7 \leq 5.5$$

Every problem was solved 11 independent runs with different random seeds within several seconds. Then the results were compared with the original NSGA-II and high performance commercial solver LINGO (ver.13.0) in Table 2. We know the enhanced method completely outperforms the usual one and is almost comparable to LINGO that aims at rigid optimization basically. This fact surely reveals the merit of the proposed enhancement due to BPD.

TABLE II. EFFECT OF ENHANCEMENT USING BENCHMARKS

Benchmark		Usual	Enhanced	LINGO
Two-bar truss	*weighting*[d.]	0.142880	0.142450	0.142433
	ε–const.	0.103530	0.099071	0.096154
Welded beam	*weighting*	0.0374149	0.0359751	0.0359182
	ε–const.	0.0164310	0.0163178	0.0159952
Disk brake	*weighting*	1.770252	1.769252	1.749935
	ε–const.	5.058774	5.037595	4.863250
Speed reducer	*weighting*	1155.47	1154.22	1154.22
	ε–const.	697.806	697.781	697.771

[c.] Objective value averaged over 11 trials,

[d.] Upper: weighting problem & Lower: ε-constraint one

2) *Real world car design problem*

Recently, a benchmark problem has been published [12] to facilitate to develop some efficient methods available for real world multi-objective optimization. It is a simultaneous design problem for three types of car structure whose outline is described as follows.

The design objectives are $f_1(\boldsymbol{x})$: maximization of number of common gauge (standard plate thickness) parts among three car types and $f_2(\boldsymbol{x})$: minimization of total weight of the three. The first goal comes from mass-customization to meet a variety of customer demands while the second from reducing cost and environmental burdens. On the other hand, design constraints include crashworthiness in the four crash modes (front 40% offset, full front, side and rear 70% offset impact), torsional stiffness and natural frequency (lateral bending, longitudinal bending and torsion) for each type. Moreover, design variables $(x_i, x_{i+74}, x_{i+148})$, i=1,…,74 are the gauge of parts of the three types.

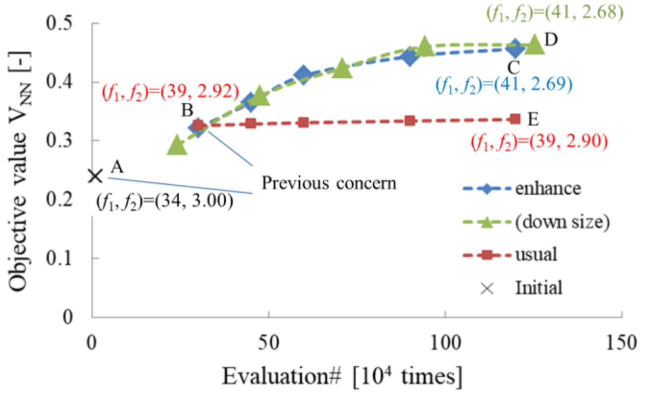

Fig.2. Comparison of profiles along with evaluation time and attaining points

After all, it is a large complicated bi-objective optimization problem composed of 222 decision variables and 54 constraints besides box constraints on the decision variables. Actually, $f_1(\boldsymbol{x})$ is evaluated in terms of the relation such that if $\max\{x_i, x_{i+74}, x_{i+148}\} - \min\{x_i, x_{i+74}, x_{i+148}\} < 0.05$, then part i is viewed as common thickness part and $f_2(\boldsymbol{x})$ is described as multiple regression equation and the constraints are as response surface models using RBF (Radial Basis Function) interpolation.

By the adaptive scalarized method termed MOON[2], this problem was solved successfully thought the procedure mentioned in the paragraph just before [Problem 5]. Then, we revealed the result is much superior to the human empirical decision [3].

Presently, we solved this problem by the proposed enhanced procedure till much longer generation and compared the convergence profiles and final achievements with the previous results. As shown in Fig.2, the profile of the enhanced method (Blue) changes nonlinearly along with evaluation number while the usual one (Red) linearly and slowly. The enhancement also works well for the downsizing version since the profile (Green) is similar to the original one. Eventually, the enhanced method is able to attain at the higher plateaus (Point C, D) than the usual one (Point E). They are better than the human decision (Point A) over 90% in total evaluation (V_{NN}), the previous one (Point B) over 40% and usual one (Point E) over 35%. These results also support the advantage of the proposed enhancement.

C. *Post-optimal Evolution toward Rational Decision*

We showed previously the post-optimal evolution can improve the prior optimization [2]. Besides such small return, we would like to emphasize its great role as the post-optimal analysis that enables us to comprehensively re-consider the prior result before the final decision. As a good example for this purpose, we take a look at the foregoing speed reducer design problem.

Looking at only the preferentially optimal solution derived from the weighting method (left side of Fig.3), we can not completely assure its adequateness due to the indefinite weight setting. By virtue of the post-optimal evolution (middle of Fig.3), we can imagine the shape of

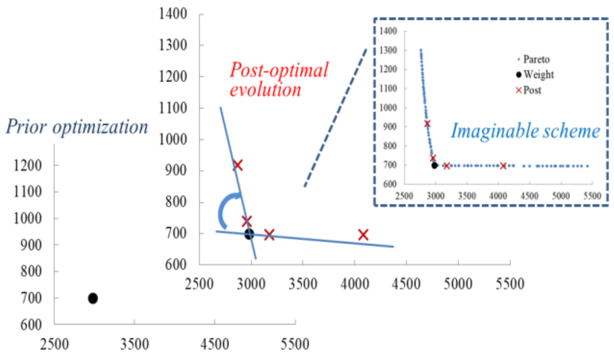

Fig.3. Utility of post-optimal evolution on speed reducer problem: this is obtained under the condition such as *popsz#*=10, *gener#*=500 and 3 elites (two best and one second best solutions in the prior optimization).

Pareto front just around the prior solution (right side of Fig.3). Then, we know the prior solution will stay unchanged over the wide range of weight setting. In other word, this solution is surely appropriate due to the robustness regarding the weight set.

Such local multi-objective analysis, as it were, can repair some shortcomings of the stiff scalarized methods and renew them as more adaptive approaches. Moreover, only small population is enough to perform that. Since these practical but simple approaches are very attractive in real world applications, we claim it can support many engineering optimizations rationally.

V. CONCLUSION

Since MOEAs can reveal a clear trade-off relation among the conflicting objectives effectively, they have been applied conveniently in MOP. To expand such availability and to enhance solution ability much more, this paper has concerned with some deployments for coping with SOP and some scalarized MOP. Actually, the original idea studied in terms of NSGA-II has been compared with other popular MOEAs to find out the more compatible methods. Then, after revealing a special property of individual solution regarding its rank, we have adopted non-Pareto optimal (best performance) dominance in the previous framework.

In numerical experiments, first, we have compared the performance among NSGA-II, MODE and MOPSO through a set of benchmarks. Then, using the best method among them (NSGA-II), the merit of the enhancement due to the new idea has been verified. Actually, multiple mechanical designs problem and the real-world car structure design problem given by the scalarized MOPs have been solved. Moreover, the significance of the post-optimal evolution has been emphasized especially for the scalarized MOPs. By virtue of those demonstrations, we claim the proposed framework can provide a practical procedure for rational decision making through SOP and/or MOP.

ACKNOWLEDGMENT

A part of this research was supported by Grant-in-Aid for Research of JSPS, 17K01250

APPENDIX PROOF OF PROPOSITION 3

We select an arbitrary individual at arbitrary generation and denote it as $\boldsymbol{x}^{p^*}$ and assume its rank is not 1. Then, there exists a certain dominating individual such as $\boldsymbol{x}_d$ at this generation. It means that

$$f_1(\boldsymbol{x}_d) < f_1(\boldsymbol{x}^{p^*}) \qquad \text{(A-1)}$$

$$f_2(\boldsymbol{x}_d) < f_2(\boldsymbol{x}^{p^*}) \qquad \text{(A-2)}$$

Here from the definition, it holds

$$f_2(\boldsymbol{x}_d) = -f_1(\boldsymbol{x}_d) \text{ and } f_2(\boldsymbol{x}^{p^*}) = -f_1(\boldsymbol{x}^{p^*}).$$

Then, we get a relation such that $-f_1(\boldsymbol{x}_d) < -f_1(\boldsymbol{x}^{p^*})$ from Eq.(A-2). This comes to $f_1(\boldsymbol{x}_d) > f_1(\boldsymbol{x}^{p^*})$. This contradicts with Eq.(A-1). Hence, rank is 1 every where and every time.

REFERENCES

[1] J. Bennett and I. Goulter, "The use of multiobjective analysis for comparing and evaluating environmental and economic goals in wetlands managements," GeoJournal, vol. 18, pp. 213-220, 1989.

[2] J. Yoo and Y. Shimizu, "A novel application of multi-objective evolutionary algorithm for practical optimizations through simple formulation and post-optimal evolution," *Proceedings of the Asia Pacific Industrial Engineering & Management Systems Conference*, Hong Kong, 2018.

[3] Y. Shimizu and J. Yoo, "Multi-objective optimization and post-optimal analysis on multiple car structure design toward energy conservation and mass-customization," *Proceedings of the Asia Pacific Industrial Engineering & Management Systems*, Hong Kong, 2018.

[4] Y. Shimizu, and A. Kawada, "Multi-objective optimization in terms of soft computing," Transaction of SICE, vol. 38, pp. 974-980, 2002.

[5] K. Miettinen, "Introduction to multiobjective optimization: noninteractive approaches," in Lecture Notes in Computer Science 5252: Multiobjective Optimization - Interactive and Evolutionary Approaches, Springer-Verlag, Berlin, 2008.

[6] Y. Shimizu, M. Takayama and H. Ohishi, "Multi-objective analysis through elite-induced evolutionary algorithm – in the case of PSA, Transaction of the Japanese Society for Evolutionary Computation, vol. 3, pp. 22-30, 2012.

[7] Y. Wang, H. Liu, Z. Cai and Y. Zhou, "An orthogonal design based constrained evolutionary optimization algorithm, Engineering Optimization," vol. 39, pp. 715-736, 2007

[8] C.A.C. Coello, "Research direction in evolutionary multi-objective optimization - current and future research topics," Transaction of the Japanese Society for Evolutionary Computation, vol. 3, pp. 110-121, 2012.

[9] K. Deb, S. Agrawal, A. Pratap and T. Meyarivan, "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II," *Proceedings of Parallel Problem Solving from Nature VI*, pp. 849-858, 2000.

[10] C. R. Raquel and P.C. Naval, "An effective use of crowding distance in multiobjective particle swarm optimization," *Proceedings of Genetic and Evolutionary Computation Conference (GECCO 2005)*, Washington, D.C., 2005.

[11] W. Gong, Z. Cai and L. Zhu, "An ficient multiobjective differential evolution algorithm for engineering design," Structural and Multidisciplinary Optimization, vol. 38, pp. 137–157, 2009.

[12] T. Kohira, H. Kemmotsu, T. Tatsukawa and A. Oyama," Proposal of benchmark problem based on real-world car structure design optimization," *Proceedings of The Genetic and Evolutionary Computation Conference (GECCO 2018)*, Kyoto, 2018.

[URL-1] LINDO Japan, http://lindo.jp.

[URL-2] MathWorks Japan, http://www.mathworks.es/matlabcentral/fileexchange/authors/289050.

[URL-3] http://www.codeforge.com/read/122123/nsga2.c.