

# A Search Method with User's Preference Direction using Reference Lines

Tomohiro Yoshikawa

Graduate School of Engineering, Nagoya University, Nagoya, Japan, {yoshikawa}@cse.nagoya-u.ac.jp

## Abstract

Recently, a lot of studies on Multi-Objective Genetic Algorithm (MOGA), in which Genetic Algorithm is applied to Multi-objective Optimization Problems (MOPs), have been reported actively. MOGA has been also applied to engineering design fields, then it is important not only to obtain Pareto solutions having high performance but also to analyze the obtained Pareto solutions and extract the knowledge in the designing problem. The another has studied the analysis methods of acquired Pareto solutions by MOGA. The aim of these methods is, however, to analyze solutions, and the feedback of the analysis results into the search is little focused. This paper proposes a search method that uses reference lines for a user's preference direction which is defined by the user's attention based on visualization results of a pre-search.

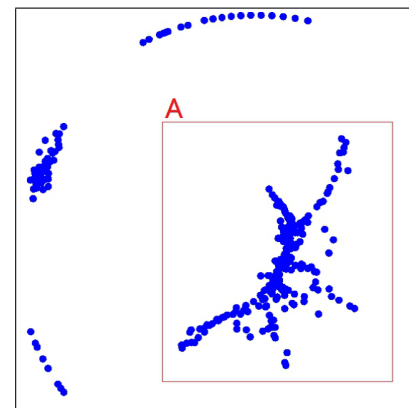
*Keywords: optimization problems, multi-objective genetic algorithm, user's preference direction, reference Line, visualization*

## 1 Introduction

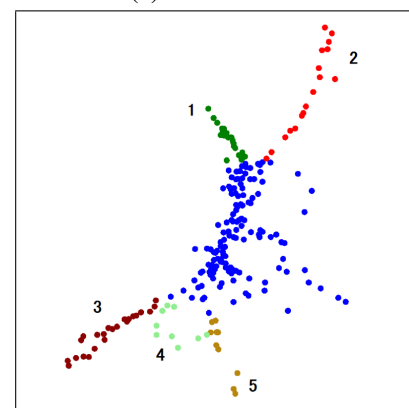
In recent years, it is reported that Multi-Objective Genetic Algorithm (MOGA) (Deb, 2001) is applied to engineering design problems in the real-world due to the improvement of computing performance (Obayashi, 2003; Deb, 2003; Oyama and Kawakatsu, 2010). In the engineering design problems, it is required not only to obtain high performance Pareto solutions using MOGA but also to analyze and extract design knowledge in the problem. And in order to analyze Pareto solutions obtained by MOGA, it is required to consider both the objective space and the design variable space.

The author has proposed some analysis methods of acquired solutions by evolutionary computation based on "visualization" (Yamashiro et al., 2006; Ishiguro et al., 2008; Yamamoto et al., 2010; Kudo and Yoshikawa, 2012). Figure 1 and Figure 2 are the examples of visualization for the analysis of Pareto solutions. However, the goal of these approaches is to analyze the acquired solutions. Designers often need better solutions than the acquired ones, more solutions that have the desired features provided by the analysis, or better fitness values on a certain objective function that keeps the other fitness values.

This paper proposes a search method for a user's



(a) Overall View.

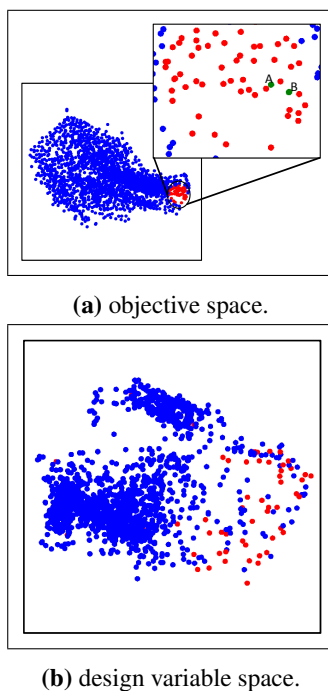


(b) Grouping in each Branch.

**Figure 1.** Visualization Result using Visualization Tool "AD-VICE" in Conceptual Design Optimization Problem of HRE (Hybrid Rocket Engine).

preference direction based on "reference lines" which is one of the mechanisms in NSGA-III proposed in (Deb and Jain, 2014). In the proposed method, a user selects the preference area in the visualized space by plotting the acquired solutions, and reference points are generated in the selected area. Reference lines are defined by making connections between the reference points and the original point. Moreover, in the proposed method, a user can move the original point based on his/her desired feature of solutions. This paper includes the results off an experiment that applies the proposed method to a real coded multi-objective knapsack problem (Hirano and Yoshikawa, 2013) and examines the ef-

fectiveness of the proposed method.



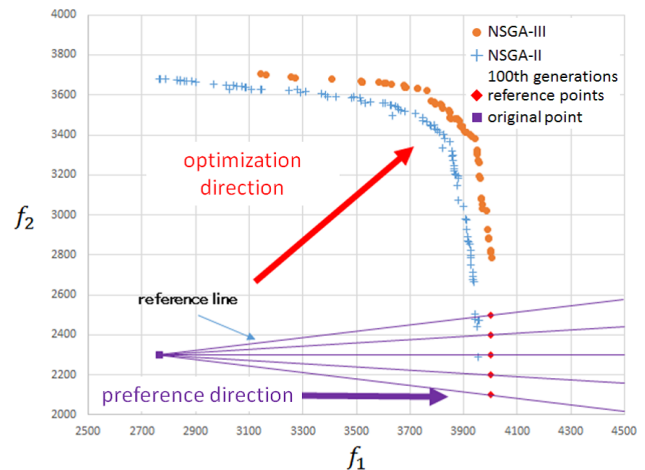
**Figure 2.** Distribution of Pareto Solutions for Non-Correspondence Area in Trajectory Designing Optimization Problem of “DESTINY”.

## 2 Proposed Method

The proposed method employs the concept of reference lines used in NSGA-III (Deb and Jain, 2014). The aim of the proposed method is to search user’s preference direction based on the analysis result generated by the visualization of acquired Pareto solutions. In the proposed method, a user selects the preference area in the visualized space by plotting the acquired solutions, and reference points are generated in the selected area. Reference lines are defined by making connections between the reference points and the original point. Moreover, in the proposed method, a user can move the original point based on his/her desired feature of solutions. As with NSGA-III, the basic algorithm of the proposed method is NSGA-II (Deb, 2002) with the added concept of reference lines. The detail of the operations and the features in the proposed method are described below.

### 2.1 Selection of Reference Points

In the proposed method, after the analysis generated by the visualization of Pareto solutions acquired by an arbitrary pre-search, a user selects the interested area or individuals based on the analysis results, and then reference points are generated. The search around the selected individuals or highly converged solutions to the user’s preference direction can be done by using reference lines connected between the original point shown in section 2.4 and the reference points. There are following two ways to select



**Figure 3.** Search by NSGA-III.

reference points.

- A user selects any individuals directly based on the information of the visualization result of objective space or the fitness values of individuals. The fitness values of these selected solutions are defined as the reference points.
- A user selects interested areas based on the information of the visualization result of objective space.  $N_r$  individuals are randomly selected from the selected area and are defined as the reference points.

### 2.2 Selection of Individuals for Next Generation

NSGA-II and NSGA-III are based on non-dominated sorting. Thus, these methods cannot search well for a user’s preference direction when the user’s preference direction is far from the center of the optimization direction. Figure 3 shows the acquired Pareto solutions generated by NSGA-III with 100 individuals and 100 generations after 100 generations of NSGA-II, in a two-objective knapsack problem.

In the proposed method, individuals close to the reference lines are selected preferentially for the next generation. It is expected that the number of neighborhood individuals for every reference line becomes same by this section according to the progress of generations. The algorithm for the selection of the next generation is described below.

Step 1: Define the nearest reference line for each individual as a neighborhood line and the individuals belonging to each reference line (neighborhood line) as neighborhood individuals.

Step 2: Select one reference line randomly.

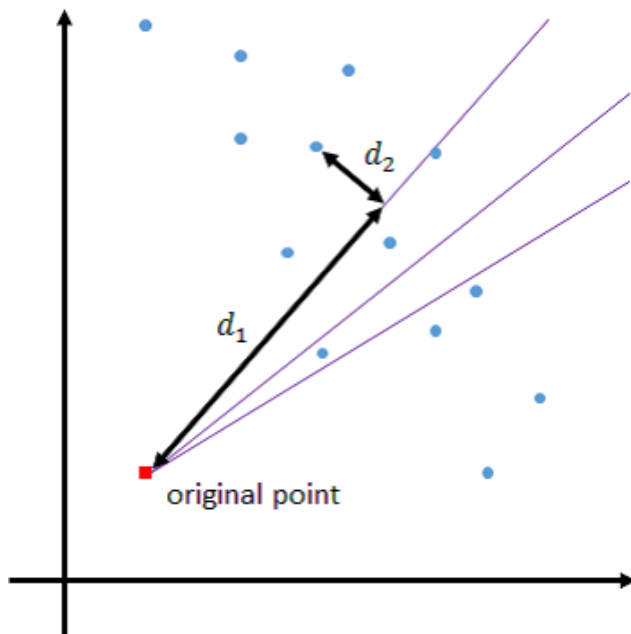


Figure 4. PBI distance.

Step 3: Select the individual that has the largest penalty-based boundary intersection (PBI) distance (Zhang and Li, 2007; Sato, 2014), which is calculated by 1 (see Figure 4), in all neighborhood individuals of the selected reference line for the next generation and remove it from the neighborhood individuals. The PBI distance is used for scalarized optimization, and  $\theta$  is a penalty parameter. When  $\theta$  is large, the individuals near the weight vector are given priority. When there is no neighborhood individual at the selected line, select the nearest individual to the line from all remaining individuals for the next generation and remove it.

Step 4: Select other reference lines randomly from the not-selected lines and return to Step 3. If all reference lines are already selected, clear the selected information and return to Step 2.

Step 5: Repeat Step 2 to Step 4 until the number of next generation individuals becomes the population size.

$$\text{PBI distance} = d_1 - \theta d_2 \quad (1)$$

### 2.3 Selection of Parents for Crossover

In the selection of parent individuals for crossover, tournament selection chooses one parent individual from neighborhood individuals for a reference line. The selected parent individual crossovers with another selected parent individual. The algorithm for the parent selection for crossover is described below.

Step 1: As described in section 2.2, define neighborhood lines and neighborhood individuals.

Step 2: Select one reference line randomly.

Step 3: Depending on the number of neighborhood individuals of the selected reference line, carry out the following selection.

- If the number of neighborhood individuals is more than 1.

The tournament selection based on PBI distance is performed, *i.e.*, two individuals are randomly chosen and the one that has the larger PBI distance is selected as the parent individual.

- If the number of neighborhood individuals is 1.

The individual becomes the parent individual.

- If the number of neighborhood individuals is 0.

The individual closest to the reference line in all individuals becomes the parent individual.

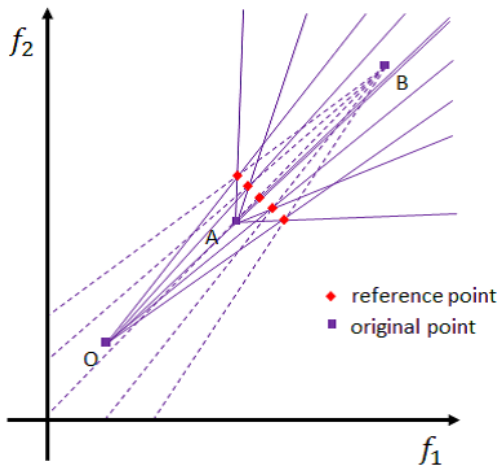
Step 4: Select another parent individual by carrying out Step 2 and Step 3. If the same individual is selected, return to Step 2 until a different one is selected.

### 2.4 Moving Original Point

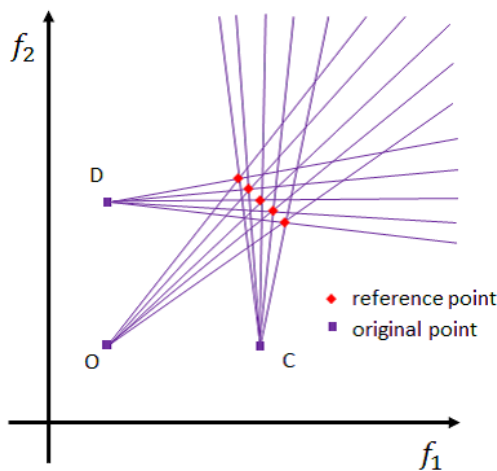
Basically, the original point is set to the worst value of each fitness value in all solutions. However, the proposed method allows a user to move the original point based on his/her desired feature of solutions and change the feature of solutions obtained by the additional search.

Figure 5 shows examples of the original point moving. Figure 5 is an example of maximizing objective functions  $f_1$  and  $f_2$ . The original point  $O$  is the standard original point described above. Point  $A$  in Figure 5(a) is the worst fitness value in the reference points. For example, moving the original point to point  $A$  widens the reference lines and it is possible to acquire solutions with large diversity in the user's preference area. Conversely, moving the original point far away from the reference points or to the opposite side of the reference points, such as point  $B$  in Figure 5(a) makes the reference lines narrow and the convergence of search solutions is expected to be high or they converge to one point (point  $B$ ).

Moreover, most fitness values are often satisfied for a user, but the fitness values of specific objective functions are not satisfied in real-world problems. In such cases, by moving the original point such as point  $C$  or  $D$  in Figure 5(b), the search direction can be concentrated to the user's preference direction. For example, when the original point is set at point  $D$ ,  $f_1$  will be maximized by keeping  $f_2$ .



(a) Adjustment of Width of Searching Individuals.



(b) Adjustment of Weight Objective Functions.

Figure 5. Moving of Original Point.

Thus a user can search preference direction, angle and width easily by moving the original point. For example, if a user wants various individuals, he/she should set the original point at the worst fitness value in the reference points. On the other hand, if a user wants converged individuals, he/she should set the original point at the almost infinite distant point. And if a user wants to keep a certain objective functions and converge others, he/she should move the original point to the center of gravity of reference points in objective functions which he/she wants to keep.

### 3 Experiment

#### 3.1 Experiment Condition

The experiment described in this paper applied the proposed method to a real coded multi-objective knapsack problem (Hirano and Yoshikawa, 2013) and studied the effectiveness of the proposed method. The number of items was 100. The cost and value of each item were given as random integer values between 0 and 99, and the upper

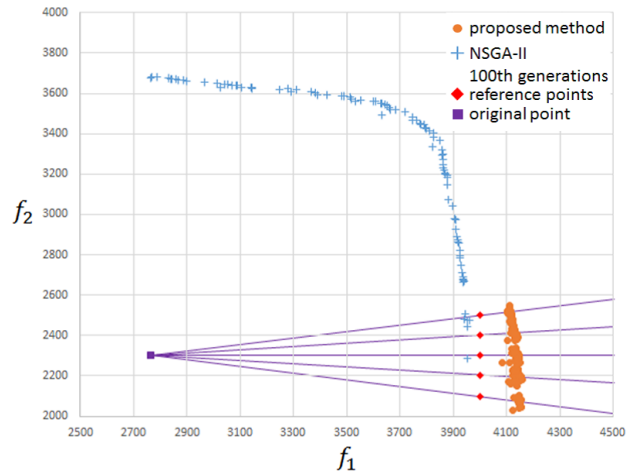


Figure 6. Search by the Proposed Method.

limit of cost was half of the total cost for all items. When the sum of the cost exceeded the upper limit, the exceeded value  $\times 5$  was subtracted from the fitness values. Though we do not have to normalize the fitness values in this problem, we should normalize each fitness value when their value scales are different from each other.

The proposed method assumes that there is a pre-search and that analysis is performed. In this experiment, the pre-search was conducted by NSGA-II. Its population size was 100 and the number of generations was 100. I selected reference points from individuals obtained by pre-search, and searched an additional 100 generations by the proposed method. Then further individuals were gained by pre-search, and added to the initial individuals.

#### 3.2 Effect of Search for Preference Direction

Figure 6 shows the search result by the proposed method using the same conditions as in Figure 3. We can see well-converged solutions for the preference direction which are not present in Figure 3. Although the reference points are usually selected from the acquired solutions in the proposed method, they use the same points as in Figure 3 for the comparison.

#### 3.3 Effect of Moving Original Point

We examined the effect of moving the original point, as described in 2.4. Figure 7 shows the individuals obtained by the proposed method using the original point (a) the worst value of each fitness value ( $f_1, f_2$ ) in all individuals including all generations (point O in Figure 5), (b) the worst value of each fitness value in the reference points (point A in Figure 5), (c) infinite distance from the reference points ( $(f_1, f_2) = (-100, -100)$ ). To evaluate the effectiveness of the proposed method, I use the metric as the performance index. Because the conventional performance indexes are not appropriate to evaluate the performance that this paper desires. Table 1 shows the average distance from the reference point of the neighbor-

**Table 1.** Convergence and Diversity of Pareto Solutions (2 objectives)

	distance from RP	S.D.	width	S.D.
(a)	136.8	12.8	303.8	46.8
(b)	106.2	11.1	447.8	25.0
(c)	140.8	14.2	276.9	25.0

**Table 2.** Difference between Center of Gravity of Pareto Solutions and that of Reference Points (2 objectives)

$f_1$	$f_2$
168.2	16.7

hood line, which corresponds to the convergence, and the width obtained Pareto solutions, which corresponds to the diversity. Each value in the table is the average of 30 trials and the standard deviation.

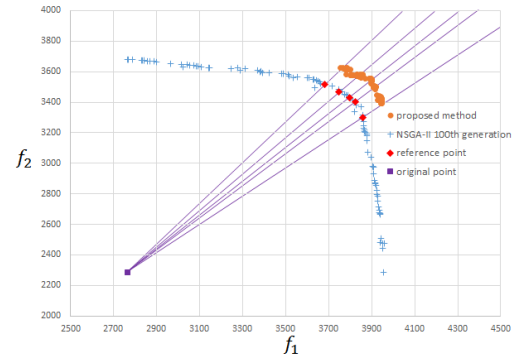
Figure 7 and Table 1 show that the proposed method searched widely by moving the original point close to the reference points, as in Figure 7(b). On the other hand, the proposed method searched narrowly by moving the original point far from the reference points like Figure 7(c), and that made the convergence a little high.

Figure 8 shows the obtained individuals when the position of the original point was set to emphasize  $f_1$  (point D in Figure 7). Moreover, Table 2 shows the difference between the center of gravity of Pareto solutions and that of reference points for each fitness value. Just as intended, the value of  $f_1$  was much improved while that of  $f_2$  remained unchanged.

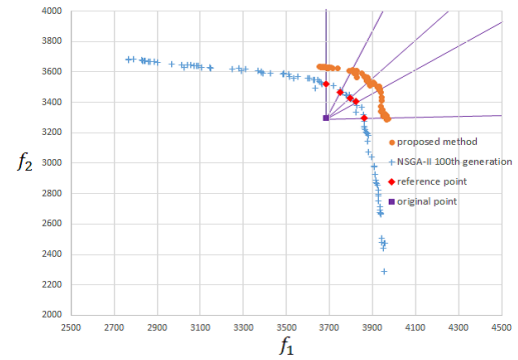
### 3.4 Effect in Many-objective Optimization Programs

Here, I examine the effect of moving the original point shown in 3.3 in the case of MaOPs. Table 3 shows the same performance indexes (average of 30 trials) with Table 1 in the 2, 4, 6, 8, and 10 objective knapsack problem, respectively. In the real world problems, too many number of objective functions is not practical. So I set the number of objective functions up to 10. This result shows the same tendency as that of Table 1 by moving the original point, in which I can adjust the spread of individuals, even in MaOPs. In MaOPs, the convergence of Pareto solutions acquired by NSGA-II is low; however, the acquired solutions by the proposed method were well-converged and the distance from RP became much larger as the number of objectives increased by concentrating the search area and direction. The selected area and RPs were the same in all trials, and the selected area in 8 objectives might be difficult to search. That could be the reason why the values for 8 objectives were comparatively lower those for 6 objectives.

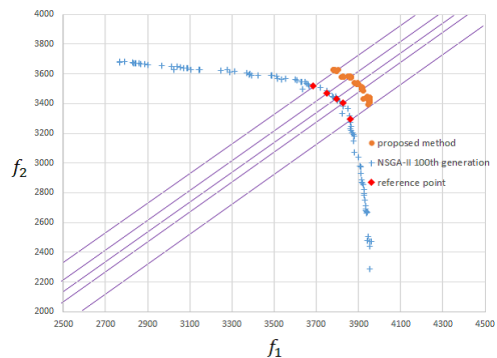
Table 4 shows the result of moving the original point as in Figure 8, in order to search  $f_1$  with priority in 2 ~ 10



(a) Worst Fitness Values in All Individuals.



(b) Worst Fitness Values in Reference Points.



(c) Infinite Distance(-100,000, -100,000).

**Figure 7.** Effect of Moving Original Point.

objective problems. The values of Table 4 are the same as those of Table 2. The performance of  $f_1$  is much higher than that of any other objective functions, for all numbers of objectives. This means that moving the original point makes a desired functional search possible in MaOPs as well as 2 objectives.

## 4 Conclusions

In this paper, I proposed a search method for a user's preference direction by using the concept of reference lines. In the proposed method, a user selects the preference area in the visualized space by plotting the acquired solutions, and reference points are generated in the selected area. Reference lines are defined by making connections between the reference points and the original point. In this

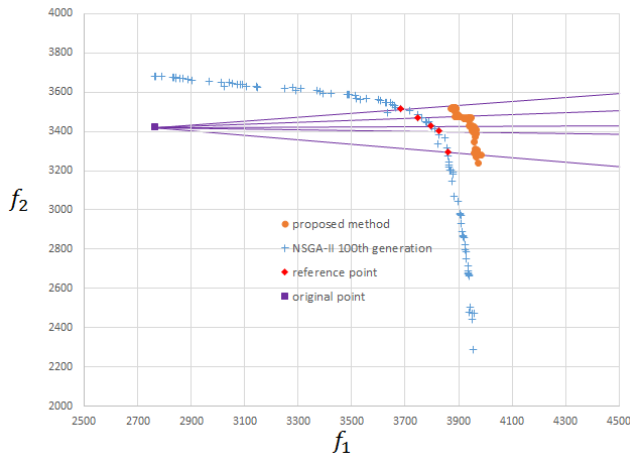


Figure 8. Search with Priority of  $f_1$ .

Table 3. Convergence and Diversity of Pareto Solutions (2~10 objectives)

	objectives	2	4	6	8	10
(a)	distance from RP	136.8	575.6	1025.5	682.4	1613.3
	S.D.	12.8	27.9	48.5	40.3	38.9
	width	303.8	1097.9	2614.0	2640.3	3050.9
	S.D.	46.8	73.1	179.2	180.3	237.6
(b)	distance from RP	106.2	516.7	865.1	483.7	533.0
	S.D.	11.1	27.6	42.2	51.1	51.3
	width	447.8	1871.2	3924.2	4152.1	5045.8
	S.D.	25.0	104.2	179.9	241.5	234.7
(c)	distance from RP	140.8	640.2	1369.4	756.1	1950.2
	S.D.	14.2	25.9	52.0	54.4	42.7
	width	276.9	979.5	1924.4	2327.1	2469.1
	S.D.	25.0	96.2	124.7	162.7	213.8

paper, I described an experiment that applied the proposed method to a real coded multi-objective knapsack problem and studied the effectiveness of the proposed method. The experimental result showed that the solutions having the desired features could be acquired by moving the original point. The experimental result showed that the proposed method also worked well in MaOPs, as well as for 2 objectives. The effect of the proposed method increased with the number of objectives. Future work includes the investigation of the number of reference points and how to define them from the user's preference area. The application and investigation of the proposed method into real-world problems will be also done.

## Acknowledgment

This work was supported by the Grant-in-Aid for Scientific Research (C) from the Ministry of Education, Culture, Sports, Science and Technology (MEXT), Japan, Grant number: 15K00336.

## References

K. Deb. *Multi-objective optimization using evolutionary algorithms*. Wiley, 2001. ISBN 047187339X.

Table 4. Difference between Center of Gravity of Pareto Solutions and that of Reference Points (2~10 objectives)

objectives	2	4	6	8	10
$f_1$	168.2	662.8	531.4	822	297.2
$f_2$	16.7	-14.4	99	135.3	-57
$f_3$	-	-11.5	31.7	45.7	15
$f_4$	-	42.2	51.9	-34.6	70.8
$f_5$	-	-	-52.9	-47.9	12.7
$f_6$	-	-	49.4	26.4	-28.6
$f_7$	-	-	-	41.4	79.3
$f_8$	-	-	-	22.8	-71.5
$f_9$	-	-	-	-	70.9
$f_{10}$	-	-	-	-	-33.2

K. Deb. A Fast and Elitist Multiobjective Genetic Algorithm : NSGA-II. *Evolutionary Multi-Criterion Optimization*, pages 182–197, 2002.

K. Deb. Unveiling innovative design principles by means of multiple conflicting objectives. *Engineering Optimization*, 35(5): 445–470, 2003.

K. Deb and H. Jain. An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach, part I: Solving problems with box constraints. *IEEE Transactions on Evolutionary Computation (TEVC)*, 18(4):577–601, 2014.

Hiroyuki Hirano and Tomohiro Yoshikawa. A study on two-step search based on pso to improve convergence and diversity for many-objective optimization problems. *2013 IEEE Congress on Evolutionary Computation (CEC)*, pages 1854–1859, 2013.

Hidetaka Ishiguro, Tomohiro Yoshikawa, and Takeshi Furuhashi. Visualization of gene-evaluation value in multi-objective problem and feedback for efficient search. *SCIS & ISIS 2008*, pages 1667–1670, 2008.

Fumiya Kudo and Tomohiro Yoshikawa. Knowledge extraction in multi-objective optimization problem based on visualization of Pareto solutions. *2012 IEEE Congress on Evolutionary Computation (CEC)*, pages 860–865, 2012.

S. Obayashi. Multi objective design optimization of aircraft configuration (in Japanese). *The Japanese Society for Artificial Intelligence*, 18(5):495–501, 2003.

K.H. Akira Oyama and Yasuhiro Kawakatsu. Application of multiobjective design exploration to trajectory design of the next-generation solar physics satellite. *Japanese Society for Evolutionary Computation*, 2010.

Hiroyuki Sato. Inverted pbi in moea/d and its impact on the search performance on multi and many-objective optimization. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*, pages 645–652. ACM, 2014.

Masafumi Yamamoto, Tomohiro Yoshikawa, and Takeshi Furuhashi. Study on effect of MOGA with interactive island model using visualization. *2010 IEEE Congress on Evolutionary Computation (CEC)*, pages 4196–4201, 2010.

Daisuke Yamashiro, Tomohiro Yoshikawa, and Takeshi Furuhashi. Visualization of search process and improvement of search performance in multi-objective genetic algorithm. *2006 IEEE Congress on Evolutionary Computation (CEC)*, pages 1151–1156, 2006.

Q. Zhang and H. Li. Moea/d: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation (TEVC)*, 11(6):712–731, 2007.