H. Ishibuchi, Y. Sakane, N. Tsukamoto, and Y. Nojima, "Selecting a small number of representative non-dominated solutions by a hypervolume-based solution selection approach," Proc. of 2009 IEEE International Conference on Fuzzy Systems, pp. 1609-1614, Jeju Island, Korea, August 20-24, 2009.

Selecting a Small Number of Representative Non-Dominated Solutions by a Hypervolume-Based Solution Selection Approach

Hisao Ishibuchi, *Member, IEEE*, Yuji Sakane, Noritaka Tsukamoto, and Yusuke Nojima, *Member, IEEE*

***Abstract*—A large number of non-dominated solutions are often obtained by a single run of an evolutionary multiobjective optimization (EMO) algorithm. In the EMO research area, it is usually assumed that a single solution is to be chosen from the obtained non-dominated solutions by the decision maker. It is, however, time-consuming and not easy for the decision maker to examine a large number of obtained non-dominated solutions. Motivated by these discussions, we proposed single-objective and multiobjective formulations of solution selection problems to present only a small number of representative non-dominated solutions to the decision maker in our former study. The basic idea is to minimize the number of solutions to be presented while maximizing their hypervolume. A number of single-objective formulations can be derived from such a two-objective solution selection problem. In this paper, single-objective rule selection is performed as a post-processing procedure of EMO algorithms to select a prespecified number of non-dominated solutions (e.g., 10 or 20 solutions). Through computational experiments on multiobjective 0/1 knapsack problems, we examine the characteristic features of selected non-dominated solutions. We also examine the effect of the choice of a reference point for hypervolume calculation on the distribution of selected non-dominated solutions.**

I. Introduction

Evolutionary multiobjective optimization (EMO) has been an active research area in the field of evolutionary computation [1]-[5]. A number of EMO algorithms have been proposed and successfully applied to various optimization problems. Well-known and frequently-used EMO algorithms such as NSGA-II [6] and SPEA [7] can be characterized by the following three features: Pareto dominance-based fitness evaluation, diversity maintenance mechanisms, and elitism. These EMO algorithms are called Pareto dominance-based algorithms because the Pareto dominance relation is used as the main criterion for fitness evaluation. Whereas Pareto dominance-based algorithms usually work very well on multiobjective problems with two or three objectives, their search ability is severely degraded by the increase in the number of objectives [8], [9]. This is because almost all individuals in a population become non-dominated with each other under the presence of many objectives. In this case, the Pareto dominance relation can not generate a sufficient selection pressure toward the Pareto front. As a result, only the diversity of solutions is improved without converging toward the Pareto front.

A new trend in the EMO research area is the handling of a multiobjective problem as an optimization problem of an indicator function. Various algorithms [10]-[14], which are often called indicator-based evolutionary algorithms (IBEAs), have been proposed in the literature. The basic idea of IBEAs was first used as a solution selection criterion for choosing a prespecified number of non-dominated solutions to be stored in an archive population [15]. The hypervolume measure [16] has been frequently used as an indicator in IBEAs. One advantage of IBEAs over Pareto dominance-based algorithms is their potential ability to handle many-objective problems. It was demonstrated in [14] that IBEAs outperformed Pareto dominance-based algorithms for many-objective problems with up to six objectives. Another advantage of IBEAs is that user preference can be easily incorporated into an indicator function to drive multiobjective search toward a specific area of the Pareto front [17], [18].

An indicator function such as the hypervolume measure in IBEAs is used to evaluate solution sets instead of solutions. Thus IBEAs can be viewed as single-objective optimizers for finding optimal solution sets [19]-[21]. In this paper, we use the hypervolume measure as an indicator function to evaluate a subset of obtained non-dominated solutions. We select only a small number of representative solutions from a large number of obtained non-dominated solutions. Only the selected solutions are presented to the decision maker as representative candidate solutions. The decision maker is supposed to choose a single solution from the presented solutions. The aim of solution selection in this paper is to relieve the burden of the decision maker by decreasing the number of solutions to be presented. One may think that the most preferred solution can be removed by solution selection. In this case, the chosen solution is not the best for the decision maker among the obtained non-dominated solutions whereas it is the best among the presented candidate solutions. This drawback is somewhat alleviated by performing an additional selection phase around the chosen solution. That is, a number of non-dominated solutions around the chosen one are presented to the decision maker. The decision maker can examine newly presented solutions. If these exists a better solution, the decision maker can update his/her chosen solution. Candidate solutions in the second round of solution selection can be found by the Euclidean distance from the

Manuscript received March 1, 2009. This work was supported in part by the Japan Society for the Promotion of Science under Grant-in-Aid for Scientific Research for Scientific Research (B) (20300084).

Hisao Ishibuchi, Yuji Sakane, Noritaka Tsukamoto, and Yusuke Nojima are with Department of Computer Science and Intelligent Systems, Graduate School of Engineering, Osaka Prefecture University, 1-1 Gakuen-cho, Naka-ku, Sakai, Osaka 599-8531, Japan (hisaoi@cs.osakafu-u.ac.jp, sakane@ci.cs.osakafu-u.ac.jp, nori@ci.cs.osakafu-u.ac.jp, nojima@cs.osakafu-u.ac.jp).

currently chosen solution. They can be also selected by our hypervolume-based solution selection approach using a reference solution close to the currently chosen solution.

This paper is organized as follows. In Section II, we explain how hypervolume maximization can be formulated as single-objective and multiobjective optimization problems [21]. In Section III, we explain how we can utilize the formulated problems to select a small number of uniformly distributed non-dominated solutions. That is, we explain how single-objective and multiobjective genetic algorithms are implemented for solution selection. We assume that a large number of non-dominated solutions have already been given by EMO algorithms. In Section IV, we demonstrate the usefulness of the proposed approach through computational experiments on 500-item knapsack problems with two and three objectives in [7]. We also examine the relation between the location of a reference point and selected non-dominated solutions for hypervolume maximization. We conclude this paper in Section V.

II. Formulations of Rule Selection

Let us consider the k-objective maximization problem:

$$\text{Maximize } \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_k(\mathbf{x})), \qquad (1)$$

$$\text{subject to } \mathbf{x} \in \mathbf{X}, \qquad (2)$$

where $\mathbf{f}(\mathbf{x})$ is the k-dimensional objective vector, $f_i(\mathbf{x})$ is the i-th objective to be maximized ($i = 1, 2, ..., k$), $\mathbf{x}$ is the decision vector, and $\mathbf{X}$ is the feasible region in the decision space.

In the EMO community, the following multiobjective decision making process is usually assumed [1]: First an EMO algorithm is used to search for non-dominated solutions that well approximate the entire Pareto front of the given multiobjective problem. Then the obtained non-dominated solutions are presented to the decision maker as candidate solutions. Finally the decision maker is supposed to choose a single solution from the presented non-dominated solutions.

We denote a set of solutions by S. In general, S can be an arbitrary subset of the feasible region $\mathbf{X}$. In the context of solution selection, S is a subset of the given non-dominated solutions. The solution set S is optimized by an IBEA. Let S^* be the optimal or near-optimal solution set obtained after the execution of an IBEA. A singe solution is to be chosen from S^* by the decision maker according to his/her preference. Let $I(S)$ be an indicator function which measures the quality of the solution set S. In this paper, we use the hypervolume measure as $I(S)$. IBEAs with a fixed population size N_{pop} can be viewed as optimizers of the following problem:

[Single-Objective Formulation with an Equality Condition]

$$\text{Maximize } I(S), \qquad (3)$$

$$\text{subject to } S \subset \mathbf{X} \text{ and } |S| = N_{\text{pop}}, \qquad (4)$$

where $|S|$ is the number of solutions in S (i.e., the cardinality of S). This formulation corresponds to the situation where the decision maker wants to see N_{pop} non-dominated solutions. When N_{pop} is very small (e.g., say $N_{\text{pop}} = 5$), it is difficult for IBEAs (with no tailored special tricks such as dynamic adjustment of the population size) to find a good solution set since the population size is very small.

When the decision maker does not want to see more than N^* non-dominated solutions, the value of N^* should be handled as the upper limit on the number of solutions in the solution set S. In this case, the single-objective formulation in (3)-(4) is modified as follows:

[Single-Objective Formulation with an Inequality Condition]

$$\text{Maximize } I(S), \qquad (5)$$

$$\text{subject to } S \subset \mathbf{X} \text{ and } |S| \leq N^*. \qquad (6)$$

When the decision maker wants to minimize the number of presented non-dominated solutions, one possible formulation is the use of the following weighted sum objective function:

[Weighed Sum-based Single-Objective Formulation]

$$\text{Maximize } w_1 \cdot I(S) - w_2 \cdot |S|, \qquad (7)$$

$$\text{subject to } S \subset \mathbf{X}, \qquad (8)$$

where $\mathbf{w} = (w_1, w_2)$ is a non-negative weight vector. One drawback of this formulation is the difficulty in specifying an appropriate weight vector. Another drawback is that no Pareto optimal solutions can be found in non-convex regions of the Pareto front independent of the specification of the weight vector.

In general, we can use any scalarizing function to combine the two terms in (7): $I(S)$ and $|S|$. Let us denote a scalarizing function as $g(\cdot)$. We assume that larger values of $g(\cdot)$ mean better solution sets. A scalarizing function-based formulation can be written as follows:

[Scalarizing function-based Single-Objective Formulation]

$$\text{Maximize } g(I(S), |S|), \qquad (9)$$

$$\text{subject to } S \subset \mathbf{X}. \qquad (10)$$

The solution set S* obtained from this formulation depends on the choice of a scalarizing function and the specifications of related parameters (e.g., the two weight values in the case of the weighted sum scalarizing function in (7)). It is, however, not easy to appropriately formulate a scalarizing function in (9) using available information on the decision maker's preference.

The two terms in (9) can be also handled separately as individual objectives as follows:

[Multiobjective Formulation]

$$\text{Maximize } I(S) \text{ and minimize } |S|, \qquad (11)$$

$$\text{subject to } S \subset \mathbf{X}. \qquad (12)$$

In standard IBEAs (except for some special versions such as its implementation with multiple populations [23]), the entire population is the solution set S since a single solution $\mathbf{x}$

of the original multiobjective problem in (1)-(2) is coded as an individual. As a result, IBEAs can not search for a large number of non-dominated solution sets of the multiobjective problem in (11)-(12). On the other hand, our multiobjective formulation in (11)-(12) usually has many non-dominated solution sets.

III. Implementation

It is possible to implement general IBEAs by coding a solution set S using a concatenated string "$\mathbf{x}_1 \mathbf{x}_2 \ldots \mathbf{x}_m$" of variable string length. In this coding, $\mathbf{x}_i$ $(i = 1, 2, \ldots, m)$ is a solution of the original multiobjective problem in (1)-(2) and m is the number of solutions included in the solution set S. It should be noted that m is not fixed since we assume variable string length. IBEAs with such a concatenated string of variable string length may need a large computation load and/or a number of sophisticated speed-up tricks to efficiently search for good solution sets. This is because the search space is huge. So we leave the implementation of general IBEAs and their performance evaluation as future research issues. In this paper, we implement a simple IBEA for a very special case where a large number of non-dominated solutions have already been given. This is because we focus on solution selection as a post-processing procedure of EMO algorithms.

Let us assume that we have N non-dominated solutions ($\mathbf{x}_1^*, \mathbf{x}_2^*, \ldots, \mathbf{x}_N^*$) of our original multiobjective problem in (1)-(2). In this case, any subset S of the N non-dominated solutions can be represented by a binary string of length N:

$$S = s_1 s_2 \ldots s_N , \tag{13}$$

where $s_j = 1$ and $s_j = 0$ mean the inclusion of the j-th solution $\mathbf{x}_j^*$ in the solution set S and its exclusion from S, respectively. Thus the binary string $s_1 s_2 \ldots s_N$ is decoded as the following solution set:

$$S = \{\mathbf{x}_i^* \mid s_i = 1,\ i = 1, 2, \ldots, N\} . \tag{14}$$

Since binary strings of the fixed string length in (13) are used as individuals, standard evolutionary algorithms can be directly applied to our single-objective and multiobjective formulations of solution selection in Section II. In our computational experiments, we applied a steady-state genetic algorithm with the $(\mu + 1)$-ES generation update mechanism to the second single-objective formulation with the upper limit on the number of solutions in (5) and (6).

We handled the constraint condition in (6) by the penalty function method. That is, the single-objective problem with the constraint condition in (5) and (6) was handled as the maximization problem of the following penalty function with no constraint condition:

$$Fitness(S) = I(S) - \alpha \cdot \max\{0, |S| - N^*\} , \tag{15}$$

where α is a penalty constant. We specified α as $\alpha = 500{,}000$ for a two-objective knapsack problem and $\alpha = (500{,}000)^2$ for a three-objective knapsack problem in our computational experiments. We used such a large penalty constant because the magnitude of the first term (i.e., the hypervolume measure) is much larger than that of the second term (i.e., the amount of the constraint violation).

Since there is a tradeoff relation between the hypervolume and the number of non-dominated solutions, the equality constraint in (4) can be also handled by the fitness function with the penalty term in (15). In our steady-state genetic algorithms, we used binary tournament selection, uniform crossover and bit-flip mutation.

IV. Computational Experiments

A. Test Problems

As test problems, we used 500-item knapsack problems with two and three objectives in [7]. We denote these test problems as 2-500 and 3-500, respectively. First we searched for a large number of non-dominated solutions of each test problem. For this task, we applied a cellular version [24] of MOEA/D (multiobjective evolutionary algorithm based on decomposition [25]) to each test problem. Then our solution selection approach was used as a post processing procedure for the cellular MOEA/D.

MOEA/D is a powerful EMO algorithm based on a scalarizing function with a number of uniformly distributed weight vectors. The original MOEA/D [25] has an archive population to store non-dominated solutions. Since archive maintenance is time-consuming when its size is very large, we used its cellular version with no archive population in our computational experiments in order to efficiently find a large number of non-dominated solutions.

In our computational experiments, we used the following parameter specifications in the cellular MOEA/D:

Population size: 10000 for the 2-500 problem,
10011* for the 3-500 problem,
(* Due to the uniform weight vector specification [23]),
Scalarizing function: Tchebycheff function
(The same implementation as in [23]),
Selection neighborhood: 2% of the population size,
Mutation probability: 1/500 (Bit-flip mutation),
Crossover probability: 0.8 (Uniform crossover),
Replacement neighborhood: 20% of the population size,
Constraint handling: Greedy repair method used in [7],
Termination condition (i.e., Number of solution evaluations):
100,000 (2-500) and 125,000 (3-500).

In the cellular MOEA/D, the population size is the same as the number of cells. Because each cell has a single individual and a single weight vector, the population size is also the same as the number of weight vectors. The spatial structure of cells is specified by uniformly distributed weight vectors. For example, the distance between cells is defined by the distance between their weight vectors. Two parents of an offspring for a cell are selected from its selection neighborhood. The newly generated offspring can replace current individuals in its replacement neighborhood (for details, see [24]-[26]).

From a single run of the cellular MOEA/D for each test problem, we obtained 212 non-dominated solutions of the 2-500 problem (see Fig. 1) and 1444 non-dominated solutions of the 3-500 problem (see Fig. 2). These non-dominated solutions for each test problems were used as candidate solutions in our solution selection approach. Our task in solution selection is to select a pre-specified number of non-dominated solutions from the candidate solutions in Fig. 1 and Fig. 2.

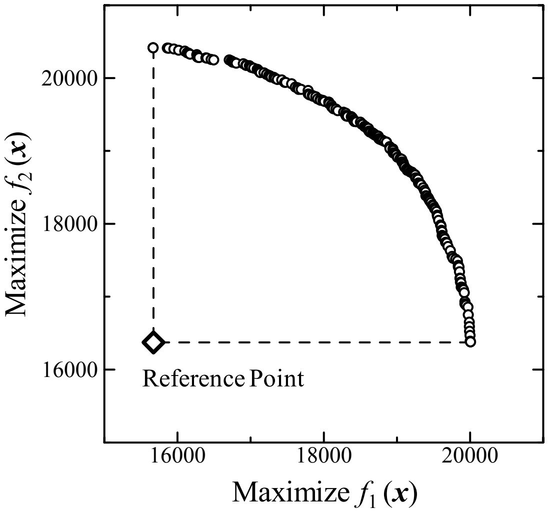

Fig. 1. Obtained non-dominated solutions of the 2-500 problem.

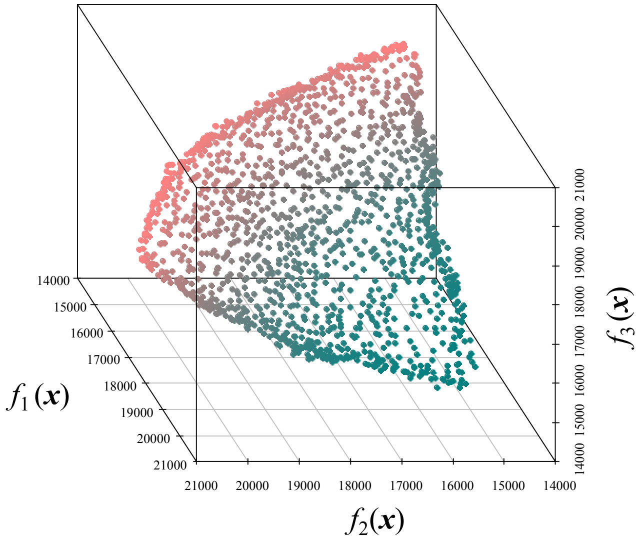

Fig. 2. Obtained non-dominated solutions of the 3-500 problem.

B. Indicator Function

As we have already explained, we used the hypervolume measure as the indicator function $I(S)$ in our computational experiments. It has been pointed out in several studies (e.g., [27]) that the calculation of the hypervolume heavily depends on the location of the reference point. If the location of the reference point is far from a solution set, extreme solutions around the edge of the solution set have large effects on the hypervolume calculation. In other words, extreme solutions have large contributions to the hypervolume. On the other hand, the hypervolume contribution of each extreme solution is decreased by moving the reference point towards the solution set. We specified the reference point for each of the two test problems by the minimum value of each objective among the given non-dominated solutions as shown in Fig. 1. Exactly speaking, the location of the reference point was specified by the minimum value minus one. The effect of the location of the reference point on finally obtained solution sets will be demonstrated in the following subsections.

C. Results on a Two-Objective Problem

First we show experimental results on the 2-500 problem where 212 non-dominated solutions in Fig. 1 were given. Our steady-state genetic algorithm was applied to the penalty function in (15) with the following parameter specifications:

Population size: 50,
Crossover probability: 0.8 (Uniform crossover),
Mutation probability: 1/500 (Bit-flip mutation),
Termination condition: 100,000 evaluations of solution sets,
The upper limit on the number of solutions: 10, 20, 30, 40.

Experimental results are shown in Fig. 3 where the upper limit on the number of selected solutions was set at ten. Exactly ten solutions were selected in Fig. 3. The reference point was set as shown in Fig. 1 by the minimum value of each objective among the given non-dominated solutions. On the other hand, Fig. 4 shows experimental results by the reference point at (0, 0) of the two-dimensional objective space. The reference point (0, 0) is not shown in Fig. 4 since its location is out of the range of this figure. The difference in the settings of computational experiments between Fig. 3 and Fig. 4 was only the location of the reference point. Since the reference point (0, 0) was far from the given candidate solutions, solutions around the edges of their distribution in Fig. 1 have large hypervolume contributions. As a result, the selected solutions have a wider range in Fig. 4 than Fig. 3.

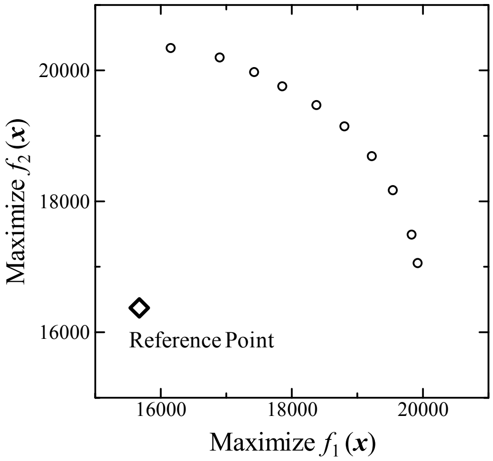

Fig. 3. Selected solutions and the reference point (2-500).

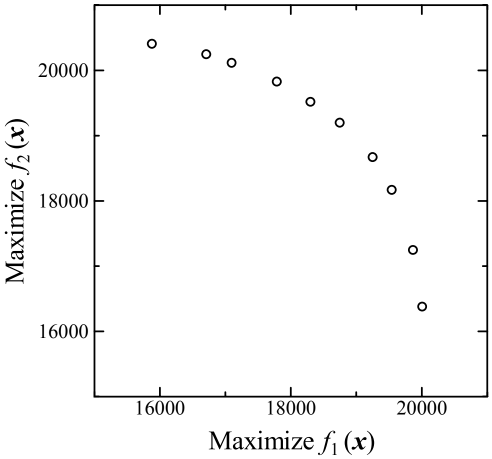

Fig. 4. Results by the reference point at (0, 0) for the 2-500 problem.

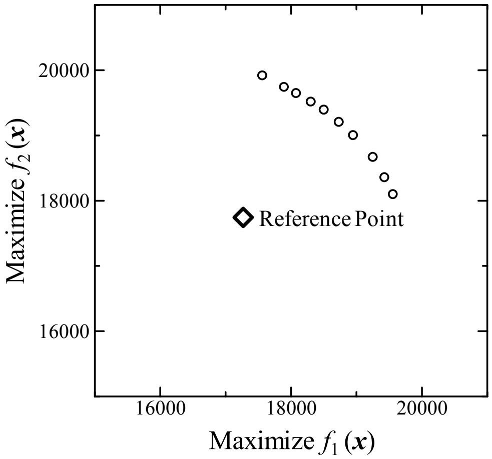

Fig. 5. Results by another reference point for the 2-500 problem.

We further demonstrate the effect of the location of the reference point on the finally obtained solution set in Fig. 5. It should be noted that the difference among Figs. 3-5 was only the location of the reference point. These figures clearly demonstrate the effect of the location of the reference point on the distribution of the selected solutions. These figures also suggest the possibility to focus of the candidate solution search on a certain region of the Pareto front (e.g., the center region of the Pareto front in Fig. 5). This may lead to a multi-step solution selection procedure. In the first step, candidate solutions over the entire Pareto front as in Fig. 4 are presented to the decision maker. Then the candidate solution search can be focused on a certain region of the Pareto front by moving the location of the reference point based on the information about the chosen solution in the first step.

D. Results on a Three-Objective Problem

We applied our approach to the 1444 solutions in Fig. 3. Experimental results are shown in Fig. 6 and Fig. 7 where the upper bound on the number of solutions was specified as 10 and 40, respectively. We can see that our approach worked well for the case of three objectives with many candidates.

In Fig. 8, we show experimental results after moving the reference point to (0, 0, 0). As explained in Fig. 4, our approach tends to find solutions around the edge of the Pareto front when the reference point is far from the Pareto front.

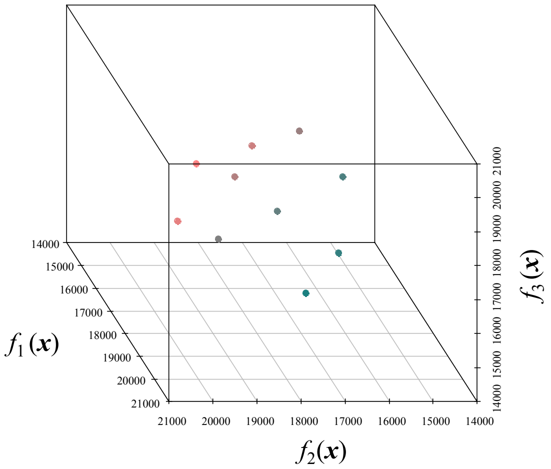

Fig. 6. Selected 10 solutions for the 3-500 problem.

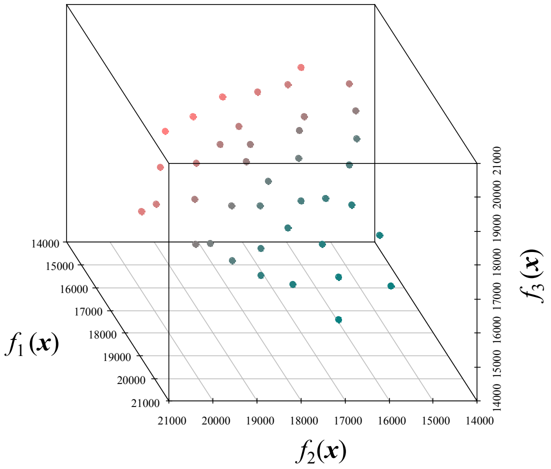

Fig. 7. Selected 40 solutions for the 3-500 problem.

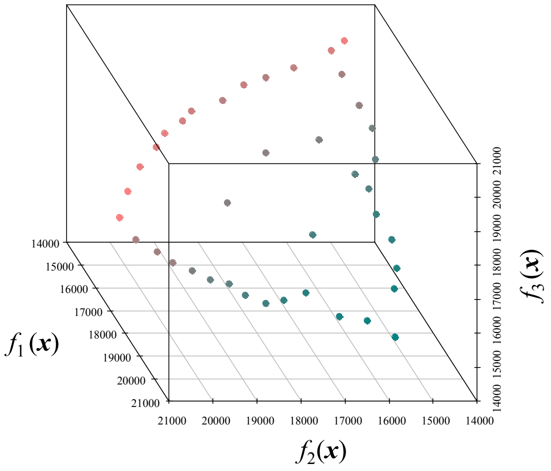

Fig. 8. Selected 40 solutions by the reference point at (0, 0, 0).

V. Concluding Remarks

We proposed an idea of solution selection to decrease the number of non-dominated solutions presented to the decision maker. The basic idea is to minimize the number of presented solutions while maximizing their hypervolume. The proposed idea was formulated as single-objective and multiobjective problems. The proposed idea was also implemented as evolutionary solution selection algorithms, which can be viewed as a special type of IBEAs. The main characteristic feature of our implementation is that an individual is a set of solutions. This means that the population is a set of solution sets. Through computational experiments, we demonstrated that a small number of uniformly distributed solutions can be selected from a large number of candidate solutions by our approach. We also demonstrated that the location of the reference point can be used to focus the search of our approach to a certain region of the Pareto front.

References

[1] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, Chichester (2001).

[2] C. A. C. Coello, D. A. van Veldhuizen, and G. B Lamont, *Evolutionary Algorithms for Solving Multi-Objective Problems*. Kluwer, Boston (2002).

[3] C. A. C. Coello and G. B. Lamont, *Applications of Multi-Objective Evolutionary Algorithms*. World Scientific, Singapore (2004).

[4] K. C. Tan, E. F. Khor, and T. H. Lee, *Multiobjective Evolutionary Algorithms and Applications*, Springer, Berlin (2005).

[5] A. Abraham, L. C. Jain, and R. Goldberg (eds.) *Evolutionary Multiobjective Optimization: Theoretical Advances and Applications*. Springer, Berlin (2005).

[6] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. on Evolutionary Computation* 6, 2 (2002) 182-197.

[7] E. Zitzler and L. Thiele, Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Trans. on Evolutionary Computation* 3, 4 (1999) 257-271.

[8] H. Ishibuchi, N. Tsukamoto, and Y. Nojima, Evolutionary many-objective optimization: A short review. *Proc. of 2008 IEEE Congress on Evolutionary Computation* (2008) 2424-2431.

[9] H. Ishibuchi, N. Tsukamoto, Y. Hitotsuyanagi, and Y. Nojima, Effectiveness of scalability improvement attempts on the performance of NSGA-II for many-objective problems. *Proc. of 2008 Genetic and Evolutionary Computation Conference* (2008) 649-656.

[10] E. Zitzler and S. Künzli, Indicator-based selection in multiobjective search. *Lecture Notes in Computer Science 3242: Parallel Problem Solving from Nature - PPSN VIII*, Springer, Berlin (2004) 832-842.

[11] M. Emmerich, N. Beume, and B. Naujoks, An EMO algorithm using the hypervolume measure as selection criterion. *Lecture Notes in Computer Science 3410: Evolutionary Multi-Criterion Optimization - EMO 2005*, Springer, Berlin (2005) 62-76.

[12] N. Beume, B. Naujoks, and M. Emmerich, SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research* 180, 3 (2007) 1653-1669.

[13] D. Brockhoff and E. Zitzler, Improving hypervolume-based multiobjective evolutionary algorithms by using objective reduction methods. *Proc. of 2007 IEEE Congress on Evolutionary Computation* (2007) 2086-2093.

[14] T. Wagner, N. Beume, and B. Naujoks, Pareto-, aggregation-, and indicator-based methods in many-objective optimization. *Lecture Notes in Computer Science 4403: Evolutionary Multi-Criterion Optimization - EMO 2007*, Springer, Berlin (2007) 742-756.

[15] J. D. Knowles, D. W. Corne, and M. Fleischer, Bounded archiving using the Lebesgue measure. *Proc. of 2003 IEEE Congress on Evolutionary Computation* (2003) 2490-2497.

[16] E. Zitzler and L. Thiele, Multiobjective optimization using evolutionary algorithms – A comparative case study. *Lecture Notes in Computer Science 1498: Parallel Problem Solving from Nature - PPSN V* (1998) 292-301.

[17] L. Thiele, K. Miettinen, P. J. Korhonen, and J. Molina, A preference-based interactive evolutionary algorithm for multiobjective optimization. *Helsinki School of Economics, Working Paper* (2007) W-412.

[18] E. Zitzler, D. Brockhoff, and L. Thiele, The hypervolume indicator revisited: On the design of Pareto-compliant indicators via weighted integration. *Lecture Notes in Computer Science 4403: Evolutionary Multi-Criterion Optimization - EMO 2007*, Springer, Berlin (2007) 862-876.

[19] E. Zitzler, L. Thiele, and J. Bader, SPAM: Set preference algorithm for multiobjective optimization. *Lecture Notes in Computer Science 5199: Parallel Problem Solving from Nature - PPSN X*, Springer, Berlin (2008) 847-858.

[20] A. Auger, J. Bader, D. Brockhoff, and E. Zitzler, Theory of the hypervolume indicator: Optimal μ-distributions and the choice of the reference point. *Foundation of Genetic Algorithm: FOGA 2009* (Workshop Booklet with No Page Numbers).

[21] H. Ishibuchi, Y. Sakane, N. Tsukamoto, and Y. Nojima, Single-objective and multi-objective formulations of solution selection for hypervolume maximization. *Proc. of 2009 Genetic and Evolutionary Computation Conference* (2009) (in press).

[22] E. Zitzler, D. Brockhoff, and L. Thiele, The hypervolume indicator revisited: On the design of Pareto-compliant indicators via weighted integration. *Lecture Notes in Computer Science 4403: Evolutionary Multi-Criterion Optimization - EMO 2007*, Springer, Berlin (2007) 862-876.

[23] J. Bader, D. Brockhoff, S. Welten, and E. Zitzler, On using populations of sets in multiobjective optimization. *Lecture Notes in Computer Science: Evolutionary Multi-Criterion Optimization - EMO 2009*, Springer, Berlin (2009) 140-154.

[24] H. Ishibuchi, Y. Sakane, N. Tsukamoto, and Y. Nojima, Effects of using two neighborhood structures on the performance of cellular evolutionary algorithms for many-objective optimization. *Proc. of 2009 IEEE Congress on Evolutionary Computation* (2009) 2508-2515.

[25] Q. Zhang and H. Li, MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Trans. on Evolutionary Computation* 11, 6 (2007) 712-731.

[26] H. Li and Q. Zhang, Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II, *IEEE Trans. on Evolutionary Computation* 13, 2 (2009) 284-302.

[27] H. Ishibuchi, Y. Nojima, and T. Doi, Comparison between single-objective and multi-objective genetic algorithms: Performance comparison and performance measures. *Proc. of 2006 IEEE Congress on Evolutionary Computation* (2006) 16-21.