Indicator-Based Evolutionary Algorithm with Hypervolume Approximation by Achievement Scalarizing Functions

Hisao Ishibuchi, Noritaka Tsukamoto, Yuji Sakane and Yusuke Nojima
Graduate School of Engineering, Osaka Prefecture University
1-1 Gakuen-cho, Naka-ku, Sakai, Osaka 599-8531, Japan
Phone: +81-72-254-9350
{hisaoi@, nori@ci., sakane@ci, nojima@}cs.osakafu-u.ac.jp

ABSTRACT
Pareto dominance-based algorithms have been the main stream in the field of evolutionary multiobjective optimization (EMO) for the last two decades. It is, however, well-known that Pareto-dominance-based algorithms do not always work well on many-objective problems with more than three objectives. Currently alternative frameworks are studied in the EMO community very actively. One promising framework is the use of an indicator function to find a good solution set of a multiobjective problem. EMO algorithms with this framework are called indicator-based evolutionary algorithms (IBEAs) where the hypervolume measure is frequently used as an indicator. IBEAs with the hypervolume measure have strong theoretical support and high search ability. One practical difficult of such an IBEA is that the hypervolume calculation needs long computation time especially when we have many objectives. In this paper, we propose an idea of using a scalarizing function-based hypervolume approximation method in IBEAs. We explain how the proposed idea can be implemented in IBEAs. We also demonstrate through computational experiments that the proposed idea can drastically decrease the computation time of IBEAs without severe performance deterioration.

Categories and Subject Descriptors
I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search – *Heuristic Methods.*

General Terms
Algorithms.

Keywords
Evolutionary multiobjective optimization (EMO), hypervolume, indicator-based evolutionary algorithm (IBEA), many objectives.

1. INTRODUCTION
Evolutionary multiobjective optimization (EMO) is a very active research area in the field of evolutionary computation. The use of Pareto dominance for fitness evaluation has been the main stream in the EMO community for the last two decades since Goldberg's suggestion in the late 1980s [15]. Pareto dominance-based EMO algorithms have been successfully applied to multiobjective problems in various application areas [9], [10], [27]. However, it has been repeatedly pointed out in the literature [16]-[18], [20], [21], [26], [37] that Pareto dominance-based EMO algorithms do not always work well on many-objective problems with more than three objectives. This is because Pareto dominance cannot generate a strong selection pressure toward the Pareto front when we have many objectives (since almost all solutions in the current population become non-dominated). Thus alternative frameworks to Pareto dominance-based EMO algorithms are studied actively.

There exist at least two trends in recent studies on alternatives to Pareto dominance-based EMO algorithms. One trend is the use of scalarizing functions to evaluate each solution (i.e., scalarizing function-based EMO algorithms [16], [17], [22], [31]). The main advantage of the use of scalarizing functions for fitness evaluation is its simplicity and efficiency. This is because scalarizing functions can be easily calculated even for many objectives.

The other trend is the use of an indicator function to evaluate a set of solutions (i.e., indicator-based EMO algorithms [33]). EMO algorithms with indicator functions are often referred to as IBEAs (indicator-based evolutionary algorithms). The hypervolume measure [35] has been often used in IBEAs [6], [14] due to its theoretically good characteristic features [36] and the high search ability of hypervolume-based algorithms [28]. The incorporation of the decision maker's preference in the hypervolume measure has been also actively discussed [1], [2], [32]. The main practical difficulty in the use of the hypervolume measure for many-objective problems is its high computation cost. The computation time for the hypervolume calculation increases exponentially with the number of objectives. For tackling this difficulty, efficient hypervolume calculation and hypervolume approximation have been also actively studied [3]-[5], [7], [8], [19], [29].

In this paper, we propose an idea of using an achievement scalarizing function-based hypervolume approximation method [19] in IBEAs and examine the effectiveness of this idea. We chose the achievement scalarizing function-based method because (i) it can be easily applied to many-objective problems even when the number of objective is actually large and (ii) its approximation accuracy and computation load can be adjusted by the number of scalarizing functions (i.e., by the number of weight vectors).

This paper is organized as follows. First we briefly explain the achievement scalarizing function-based approximation method [19] of the hypervolume measure in Section 2. Next we propose

GECCO'10, July 7-11, 2010, Portland, Oregon, USA.
Copyright 2010 ACM 978-1-4503-0072-8/10/07…$10.00.

an idea of using the hypervolume approximation method in IBEAs in Section 3. This idea is implemented by combining the hypervolume approximation method into a high-performance IBEA called SMS-EMOA (*S* metric selection EMO algorithm [6]). Then the performance of SMS-EMOA with the hypervolume approximation method is evaluated by comparing it with NSGA-II [11], SPEA2 [34], MOEA/D [31] and the original SMS-EMOA with no hypervolume approximation through computational experiments in Section 4. Experimental results show that the computation time of SME-EMOA can be drastically decreased by the proposed idea without severe performance deterioration. Finally we conclude this paper in Section 5.

2. HYPERVOLUME APPROXIMATION

2.1 Multiobjective Optimization Problem

Let us consider the following *k*-objective minimization problem:

$$\text{Minimize } \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), ..., f_k(\mathbf{x})) \text{ subject to } \mathbf{x} \in \mathbf{X}, \tag{1}$$

where $\mathbf{f}(\mathbf{x})$ is the *k*-dimensional objective vector, $f_i(\mathbf{x})$ is the *i*-th objective to be minimized ($i = 1, 2, ..., k$), $\mathbf{x}$ is the decision vector, and $\mathbf{X}$ is the feasible region in the decision space. Let $\mathbf{y}$ and $\mathbf{z}$ be two feasible solutions of the *k*-objective minimization problem in (1). If the following conditions hold, $\mathbf{y}$ can be viewed as being better than $\mathbf{z}$ for the minimization problem in (1):

$$\forall\, i : f_i(\mathbf{y}) \le f_i(\mathbf{z}) \text{ and } \exists\, j : f_j(\mathbf{y}) < f_j(\mathbf{z}). \tag{2}$$

In this case, we say that **y** dominates **z** (equivalently **z** is dominated by **y**: **y** is better than **z**).

If **y** is not dominated by any other feasible solutions, **y** is referred to as a Pareto-optimal solution of the *k*-objective minimization problem in (1). The set of all Pareto-optimal solutions forms the tradeoff surface in the objective space. This tradeoff surface is referred to as the Pareto front (or the Pareto frontier). EMO algorithms are usually designed to search for a set of well-distributed non-dominated solutions that approximates the entire Pareto front very well.

2.2 Hypervolume Approximation Method

Let S be a set of N solutions: $S = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$. The hypervolume of the solution set S is the volume of the region dominated by S in the objective space. A reference point $\mathbf{r} = (r_1, \ldots, r_k)$ in the objective space is needed to limit this region. In Fig. 1, the hypervolume of the set of four solutions $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$, $\mathbf{x}_4$ is illustrated as the area of the shaded region for the reference point $\mathbf{r} = (r_1, r_2)$.

An idea of approximating the hypervolume using a number of achievement scalarizing functions with uniformly distributed weight vectors was proposed by Ishibuchi et al. [19]. In this subsection, we explain the achievement scalarizing function-based approximation method [19], which will be combined in SMS-EMOA [6] in Section 3.

An achievement scalarizing function for the multiobjective minimization problem in (1) can be written for a positive weight vector $\mathbf{w} = (w_1, w_2, ..., w_k)$ and the reference point $\mathbf{r} = (r_1, \ldots, r_k)$ as follows [23], [24], [30]:

$$g(\mathbf{x}, \mathbf{r}, \mathbf{w}) = \max_{i=1,\ldots,k} w_i (f_i(\mathbf{x}) - r_i), \tag{3}$$

where the same reference point **r** is used as in the definition of the hypervolume. This function is to be minimized.

An achievement scalarizing function is illustrated in Fig. 2 where its contour lines are shown by parallel solid lines for the positive weight vector $\mathbf{w} = (w_1, w_2) = (3, 2)$. The dashed line in Fig. 2 can be viewed as a search direction to minimize the achievement scalarizing function. The direction of a positive vector along the dashed line is calculated as $(1/w_1, 1/w_2) = (1/3, 1/2)$, i.e., (2, 3).

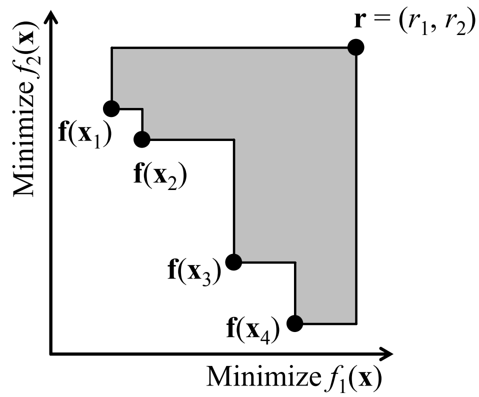

Figure 1. Hypervolume of a solution set with four solutions.

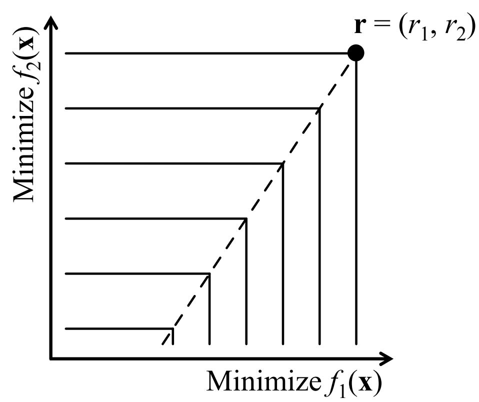

Figure 2. Contour lines of an achievement scalarizing function.

In order to approximate the hypervolume of the solution set S, first the minimum value of the achievement scalarizing function in (3) is calculated over all solutions in S as follows:

$$\begin{aligned} g^*(S, \mathrm{r}, \mathbf{w}) &= \min_{\mathbf{x} \in S} g(S, \mathrm{r}, \mathbf{w}) \\ &= \min_{\mathbf{x} \in S} \max_{i=1,\ldots,k} w_i (f_i(\mathbf{x}) - r_i). \end{aligned} \tag{4}$$

This calculation is illustrated in Fig. 3 for the achievement scalarizing function in Fig. 2 and the solution set S in Fig. 1. In Fig. 3, the minimum value is obtained by $\mathbf{x}_3$.

The basic idea of hypervolume approximation in [19] is to measure the distance from the reference point **r** to the solution set S (more specifically, to the attainment surface realized by S) using the achievement scalarizing function. As shown in Fig. 3, let "A" be the intersecting point of the contour line with the minimum value in (4) and the search direction line (i.e., the dashed line in Fig. 3). This point "A" is also on the attainment surface as shown in Fig. 4. This means that the length of the line from the reference point **r** to "A" is the distance from the reference point **r** to the attainment surface of the solution set S along the search direction depicted by the dashed line with the direction $(1/w_1, 1/w_2)$.

The distance from the reference point **r** to the attainment surface can be calculated for various weight vectors as shown in Fig. 5. From Fig. 5, we can see intuitively that the average length of the lines with various search directions in Fig. 5 is closely related to the hypervolume.

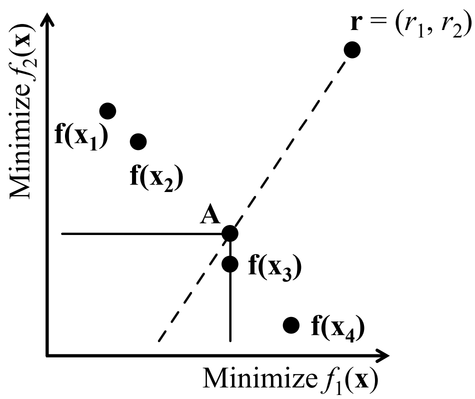

Figure 3. Minimum value of (4) over *S*.

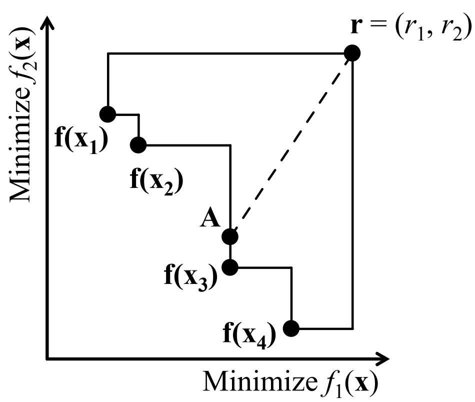

Figure 4. Distance between the reference point r and the attainment surface.

The distance from the reference point **r** to the attainment surface is calculated for various weight vectors $\mathbf{w} = (w_1, w_2, ..., w_k)$. Then the average distance is calculated. Uniformly distributed weight vectors can be generated from the following formulation as in Murata et al. [25] and Zhang & Li [31]:

$$w_1 + w_2 + \cdots + w_k = d\ , \quad (5)$$

$$w_i \in \{0, 1, ..., d\} \text{ for } i = 1, 2, ..., k\ , \quad (6)$$

where d is a pre-specified integer.

For example, when the value of d is specified as $d = 2$ for a three-objective problem (i.e., $k = 3$), six weight vectors (2, 0, 0), (1, 1, 0), (1, 0, 1), (0, 2, 0), (0, 1, 1), (0, 0, 2) are generated. The generated weight vectors are normalized so that their length is one when they are used in the achievement scalarizing function.

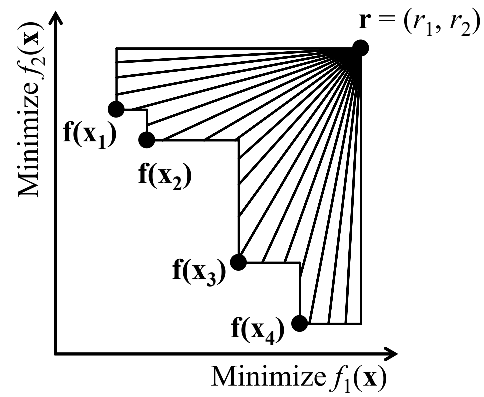

Figure 5. Illustration of the hypervolume approximation.

3. COMBINATION WITH SMS-EMOA

3.1 SMS-EMOA

SMS-EMOA [6] is a hypervolume-based IBEA with the $(\mu+1)$-ES generation update mechanism. In SMS-EMOA, each individual in the current population is primarily evaluated by the non-dominated sorting in the same manner as NSGA-II [11]. The hypervolume contribution is used as a secondary criterion to evaluate individuals with the worst rank. The worst individual with the minimum hypervolume contribution among those individuals with the worst rank is removed from the current generation. It was demonstrated by Wagner et al. [28] that SMS-EMOA outperformed many other representative EMO algorithms on many-objective problems. Whereas SMS-EMOA has high search ability, it also needs large computation load for the calculation of hypervolume contribution of each individual with the worst rank. It should be noted that almost all individuals are non-dominated with each other in the case of many-objective problems. Thus the calculation of hypervolume contribution is needed for almost all individuals in this case, which leads to long computation time of SMS-EMOA.

3.2 Use of Hypervolume Approximation

In this subsection, we propose an idea of using the hypervolume approximation method in Section 2 in SMS-EMOA. That is, we modify the hypervolume approximation method for the calculation of hypervolume contribution in SMS-EMOA.

Let us consider the calculation of the hypervolume contribution of $\mathbf{x}_3$ in Fig. 5. First we calculate the direction of the vector $\mathbf{w_A}$ from $\mathbf{x}_3$ to the reference point **r** as shown in Fig. 6. This vector shows the search direction depicted by the dashed line in Fig. 6. Then we calculate the weight vector $\mathbf{w_B}$ such that the weight vector $\mathbf{w_A}$ is the search direction of the achievement scalarizing function with the weight vector $\mathbf{w_B}$. The weight vector $\mathbf{w_B}$ can be obtained by calculating the element-wise inverse (e.g., $\mathbf{w_B}$ = ($1/w_{A1}$, $1/w_{A2}$) for $\mathbf{w_A} = (w_{A1}, w_{A2})$). The obtained vector is normalized.

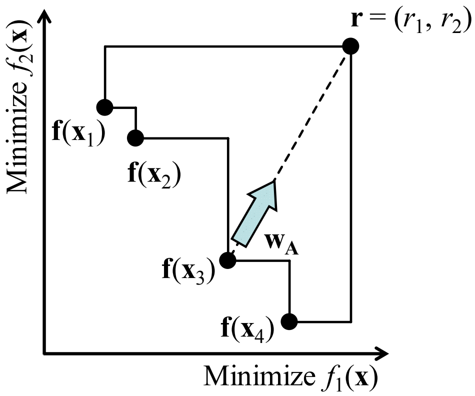

Figure 6. Direction of the weight vector $\mathbf{w_A}$.

Next we generate a number of uniformly distributed lines from the reference point **r** around the dashed line in Fig. 6 (i.e., around the solution $\mathbf{x}_3$) as shown in Fig. 7. When the hypervolume of the solution set S = {$\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$, $\mathbf{x}_4$} was approximated, all subspace that dominates the reference point **r** was covered by a set of uniformly distributed lines as shown in Fig. 5. On the other hand, we use somewhat focused lines around the solution $\mathbf{x}_3$ in order to calculate its hypervolume contribution as in Fig. 7.

In the simplest case, we use only the dashed line in Fig. 6 in order to calculate the hypervolume contribution of $\mathbf{x}_3$ whereas we show a number of uniformly distributed lines in Fig. 7 for illustration purpose. That is, we use only a single line for the calculation of the hypervolume contribution of each solution. The line is easily obtained by connecting the reference point **r** and the solution for which the hypervolume contribution is to be calculated.

When multiple lines are used for the contribution calculation, we first specify the range for each element of the weight vector **w** = (w_1, w_2, ..., w_k) in the achievement scalarizing function as follows:

$$w_i \in [w_{Bi} - \delta,\ w_{Bi} + \delta] \ \text{for}\ i = 1, 2, ..., k\ , \qquad (7)$$

where w_{Bi} is the i-th element of the weight vector $\mathbf{w_B}$ calculated in the above-mentioned manner, and δ is a small real number. The set of uniformly distributed weight vectors satisfying (5) and (6) are transformed so that they satisfy (7). In order to have a set of uniformly distributed weight vectors around $\mathbf{w_B}$ (i.e., a set of uniformly distributed lines around the dashed line in Fig. 7), the center vector (1, 1, ..., 1) among the generated weight vectors by (5) and (6) are transformed to the weight vector $\mathbf{w_B}$. The other weight vectors are transformed accordingly in the range specified by (7). The transformed weight vectors are normalized before they are used in the achievement scalarizing function.

An appropriate specification of the value of δ in (7) is not easy. Intuitive speaking, if we have many solutions in the solution set S, the value of δ can be small since we can concentrate on a small region around the solution (e.g., $\mathbf{x}_3$ in Fig. 7 for which the hypervolume contribution is to be calculated). On the other hand, if we do not have many solutions in the solution set S, a large value of δ seems to be needed.

In our computational experiments, we specified δ as follows:

$$\delta = 1/(|S_{\text{Worst}}| - 1), \qquad (8)$$

where $|S_{\text{Worst}}|$ denotes the number of solutions with the worst rank in the current population.

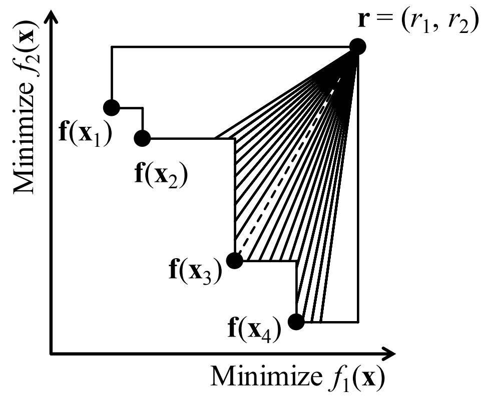

Figure 7. Approximating the hypervolume of the solution set S around $\mathbf{x}_3$.

When a set of uniformly distributed weight vectors is generated around $\mathbf{w_B}$, we can approximately calculate the hypervolume of the solution set S around the solution $\mathbf{x}_3$ as shown in Fig. 7. In the same manner, we can also approximately calculate the hypervolume of the solution set S excluding $\mathbf{x}_3$ as shown in Fig. 8.

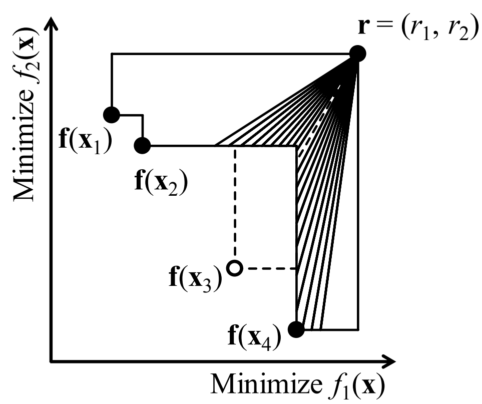

Figure 8. Approximating the hypervolume of $S\backslash\{\mathbf{x}_3\}$ around $\mathbf{x}_3$.

The hypervolume contribution of $\mathbf{x}_3$ can be approximately calculated as the difference between Fig. 7 and Fig. 8, which is illustrated in Fig. 9. From Fig. 9, we can see that we need only those lines for which the achievement scalarizing function assumes the minimum value at $\mathbf{x}_3$ among the four solutions in S. The hypervolume contribution of $\mathbf{x}_3$ can be approximated by averaging the length of the lines in Fig. 9.

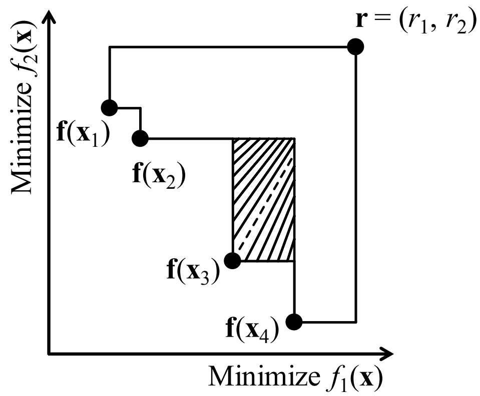

Figure 9. Calculation of the hypervolume contribution of $\mathbf{x}_3$.

4. COMPUTATIONAL EXPERIMENTS

4.1 Test Problems

As test problems, we used DTLZ1 and DTLZ2 [12], [13] with three and six objectives. A series of DTLZ test problems are scalable in the number of objectives and variables. The number of variables L in DTLZ1 and DTLZ2 is $k+M-1$. In Deb et al. [12], [13], $M=5$ and $M=10$ were recommended for DTLZ1 and DTLZ2, respectively. We also used these specifications.

The Pareto front of DTLZ1 is a linear hyper-plane satisfying $f_1+f_2+\ ...\ +f_k=0.5$. On the other hand, DTLZ2 has a spherical Pareto front satisfying $|\mathbf{f}(\mathbf{x})| = 1$.

4.2 Performance Measures

We used the following two performance measures [28] to evaluate the convergence of solutions toward the Pareto front and their diversity along the Pareto front.

Convergence Measure

This measure was used to calculate the convergence of solutions toward the Pareto front in the objective space. The convergence measures for DTLZ1 and DTLZ2 were calculated as follows [28]:

$$d_{\mathrm{DTLZ1}} = |f_1+f_2+\ ...\ +f_k-0.5|, \tag{9}$$

$$d_{\mathrm{DTLZ2}} = |\mathbf{f}(\mathbf{x})| - 1. \tag{10}$$

Relative Hypervolume

The reference points $\mathbf{r} = 0.7^k$ for DTLZ1 and $\mathbf{r} = 1.1^k$ for DTLZ2 were used as in [28]. Solutions that did not dominate the reference point $\mathbf{r}$ were discarded in hypervolume calculation. The relative hypervolume was calculated using the analytical optimal value.

We also report the average computational time of each algorithm for each test problem. The computational time was measured on a PC with Intel (R) Xeon (R) CPU 3.2 GHz.

4.3 EMO Algorithms

We used the following EMO algorithms in our computational experiments.

NSGA-II [11] (Pareto dominance-based EMO algorithm)
SPEA2 [34] (Pareto dominance-based EMO algorithm)
MOEA/D [31] (Scalarizing function-based EMO algorithm)
IBEA [33] (Indicator-based EMO algorithm)
SMS-EMOA [6] (Indicator-based EMO algorithm)

SMS-EMOA using the proposed approximation method of the hypervolume contribution is referred to as SMS-EMOA-Apr(X) where X is an integer indicating the number of lines (weight vectors) used for the approximation of hypervolume contribution.

4.4 Parameter Settings

We used the following parameter settings in our computational experiments. These settings are the same as Wagner et al. [28].

Population size: 100 (except for MOEA/D),
Crossover probability: 1.0 (SBX [10]),
Mutation probability: $1/L$ (Polynomial mutation [10]),
Stopping condition: 30000 fitness evaluations.

The population size of MOEA/D is set as follows (to use a set of uniformly distributed weight vectors for each problem): 105 for three-objective problems, and 126 for six-objective problem. The distribution indices for SBX and polynomial mutation were specified as 15 (i.e., $\eta_c = 15$) and 20 (i.e., $\eta_m = 15$), respectively.

We examined the following three cases with respect to the number of lines (i.e., weight vectors) for the approximation of hypervolume contribution of each solution:

(i) Use of only a single weight vector: $\mathbf{w} = (1, 1, \ldots, 1)$.
(ii) Use of the $(k + 1)$ weight vectors: $\mathbf{w} = (1, 1, \ldots, 1), (1, 0, \ldots, 0), (0, 1, \ldots, 0), (0, 0, \ldots, 1)$.
(iii) Specification of d in (5) and (6) as the number of objectives. In this case, the number of weight vectors is 10 for the three-objective problems and 462 for the six-objective problems.

4.5 Experimental Results

We summarize the results for the three-objective DTLZ1 and DTLZ2 in Table 1 and Table 2, respectively, where the best results with respect to the relative hypervolume were obtained by SMS-EMOA. At the same time, SMS-EMOA needed the longest computation time among the examined EMO algorithms. The proposed method slightly deteriorated the performance of SMS-EMOA and significantly decreased its computation time. We show the obtained solutions for the three-objective DTLZ2 by a single run of MOEA/D, SMS-EMOA and SMS-EMOA-Apr(10) in Figs. 10-12, respectively. Very similar (almost the same) results were obtained by SMS-EMOA in Fig. 11 and SMS-EMOA-Apr(10) in Fig. 12.

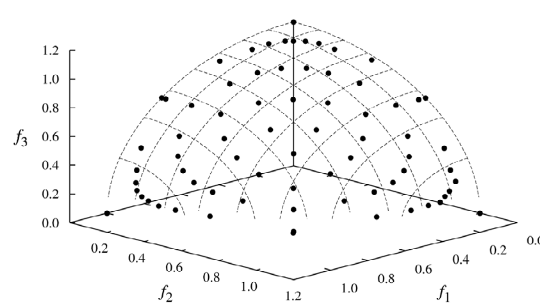

Figure 10. Solutions obtained by MOEA/D for DTLZ2.

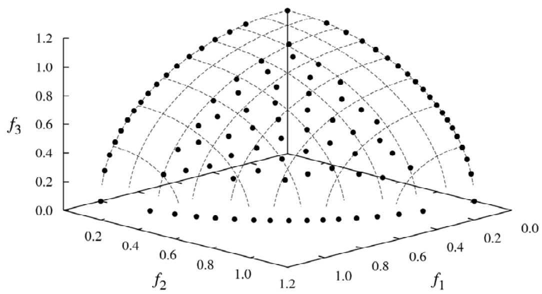

Figure 11. Solutions obtained by SMS-EMOA for DTLZ2.

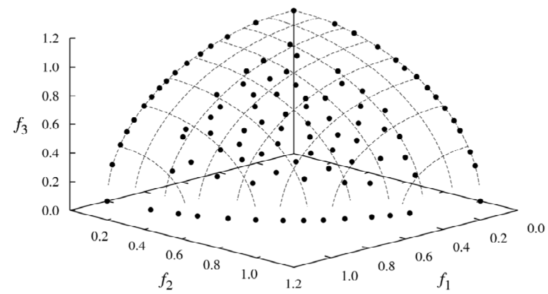

Figure 12. Solutions obtained by SMS-EMOA-Apr(10).

Table 1 Experimental results on the three-objective DTLZ1

Algorithm	Convergence Measure		Relative Hypervolume		Computation Time (Sec.)	
	mean	std.dev	mean	std.dev	mean	std.dev
NSGA-II	0.26679	0.75681	0.94640	0.09813	0.6	0.00764
SPEA2	0.07173	0.12538	0.95371	0.11162	1.6	0.01152
MOEA/D	0.00228	0.00197	0.96546	0.00097	0.4	0.00486
IBEA	0.00068	0.00076	0.74904	0.03671	3.0	0.00991
SMS-EMOA	0.00070	0.00050	0.98369	0.00029	64.9	3.73887
SMS-EMOA-Apr(1)	0.00157	0.00160	0.95367	0.00834	14.0	0.19966
SMS-EMOA-Apr(4)	0.00148	0.00134	0.97950	0.00148	18.4	0.65777
SMS-EMOA-Apr(10)	0.00100	0.00071	0.98195	0.00079	27.9	1.15496

Table 2 Experimental results on the three-objective DTLZ2

Algorithm	Convergence Measure		Relative Hypervolume		Computation Time (Sec.)	
	mean	std.dev	mean	std.dev	mean	std.dev
NSGA-II	0.00913	0.00150	0.86783	0.00628	0.7	0.00521
SPEA2	0.00807	0.00127	0.90686	0.00307	1.9	0.01096
MOEA/D	2.0×10^{-5}	1.0×10^{-6}	0.87775	0.00107	0.5	0.00349
IBEA	1.3×10^{-5}	8.4×10^{-6}	0.92031	0.00070	2.9	0.02242
SMS-EMOA	1.7×10^{-5}	1.3×10^{-6}	0.93878	1.7×10^{-5}	145.9	1.05778
SMS-EMOA-Apr(1)	0.00094	0.00035	0.92789	0.00116	18.5	0.05679
SMS-EMOA-Apr(4)	0.00077	0.00014	0.93079	0.00049	32.3	0.13859
SMS-EMOA-Apr(10)	0.00038	7.8×10^{-5}	0.93377	0.00056	61.9	0.63710

Table 3 Experimental results on the six-objective DTLZ1

Algorithm	Convergence Measure		Relative Hypervolume		Computation Time (Sec.)	
	Mean	std.dev	mean	std.dev	mean	std.dev
NSGA-II	356.702	30.8628	0.00000	0.00000	1.1	0.01374
SPEA2	419.299	23.9378	0.00000	0.00000	3.0	0.01711
MOEA/D	0.01323	0.03296	0.95624	0.02652	0.5	0.00763
IBEA	0.00220	0.00487	0.65256	0.08408	12.9	0.21376
SMS-EMOA	0.00333	0.00274	0.99698	0.00010	124364.0	5283.35
SMS-EMOA-Apr(1)	0.05668	0.16208	0.96589	0.00503	30.7	0.41392
SMS-EMOA-Apr(7)	0.05538	0.16554	0.98045	0.00215	73.9	2.65380
SMS-EMOA-Apr(462)	0.00332	0.00170	0.98053	0.00325	3391.2	233.544

Table 4 Experimental results on the six-objective DTLZ2

Algorithm	Convergence Measure		Relative Hypervolume		Computation Time (Sec.)	
	mean	std.dev	mean	std.dev	mean	std.dev
NSGA-II	1.60695	0.09077	0.00113	0.00205	1.2	0.00803
SPEA2	1.93405	0.07256	0.00035	0.00151	3.1	0.00797
MOEA/D	0.00070	0.00040	0.68517	0.02681	0.6	0.00695
IBEA	0.00014	5.0×10^{-5}	0.87407	0.05335	11.7	0.06397
SMS-EMOA	0.00023	4.9×10^{-6}	0.90462	4.4×10^{-5}	196311.0	6073.04
SMS-EMOA-Apr(1)	0.00456	0.00034	0.88299	0.00301	38.0	0.31043
SMS-EMOA-Apr(7)	0.00619	0.00054	0.88479	0.00163	117.3	0.18306
SMS-EMOA-Apr(462)	0.00498	0.00116	0.88599	0.00185	6138.1	26.6821

We summarize the results for the six-objective DTLZ1 and DTLZ2 in Table 3 and Table 4, respectively, where the best results with respect to the relative hypervolume were obtained by SMS-EMOA as in Table 1 and Table 2. Good results were not obtained by the two Pareto dominance-based EMO algorithms (i.e., NSGA-II and SPEA2) for the six-objective problems. As in Table 1 and Table 2, the use of the proposed approximation method slightly degraded the performance of SMS-EMOA. The decrease in computation time by the proposed method is much more significant in Table 3 and Table 4 for the six-objective problems (e.g., from 196311 seconds to 38 seconds) than Table 1 and Table 2 for the three-objective problems (e.g., from 145 second to 18).

With respect to the convergence measure, IBEA is the best in all the four tables. Very small values of the convergence measure were always obtained by IBEA. This means that the obtained non-dominated solutions by IBEA were very close to the Pareto front. This does not always mean that the overall performance of IBEA is the best among the examined EMO algorithms in Tables 1-4. For example, IBEA is the worst in Table 1 with respect to the relative hypervolume. This means that the obtained non-dominated solutions by IBEA do not cover the entire Pareto front. The average value of the relative hypervolume is also small in Table 3.

5. CONCLUDING REMARKS

In this paper, we proposed an idea of using an achievement scalarizing function-based method to approximate hypervolume contribution in IBEAs (indicator-based evolution algorithms). Our idea was implemented by combining the hypervolume contribution approximation method into a high-performance IBEA called SMS-EMOA. Through computational experiments on three-objective and six-objective test problems, it was demonstrated that the proposed idea drastically decreased the computation time of SMS-EMOA without severely deteriorating its performance. An interesting observation in our computational experiments is that SMS-EMOA had high search ability even when the hypervolume contribution of each solution was simply and very roughly approximated using only a single achievement scalarizing function corresponding to the line between the reference point and the solution to be evaluated. In this case, the computation time of SMS-EMOA was decreased to about 1/5000 for six-objective test problems.

In this paper, we used only four test problems with three and six objectives in our computational experiments. It is clear that we need further computational experiments on various test problems including difficult ones with complicated Pareto sets [22] in order to examine the effectiveness of the proposed idea over a wide range of multiobjective problems. It is also needed to examine the sensitivity of the performance of the proposed idea on the specification of the number of weight vectors in detail. In addition to further computational experiments, it is also a future research topic to implement the proposed idea in a much simpler manner. The current implementation explained in Subsection 3.2 is very complicated. More intuitively understandable implementation may be needed for increasing its practical usability as a convenient heuristic method to approximate hypervolume contribution.

6. REFERENCES

[1] Auger, A., Bader, J., Brockhoff, D., and Zitzler, E. Investigating and exploiting the bias of the weighted hypervolume to articulate user preferences. *Proc. of 2009 Genetic and Evolutionary Computation Conference* (2009) 563-570.

[2] Auger, A., Bader, J., Brockhoff, D., and Zitzler, E. Articulating user preferences in many-objective problems by sampling the weighted hypervolume. *Proc. of 2009 Genetic and Evolutionary Computation Conference* (2009) 555-562.

[3] Bader, J., Deb, K., Zitzler, E. Faster hypervolume-based search using Monte Carlo sampling. *Lecture Notes in Economics and Mathematical Systems 634: MCDM 2008*, Springer, Berlin (2010) 313–326.

[4] Beume, N. S-metric calculation by considering dominated hypervolume as Klee's measure problem. *Evolutionary Computation* 17, 4 (2009) 477-492.

[5] Beume, N., Fonseca, C. M., Lopez-Ibanez, M., Paquete, L., and Vahrenhold, J. On the complexity of computing the hypervolume indicator. *IEEE Trans. on Evolutionary Computation* 13, 5 (2009) 1075-1082.

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

[7] Bradstreet, L., While, L., and Barone, L. A fast incremental hypervolume algorithm. *IEEE Trans. on Evolutionary Computation* 12, 6 (2008) 714-723.

[8] Bringmann, K., and Friedrich, T. Approximating the least hypervolume contributor: NP-hard in general, but fast in practice. *Lecture Notes in Computer Science 5467: Evolutionary Multi-Criterion Optimization - EMO 2009*. Springer, Berlin (2009) 6-20.

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

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

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

[12] Deb, K., Thiele, L., Laumanns, M., and Zitzler, E. Scalable multiobjective optimization test problems. *Proc. of 2002 Congress on Evolutionary Computation* (2002) 825-830.

[13] Deb, K., Thiele, L., Laumanns, M., and Zitzler, E. Scalable test problems for evolutionary multiobjective optimization. in Abraham, A., Jain, L. C., and Goldberg, R. (eds.), *Evolutionary Multiobjective Optimization: Theoretical Advances and Applications*, Springer, Berlin (2005) 105-145.

[14] Emmerich, M., Beume, N., and Naujoks, B. 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.

[15] Goldberg, D. E. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Reading (1989).

[16] Hughes, E. J. Evolutionary many-objective optimization: Many once or one many?. *Proc. of 2005 IEEE Congress on Evolutionary Computation* (2005) 222-227.

[17] Hughes, E. J. MSOPS-II: A general-purpose many-objective optimizer. *Proc. of 2007 IEEE Congress on Evolutionary Computation* (2007) 3944-3951.

[18] Ishibuchi, H., Tsukamoto, N., Hitotsuyanagi, Y., and Nojima, Y. 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.

[19] Ishibuchi, H., Tsukamoto, N., Sakane, Y., and Nojima, Y., Hypervolume approximation using achievement scalarizing functions for evolutionary many-objective optimization. *Proc. of 2009 IEEE Congress on Evolutionary Computation* (2009) 530-537.

[20] Jaszkiewicz, A. On the computational efficiency of multiple objective metaheuristics. The knapsack problem case study. *European Journal of Operational Research* 158, 2 (2004) 418-433.

[21] Khara, V., Yao, X., and Deb, K. Performance scaling of multi-objective evolutionary algorithms. *Lecture Notes in Computer Science 2632: Evolutionary Multi-Criterion Optimization - EMO 2003*, Springer, Berlin (2004) 367-390.

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

[23] Miettinen, K. *Nonlinear Multiobjective Optimization*. Kluwer, Boston (1998).

[24] Miettinen, K., and Makela, M. On scalarizing functions in multiobjective optimization. *OR Spectrum* 24, 2 (2002) 193-213.

[25] Murata, T., Ishibuchi, H., and Gen, M. Specification of genetic search directions in cellular multi-objective genetic algorithm, *Lecture Notes in Computer Science 1993: Evolutionary Multi-Criterion Optimization - EMO 2001*, Springer, Berlin (2001), 82-95.

[26] Purshouse, R. C. and Fleming, P. J. On the evolutionary optimization of many conflicting objectives. *IEEE Trans. on Evolutionary Computation* 11, 6 (2007) 770-784.

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

[28] Wagner, T., Beume, N., and Naujoks, B. 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.

[29] While, L., Hingston, P., Barone, L., and Huband, S. A faster algorithm for calculating hypervolume. IEEE Trans. on Evolutionary Computation 10, 1 (2006) 29-38.

[30] Wierzbicki, A. P., Makowski, M., and Wessels, J. (eds.), *Model-Based Decision Support Methodology with Environmental Applications*, Kluwer, Dordrecht (2000).

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

[32] Zitzler, E., Brockhoff, D., and Thiele, L. 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.

[33] Zitzler, E., and Künzli, S. Indicator-based selection in multiobjective search. *Proc. of PPSN 2004*, 832-842.

[34] Zitzler, E., and Laumanns, M., and Thiele, L. SPEA2: Improving the strength Pareto evolutionary algorithm. *TIK-Report*, No. 103 (2001).

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

[36] Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M., and da Fonseca, V. G. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Trans. on Evolutionary Computation* 7, 2 (2003) 117-132.

[37] Zou, X., Chen, Y., Liu, M., and Kang, L. A new evolutionary algorithm for solving many-objective optimization problems. *IEEE Trans. on Systems, Man, and Cybernetics: Part B - Cybernetics* 38, 5 (2008) 1402-1412.