2011 Seventh International Conference on Computational Intelligence and Security

MOEA/D with Uniform Design for the Multiobjective 0/1 Knapsack Problem

Yan-Yan Tan, Yong-Chang Jiao and Xin-Kuan Wang
National Key Laboratory of Antennas and Microwave Technology
Xidian University
Xi'an, China
Email: yytan@mail.xidian.edu.cn

***Abstract*—The 0/1 knapsack problem is a well known problem occurring in many real world problems. The problem is NP-complete. The multiobjective 0/1 knapsack problem is a generalization of the 0/1 knapsack problem in which multiple knapsacks are considered. A new version of MOEA/D for solving multiobjective 0/1 knapsack problem is proposed in this paper. This algorithm adopts the uniform design method to generate the aggregation coefficient vectors so that the decomposed scalar optimization subproblems are uniformly scattered, and therefore the algorithm could explore uniformly the region of interest from the initial iteration. Experimental results demonstrate that the proposed algorithm outperforms NSGA-II, SPEA2 and PESA significantly for the 2-objective, 3-objective and 4-objective benchmark knapsack problems.**

***Keywords*-knapsack problem; multiobjective evolutionary algorithm; uniform design; decomposition**

I. Introduction

The 0/1 knapsack problem is a widely studied problem due its practical importance. In the last years the generalization of this problem has been well studied and many algorithms for solving this variant have been proposed. Evolutionary approaches for solving the multiobjective 0/1 knapsack problem are of great interest. Many papers on the multiobjective knapsack problem (MOKP) and on the algorithms proposed for solving it can be found in the literature [1]–[7]. MOEA/D is a very recent multiobjective evolutionary algorithm (MOEA) using decomposition [8]. Up to present, it has been applied for solving a number of multiobjective optimization problems (MOPs) efficiently [8]–[12]. In this paper we propose a new version of MOEA/D with uniform design for multiobjective 0/1 knapsack problem. The proposed algorithm is experimentally compared with NSGA-II, SPEA2 and PESA on some benchmark knapsack problems.

The rest of the paper is organized as follows. In section II of the paper both single and multiobjective 0/1 knapsack problems are presented. The new proposed algorithm description is given in section III. The uniform design method for generating the aggregation coefficient vectors is also given in section III. Some comparisons with other algorithms are performed in section IV. Section V concludes this paper.

II. Problem Formulation

The classical 0/1 knapsack problem can be formulated as follow [2]. A set of n items and a knapsack of capacity c are considered. Each item has a profit p_j and a weight w_j. The problem is to select a subset of the items whose total weight does not exceed knapsack capacity c and whose total profit is maximum. Using the variables x_j (with $x_j = 1$ if the item j is selected and $x_j = 0$ otherwise) the problem can be written:

$$\begin{aligned} \text{maxmize } & \sum_{j=1}^{n} p_j x_j \\ \text{subject to } & \sum_{j=1}^{n} w_j x_j \leq c \\ & x_j \in \{0,1\}, j = \{1,\dots,n\}. \end{aligned} \tag{1}$$

The problem can be extended for an arbitrary number of knapsacks. Given a set of n items and a set of m knapsacks, the multiobjective 0/1 knapsack problem (MOKP) can be stated as:

$$\begin{aligned} \text{maxmize } & f_i(x) = \sum_{j=1}^{n} p_{ij} x_j, i = 1,\dots,m \\ \text{subject to } & \sum_{j=1}^{n} w_{ij} x_j \leq c_i, i = 1,\dots,m \\ & x = (x_1,\dots,x_n)^\top \in \{0,1\}^n \end{aligned} \tag{2}$$

where $p_{ij} \geq 0$ is the profit of item j in knapsack i, $w_{ij} \geq 0$ is the weight of item j in knapsack i, and c_i is the capacity of knapsack i. $x_i = 1$ means that item i is selected and put in all the knapsacks.

III. The Proposed Algorithm

A. Decomposition Strategies Used in the Proposed Algorithm

A multiobjective optimization problem (MOP) can be stated as follows:

$$\begin{aligned} \text{maxmize } & F(x) = (f_1(x),\dots,f_m(x))^\top \\ \text{subject to } & x \in \Omega \end{aligned} \tag{3}$$

MOEA/D decomposes an MOP (3) into a number of single objective optimization subproblems, and optimizes them

978-0-7695-4584-4/11 $26.00 © 2011 IEEE
DOI 10.1109/CIS.2011.29

simultaneously. The objective in each of these subproblems is an aggregation of all the objectives. Each subproblem is optimized by using information only from its neighboring subproblems. Neighborhood relations among subproblems are defined based on the distances between their aggregation coefficient vectors. The optimal solutions to two neighboring subproblems should be very similar.

There are several approaches for converting an MOP into a number of scalar optimization problems and they can be found in the literature (e.g., [13]). The most popular ones among them include the weighted sum approach and Tchebycheff approach which are introduced in the following:

- Weighted sum approach [8], [13]
  Let $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m)^\top$ be a weight vector, i.e., $\lambda_i \geq 0$ for all $i = 1, \ldots, m$ and $\sum_{i=1}^{m} \lambda_i = 1$. Then, the optimal solution to the following scalar optimization problem

$$\begin{aligned} \text{maxmize } g^{ws}(x|\lambda) &= \sum_{i=1}^{m} \lambda_i f_i(x) \\ \text{subject to } x &\in \Omega \end{aligned} \tag{4}$$

  is a Pareto optimal point to (3), where we use $g^{ws}(x|\lambda)$ to emphasize that λ is a coefficient vector in this objective function, while x is the variables to be optimized. To generate a set of different Pareto optimal vectors, one can use different weight vectors λ in the above scalar optimization problem. This approach considers a convex combination of the different objectives. So, if Pareto front (PF) is convex, this approach could work well. However, not every Pareto optimal vector can be obtained by this approach in the case of nonconvex PFs [13]. To overcome these shortcomings, Tchebycheff approach is suggested.
- Tchebycheff approach [8], [13]
  In this approach, the scalar optimization problem is in the form

$$\begin{aligned} \text{minimize } g^{te}(x|\lambda, z^*) &= \max_{1 \leq i \leq m} \{\lambda_i \, |f_i(x) - z_i^*|\} \\ \text{subject to } x &\in \Omega \end{aligned} \tag{5}$$

  where $z^* = (z_1^*, \ldots, z_m^*)^\top$ is the reference point, i.e., $z_i^* = \min_{1 \leq i \leq m} (f_i(x) | x \in \Omega)$ for each $i = 1, \ldots, m$. For each Pareto optimal point x^* there exists a weight vector λ such that x^* is the optimal solution of (5) and each optimal solution of (5) is a Pareto optimal solution of (3). Therefore, one is able to obtain different Pareto optimal solutions by altering the weight vector. One weakness with this approach is that its aggregation function is not smooth for a continuous MOP. However, since this work aims to solve knapsack problems which is a type of discrete problem, it still can be used in this paper.

B. Uniform Design Method for Generating the Aggregation Coefficient Vectors

This subsection introduces the so-called uniform design (UD) method that can be applied to generate the aggregation coefficient vectors of the decomposed subproblems for the proposed algorithm. The essence of the UD method is to find a set of points that are uniformly scattered over the design region of interest. Note that a 'uniformly scattered' set of points as stated here means roughly that the set has a small discrepancy, not a set of points which are uniformly distributed in the usual statistical sense. Let $\lambda^1, \ldots, \lambda^N$ be the aggregation coefficient vectors, subproblem j corresponds to the coefficient vector $\lambda^j = (\lambda_1^j, \ldots, \lambda_m^j)^\top$, $\lambda_i^j \geq 0, i = 1, \ldots, m$ and $\sum_{i=1}^{m} \lambda_i^j = 1$ for all $j = 1, \ldots, N$, m is the number of objectives, N is the population size (i.e., the number of aggregation coefficient vectors). The essence of the design is to find a set of points $\lambda^1, \ldots, \lambda^N$ which are uniformly scattered over the design region.

Without loss of generality, let the design space be the m-dimensional unit cube $C^m = [0,1]^m$, we represent any point in C^m by $x_i = (x_{i1}, \ldots, x_{im})^\top$, where $x_{i1}, \ldots, x_{im} \in [0,1]$. For a given positive integer N, a uniform design with N points in C^m is a collection of points $P^* = \{x_1^*, \ldots, x_N^*\} \subset C^m$ such that $M(P^*) = \min M(P)$, where the minimization is carried out over all $P = \{x_1, \ldots, x_N\} \subset C^m$ with respect to some measure of uniformity, M. There is more than one choice for M, for example, the star discrepancy, the most commonly used centered L_2-discrepancy (CD) and the wrap-around L_2-discrepancy (WD). The centered L_2-discrepancy (CD), denoted by $\mathrm{CD}_2(P)$ [14]:

$$\begin{aligned} \mathrm{CD}_2(P) = &\left(\frac{13}{12}\right)^m - \frac{2}{N}\sum_{k=1}^{N}\prod_{i=1}^{m}\left(1 + \frac{1}{2}\left|x_{ki} - \frac{1}{2}\right| \right. \\ &\left. - \frac{1}{2}\left|x_{ki} - \frac{1}{2}\right|^2\right) + \frac{1}{N^2}\sum_{k=1}^{N}\sum_{j=1}^{N}\prod_{i=1}^{m}\left(1 + \frac{1}{2}\right. \\ &\left.\left|x_{ki} - \frac{1}{2}\right| + \frac{1}{2}\left|x_{ji} - \frac{1}{2}\right| - \frac{1}{2}\left|x_{ki} - x_{ji}\right|\right) \end{aligned} \tag{6}$$

is used in our experiments, for it is convenient to compute and invariant under relabeling of coordinate axes. The CD is also invariant under reflection of points about any plane passing through the center and parallel to the faces of the unit cube C^m, that is, invariant when the i-th coordinate x_i is replaced by $1 - x_i$. Corresponding formulas for other discrepancies can be found in Fang et al. [14].

Now the question is how to determine the set of points with the lowest discrepancy. There are several methods for serving this purpose, good lattice point (glp) is adopted in our algorithm. Let $(N; h_1, \ldots, h_{m-1})$ be a vector with integral components satisfying $1 \leq h_i < N, h_i \neq h_j (i \neq j), m - 1 < N$ and the greatest common divisors $(N, h_i) = 1, i = 1, \ldots, m-1$. Let

$$u_{ki} \equiv k h_i (\mathrm{mod} N), k = 1, \ldots, N; i = 1, \ldots, m-1. \tag{7}$$

The set $S_N = \left\{u_k = (u_{k1}, \ldots, u_{km-1})^\top, k = 1, \ldots, N\right\}$ is called a lattice point set of the generating vector $(N; h_1, \ldots, h_{m-1})$. If the set S_N has the smallest discrepancy among all possible generating vectors, the set S_N is the uniform design that we need to construct.

For satisfying the restriction that $\sum_{i=1}^{m} \lambda_i^j = 1$ for all $j = 1, \ldots, N$, let $U = (u_{ki})_{k=1,\ldots,N; i=1,\ldots,m-1}$ be the uniform design constructed above. Calculate

$$c_{ki} = (u_{ki} - 0.5)/N \tag{8}$$

and

$$\begin{aligned} \lambda_1^k &= 1 - c_{k1}^{\frac{1}{m-1}} \\ \lambda_i^k &= \left(1 - c_{ki}^{\frac{1}{m-i}}\right) \prod_{j=1}^{i-1} c_{kj}^{\frac{1}{m-j}} \quad i = 2, \ldots, m-1 \\ \lambda_m^k &= \prod_{j=1}^{m-1} c_{kj}^{\frac{1}{m-j}} \end{aligned} \tag{9}$$

for each $k = 1, \ldots, N$. Then the uniformly scattered weight vectors $\{\lambda^k = (\lambda_1^k, \ldots, \lambda_m^k), k = 1, \ldots, N\}$ are achieved through the uniform design method.

C. Framework of the Proposed Algorithm

Uniform design multiobjective evolutionary algorithm based on decomposition (UMOEA/D) proposed in this paper is a new version of MOEA/D with uniform design for the multiobjective 0/1 knapsack problem. UMOEA/D needs to decompose the MOP under consideration. Any decomposition approaches can serve this purpose. In the following description, we suppose that the weighted sum approach is employed. It is very trivial to modify the following MOEA/D when other decomposition methods are used.

Let $\lambda^1, \ldots, \lambda^N$ be a set of uniformly scattered weight vectors. With the weighted sum approach, the objective function of the j-th subproblem is in the form [13]:

$$g^{ws}(x|\lambda^j) = \sum_{i=1}^{m} \lambda_i^j f_i(x) \tag{10}$$

where $\lambda^j = \left(\lambda_1^j, \ldots, \lambda_m^j\right)^\top$. UMOEA/D maxmizes all these N objective functions simultaneously in a single run. At each generation, UMOEA/D maintains the following:

- A population of N points $x^1, \ldots, x^N \in \{0,1\}^n$, where x^i is the current solution to the i-th subproblem;
- $FV^1, \ldots, FV^N$, where FV^i is the F-value of x^i, i.e., $FV^i = F(x^i)$ for each $i = 1, \ldots, N$;
- $z = (z_1, \ldots, z_m)^\top$, where z_i is the best value found so far for objective f_i;
- An external population (EP), which is used to store nondominated solutions found during the search.

Consequently, the general framework of UMOEA/D can be stated as follows:

Input:

- MOKP (2);
- a stopping criterion;
- N : the number of subproblems considered in UMOEA/D;
- $\lambda^1, \ldots, \lambda^N$: a uniform spread of N weight vectors;
- T : the number of the weight vectors in the neighborhood of each weight vector.

Output: EP

Step 1 Initialization:

Step 1.1: Set EP=ϕ.

Step 1.2: Compute the Euclidean distances between any two weight vectors and then work out the T closest weight vectors to each weight vector. For each $i = 1, \ldots, N$, set $B(i) = \{i_1, \ldots, i_T\}$, where $\lambda^{i_1}, \ldots, \lambda^{i_T}$ are the T closest weight vectors to λ^i.

Step 1.3: Generate an initial population $x^1, \ldots, x^N$ by randomly sampling from $\{0,1\}^n$. Apply a greedy repair method to the initial infeasible solutions. Set $FV^i = F(x^i)$.

Step 1.4: Initialize $z = (z_1, \ldots, z_m)^\top$ by setting $z_j = \max\limits_{1 \le i \le N} f_j\left(x^i\right), j = 1, \ldots, m$.

Step 2 Update:

For $i = 1, \ldots, N$, do:

Step 2.1 Reproduction: Randomly select two indexes k, l from $B(i)$, and then generate a new solution $\overline{y}$ from x^k and x^l by using genetic operators.

Step 2.2 Improvement: Apply a greedy repair heuristic on $\overline{y}$ to produce y.

Step 2.3 Update of z: For each $j = 1, \ldots, m$, if $z_j < f_j(y)$, then set $z_j = f_j(y)$.

Step 2.4 Update of neighboring solutions: For each index $j \in B(i)$, if $g^{ws}\left(y \left| \lambda^j\right.\right) \ge g^{ws}\left(x^j \left| \lambda^j\right.\right)$, then set $x^j = y$ and $FV^j = F(y)$.

Step 2.5 Update of EP:

Remove from EP all the vectors dominated by $F(y)$. Add $F(y)$ to EP if no vectors in EP dominate $F(y)$.

Step 3 Stopping criteria: If the stopping criterion is satisfied, then stop and output EP. Otherwise, go to **Step 2**.

The greedy repair method used in **Step 1.3** and **Step 2.2** is described as follows: Let set $J = \{j | 1 \le j \le n \bigwedge x_j = 1\}$ is the set of selected items and set $I = \left\{i | 1 \le i \le m \bigwedge \sum_{j=1}^{n} w_{ij} x_j > c_i\right\}$ is a set of overfilled knapsacks. Repeatedly select item $k \in J$ to remove until none of the knapsack is overfilled, such that:

$$k = \arg\min_{j \in J} \frac{g^{ws}(x) - g^{ws}(x^{j-})}{\sum\limits_{i \in I} w_{ij}} \tag{11}$$

where x^{j-} is different from x only by item j, i.e., $x_i^{j-} = x_i$ for all $i \neq j$ and $x_j^{j-} = 0$.

IV. Experimental Study

As test problems three instances of the knapsack problem are taken from [2], each with 750 items and 2, 3, and 4 objectives, respectively. For the random choice of the profit and weight values as well as the constraint handling technique we refer to the original study. The results obtained by UMOEA/D are compared with the results obtained by NSGA-II, SPEA2 and PESA. The population size N is set to 250 for $m = 2$, 300 for $m = 3$, and 350 for $m = 4$. All these four algorithms stop after $500 \times N$ calls of evaluation. For each algorithm and each problem, 30 runs with different random seeds have been carried out.

A. Performance Assessment

In order to compare the performance of the new proposed algorithm with other algorithms, two performance metrics are used: S-metric and C-metric.

- **Size of the space covered (S):** The S-metric [2], also known as the hypervolume metric, measures the size of the region which is dominated by the obtained Pareto front. Therefore the higher value of the S-metric is preferred. In low dimension, 2- and 3- objective spaces, it is known as area and volume respectively. The S-metric requires a reference point which must be dominated by all solutions of the Pareto front. The choice of the reference point $r = (r_1, \ldots, r_m)$ used in S-metric refers to [15]:

$$\begin{aligned} u_i &= \max_{x \in P}\{f_i(x)\} \\ l_i &= \min_{x \in P}\{f_i(x)\} \qquad (12) \\ r_i &= l_i - (u_i - l_i) \times 0.1 \end{aligned}$$

  u and l are referred as the upper bound and lower bound respectively for the solution set P obtained by all algorithms. In other words, u is the least vector that dominates all solutions in P in the objective space and l the most vector that is dominated by all solutions in P in the objective space.
- **Coverage of two sets (C):** The C-metric [2] measures the 'degree' of dominance of a Pareto front over another Pareto front. Let A and B be two approximations to the Pareto front of an MOP, $C(A, B)$ is defined as the percentage of the solutions in B that are dominated by at least one solution in A, i.e.,

$$C(A,B) = \frac{|\{u \in B | \exists v \in A : v \text{ dominates } u\}|}{|B|} \qquad (13)$$

  $C(A,B)$ is not necessarily equal to $1 - C(A,B)$. $C(A,B) = 1$ means that all solutions in B are dominated by some solutions in A, while $C(A,B) = 0$ implies that no solution in B is dominated by a solution in A.

Table I
Average Values of S-metric Based on 30 Independent Runs

Problem	UMOEA/D	NSGA-II	SPEA2	PESA
750-2	$6.8940 \cdot 10^7$	$4.7309 \cdot 10^7$	$4.9592 \cdot 10^7$	$4.6498 \cdot 10^7$
750-3	$7.3764 \cdot 10^{11}$	$3.2640 \cdot 10^{11}$	$3.3217 \cdot 10^{11}$	$3.0542 \cdot 10^{11}$
750-4	$9.1431 \cdot 10^{15}$	$2.4015 \cdot 10^{15}$	$2.4171 \cdot 10^{15}$	$2.0926 \cdot 10^{15}$

Table II
Average Values (Standard Deviation) of C-metric Based on 30 Independent Runs

Algorithm		C(A,B)		
A	B	750-2	750-3	750-4
UMOEA/D	NSGA-II	0.6507(0.217)	0.9834(0.034)	1(0)
	SPEA2	0.8577(0.164)	0.9587(0.053)	0.9984(0.002)
	PESA	0.2756(0.218)	0.6829(0.268)	0.9422(0.073)
NSGA-II	UMOEA/D	0.0909(0.063)	0.0002(0)	0(0)
SPEA2		0.0524(0.061)	0.0006(0.001)	0(0)
PESA		0.1584(0.063)	0.0079(0.008)	0.0001(0)

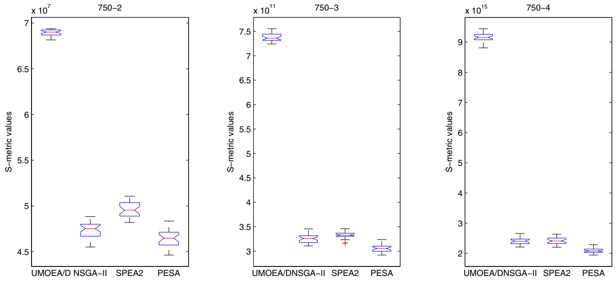

Figure 1. Box Plots of the S-metric Values Based on 30 Independent Runs

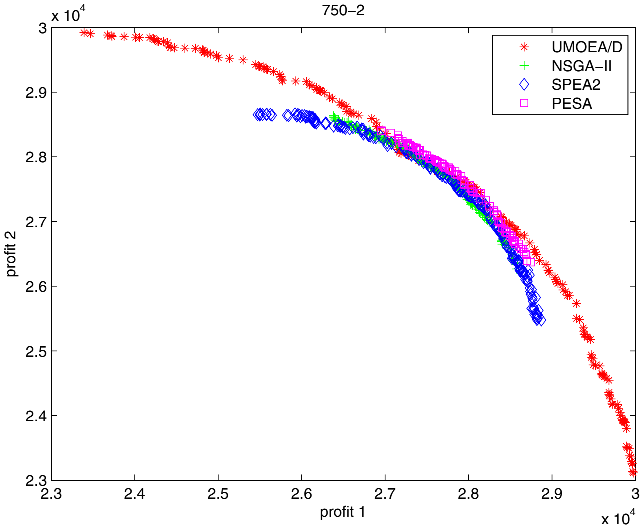

Figure 2. Plots of the Nondominated Solutions with the Largest S-metric in 30 Runs for the 2-objective MOKP Test Problem

B. Experimental Results

Table I presents the mean of the S-metric values in these four algorithms for each instance. Table II shows the mean and standard deviation of the C-metric values of the final approximations obtained by UMOEA/D and other three algorithms. The box plot of the S-metric values based on 30 independent runs among these four algorithms is visualized in Fig. 1. Fig. 2 plots the distributions of EP with the largest S-metric value found in each algorithm for the 2-objective instance.

It is evident from table I and Fig. 1 that the S-metric value of UMOEA/D is significantly larger than that of other three algorithms for each problem. These results indicate that UMOEA/D dominates more search space than NSGA-II, SPEA2 and PESA. Fig. 2 visually shows that UMOEA/D can find higher quality solutions than other three compared algorithms.

Table II shows that the final EP obtained by UMOEA/D is better than that obtained by NSGA-II, SPEA2 and PESA in terms of C-metric for all the test instances. Taking instance 750-3 as an example, on average, 98.34%, 95.87% and 68.29% of the final solutions generated by NSGA-II, SPEA2 and PESA respectively are dominated by those generated by UMOEA/D, and only 0.02%, 0.06% and 0.79% respectively vice versa. Overall, we can claim that UMOEA/D outperforms NSGA-II, SPEA2 and PESA on these test problems.

V. Conclusion

In this paper a new version of MOEA/D for solving multiobjective 0/1 knapsack problem is proposed. This algorithm adopts the uniform design method to generate the aggregation coefficient vectors so that the decomposed scalar optimization subproblems are uniformly scattered, and therefore the algorithm could explore uniformly the region of interest from the initial iteration. Several numerical experiments have been performed using several well-known instances of the knapsack problem. A comparison with NSGA-II, SPEA2 and PESA is also performed. Experimental results have shown that the proposed algorithm significantly outperforms the compared algorithms in all experiments.

References

[1] A. Jaszkiewicz, Comparison of local search-based metaheuristics on the multiple-objective knapsack problem, *Foundations of Computing and Decision Sciences*, 26(1):99-120, 2001.

[2] E. Zitzler and L. Thiele, Multiple objective evolutionary algorithms: a comparative case study and the strength Pareto approach, *IEEE Transactions on Evolutionary Computation*, 3(4):257-271, 1999.

[3] J. D. Knowles and D. W. Corne, A comparison of diverse approaches to memetic multi-objective combinatorial optimization, *Proceedings of the 2000 genetic and evolutionary computation conference workshop program*. Las Vegas, Nevada, 103-108, 2000.

[4] A. Jaszkiewicz, Genetic local search for multiple objective combinatorial optimization, *Technical Report RA-014/98*, Institute of Computing Science, Poznan University of Technology, 1998.

[5] E. L. Ulungu, J. Teghem, P. Fortemps and D. Tuyttens, MOSA method: a tool for solving multiobjective combinatorial optimization problems, *Journal of Multi-Criteria Decision Analysis*, 8:221-236, 1999.

[6] A. Jaszkiewicz, On the performance of multiple objective genetic local search on the 0/1 knapsack problemła comparative experiment, *IEEE Transactions on Evolutionary Computation*, 6(4):402-412, 2002.

[7] E. Zitzler, M. Laumanns and L. Thiele, SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization, *Proc. Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*, Athens, Greece, 95-100, 2001.

[8] Q. Zhang and H. Li, MOEA/D: A multi-objective evolutionary algorithm based on decomposition, *IEEE Trans. Evol. Comput.*, 11(6):712-731, 2007.

[9] H. Li and Q. Zhang, Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II, *IEEE Trans. Evol. Comput.*, 12(2):284-302, 2009.

[10] P. C. Chang, S. H. Chen, Q. Zhang and J. L. Lin, MOEA/D for flowshop scheduling problems, *Proc. of Congress on Evolutionary Computation (CEC'08)*, Hong Kong, China, 1433-1438, 2008.

[11] Q. Zhang, W. Liu, E. Tsang and B. Virginas, Expensive multiobjectiveoptimization by MOEA/D with gaussian process model, *Technical Report CES-489*, the School of Computer Science and Electronic Engineering, University of Essex, Tech. Rep., 2009.

[12] H. Ishibuchi, Y. Sakane, N. Tsukamoto and Y. Nojima, Adaptation of scalarizing functions in MOEA/D: An adaptive scalarizing functionbased multiobjective evolutionary algorithm, *Proc. of the 5th International Conference devoted to Evolutionary Multi-Criterion Optimization (EMO'09)*, Nantes, France, 438-452, 2009.

[13] K. Miettinen, *Nonlinear Multiobjective Optimization*. Boston: Kluwer Academic, 1999.

[14] K. T. Fang and D. K. J. Lin, Uniform designs and their application in industry, *Handbook of Statistics*, 22:131-170, 2003.

[15] Khoi Nguyen Le, BSc, MRes, *A study of evolutionary multiobjective algorithms and their application to knapsack and nurse scheduling problems*. PhD thesis, University of Nottingham, 2011.