WCCI 2010 IEEE World Congress on Computational Intelligence
July, 18-23, 2010 - CCIB, Barcelona, Spain CEC IEEE

MOEA/D with NBI-style Tchebycheff approach for Portfolio Management

Qingfu Zhang, Hui Li, Dietmar Maringer and Edward Tsang

***Abstract*— MOEA/D is a generic multiobjective evolutionary optimization algorithm. MOEA/D needs a approach to decompose a multiobjective optimization problem into a number of single objective optimization problems. The commonly-used weighted sum approach and the Tchebycheff approach may not be able to handle disparately scaled objectives. This paper suggests a new decomposition approach, called NBI-style Tchebycheff approach, for MOEA/D to deal with such objectives. A portfolio management MOP has been used as an example to test the effectiveness of MOEA/D with NBI-style Tchebycheff approach.**

I. Introduction

A multiobjective optimization problem (MOP) has more than one often conflicting objectives. No single solution can optimize these objectives at the same time. One has to balance these objectives. A solution to a multiobjective optimization is called Pareto optimal if any improvement in one objective must lead to deterioration to at least one other objective. Pareto optimal solutions are candidates for the best trade-off solution since there is no reason for choosing a nonPareto optimal solution which can be improved in at least one objective and not be deteriorated in any other objectives. There may be many, even infinitely many, Pareto solutions to a multiobjective optimization problem. A decision maker often requires a set of well representative Pareto optimal solutions for comparison before making their final decision.

A number of evolutionary algorithms have been proposed for multiobjective optimization problems over the last two decades [1][2]. The major advantage of these multiobjective evolutionary algorithms (MOEA) over other traditional methods are that they work with a population of candidate solutions and thus are able to produce a set of Pareto optimal solutions in a single run. The majority of the state-of-the-art MOEAs treat a MOP as a whole and use the Pareto dominance relationships among the solutions visited so far for identifying promising areas in the search space. These Pareto dominance based algorithms could drive its population towards the Pareto front. They, however, often fail to generate a set of solution uniformly distributed along the Pareto front since it is very different to allocate the computational resources to different parts of the Pareto fronts in a rational way. To overcome this shortcoming, we have recently proposed a simple and generic multiobjective evolutionary algorithm framework based on decomposition (MOEA/D) [3]. It decomposes a MOP into a number of scalar optimization subproblems. The optimal solutions to these subproblems are Pareto optimal to the MOP in question under some mild conditions. These solutions could provide a good approximation to the Pareto front if a proper decomposition scheme is used. MOEA/D optimizes these subproblems simultaneously by evolving a population of solutions. One of the key components in MOEA/D is its decomposition methods. Two commonly-used aggregation methods, i.e., the weighted Tchebycheff approach and the weighted sum approach, have been tried in MOEA/D. The major shortcoming of these two approach is that they are sensitive to scales of the objectives. One contribution of this paper is to propose a simple decomposition method, the NBI-style Tchebycheff approach, for MOEA/D for overcoming it.

A central task in financial management is to combine financial assets into a portfolio under some real-life constraints [4][5][6]. Often, an investor has to balance two conflicting objectives, namely, maximization of the expected return of the portfolio and minimization of its variance (i.e. risk). Therefore, a portfolio optimization problem by nature is biobjective and their two objectives are often of very different scales. In this paper, we apply MOEA/D with the NBI-style Tchebycheff approach on a portfolio management problem. Some effort has also been made to deal with the constraints. We have compared our algorithm with NSGA-II on this problem. Our experimental results show that our approach is a very promising tool for this biobjective portfolio management problem.

II. MOEA/D with the NBI-style Tchebycheff approach for Biobjective Optimization

In the following, we consider the following generic biobjective optimization problem:

$$\begin{aligned} \text{minimize} \quad & F(x) = (f_1(x), f_2(x)) \\ \text{subject to} \quad & x \in \Omega \end{aligned} \tag{1}$$

MOEA/D (multiobjective evolutionary algorithm based on decomposition) [3] is a simple and generic multiobjective evolutionary algorithm. It uses an aggregation method to decompose the MOP into N single objective optimization subproblems and solves these subproblems simultaneously (where N is a control parameter set by users). In MOEA/D, N procedures are employed and different procedures are

Q. Zhang is with School of Computer Science and Electronic Engineering, University of Essex, Wivenhoe Park, Colchester, CO4 3SQ, U.K (e-mail: qzhang@essex.ac.uk).

H. Li is with Department of Computer Science, University of Nottingham, Nottingham, NG8 1BB, UK. (e-mail: hui.li@nottingham.ac.uk).

D. Maringer is with Faculty for Economics and Management (WWZ), University of Basel, Switzerland. (email: dietmar.maringer@unibas.ch).

E. Tsang is with School of Computer Science and Electronic Engineering, University of Essex, Wivenhoe Park, Colchester, CO4 3SQ, U.K (e-mail: edward@essex.ac.uk).

used for solving different subproblems. A neighborhood relationship among all the subproblems (procedures) is defined based on the distances of their weight vectors. Neighboring subproblems should have similar fitness landscapes and optimal solutions. Therefore, neighboring procedures can speed up their searches by exchanging information. In a simple version of MOEA/D [3], each individual procedure keeps one solution in its memory, which could be the best solution found so far for its subproblems; it generates a new solution by performing genetic operators on several solutions from its neighboring procedures, and updates its memory if the new solution is better than old one for its subproblem. A procedure also passes its new generated solution on to some (or all) of its neighboring procedures, who will update their current solutions if the received solution is better. A major advantage of MOEA/D is that single objective local search can be used in each procedure in a natural way, since its task is to optimizie a single objective subproblem.

The weighted Tchebycheff approach and the weighted sum approach are two commonly-used strategies for decomposing an MOP into multiple single objective subproblems. The weighted sum approach can deal with convex MOPs but may fail in other MOPs. The weighted Tchebycheff approach can deal with non-convex MOPs. Both approaches are sensitive to the scales of objectives. In [7], a direction-based decomposition method, called the normal boundary intersection (NBI) approach, was proposed. It attempts to find the intersection points between the Pareto front and a number of straight lines, which are defined by a normal vector and a set of uniformly-distributed points in the convex hull of individual minima (CHIM). Compared with the weighted Tchebycheff approach and the weighted sum approach, the NBI approach is relatively insensitive to the scales of objective functions. However, NBI can not be easily used within MOEA/D since it introduces extra constraints.

In this paper, we take the advantages of both the NBI approach and the Tchebycheff approach for decomposition and propose a NBI-style Tchebycheff approach for decomposing the biobjecitive optimization (1). In the following, we explain how it works. Let $F^1 = (F_1^1, F_2^1)$ and $F^2 = (F_1^2, F_2^2)$ be the two extreme points of the PF of (1) in the objective space, N reference points $r^{(i)}$, $i = 1, \ldots, N$, are set to be N points which are evenly distributed along the line segment linking F^1 and F^2, i.e.

$$r^{(i)} = \alpha_i F^1 + (1 - \alpha_i) F^2 \quad (2)$$

where

$$\alpha_i = \frac{N - i}{N - 1}$$

for $i = 1, \ldots, N$. Then we can decompose (1) into N single objective subproblems. The i-th one is to minimize the following function:

$$g^{(tn)}(x|r^{(i)}, \lambda) = \max\{\lambda_1 (f_1(x) - r_1^{(i)}), \lambda_2 (f_2(x) - r_2^{(i)})\} \quad (3)$$

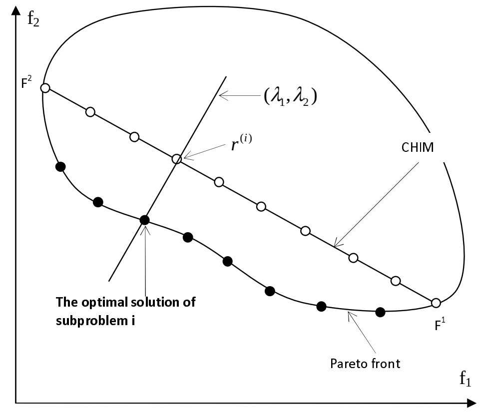

Fig. 1. Illustration of the distribution of $r^{(i)}$.

where λ_1 and λ_2 are

$$\lambda_1 = |F_2^2 - F_2^1| \quad (4)$$
$$\lambda_2 = |F_1^2 - F_1^1| \quad (5)$$

As shown in Fig. 1, the line segment F^1F^2 can be regarded as a linear approximation to the PF, and the direction defined by (λ_1, λ_2) is perpendicular to F^1F^2. The optimal solutions to the above subproblems can be evenly distributed along the PF. This decomposition requires the position of the two extreme points. In MOEA/D, these two points can be substituted by their approximates.

MOEA/D with the above NBI-style Tchebycheff decomposition is given as follows:

The algorithm aims at minimizing $g^{(tn)}(x|r^{(1)}, \lambda)$, …, $g^{(tn)}(x|r^{(N)}, \lambda)$ simultaneously. x^i is the current solution to the subproblem of minimization of $g^{(tn)}(x|r^{(i)}, \lambda)$. In Step 1.1, the neighborhood structure among subproblems are established. Since neighboring subproblems have close reference points, it is reasonable to assume that two neighboring subproblems have similar optimal solutions. Step 1.2 randomly select a solution x^i from the feasible search region to be the initial solution to subproblem i, $i = 1, \ldots, N$. Step 1.3 finds the smallest function value for each individual objective and then initialize two extreme points F^1 and F^2, which are needed to compute the objective function value of each subproblem. Step 2.1 is to generate a new solution y for subproblem i. The parents of y are current solutions to some neighbors of subproblem i. Therefore, it is very likely that y is a good solution to subproblem i and its neighbors. For this reason, y is used to update the current solutions to these subproblems in Step 2.3. Step 2.2 is to check if $F(y)$ can be a better approximate to the two extreme points of the CHIM.

It should be pointed out that several other improvements on MOEA/D have been made very recently. Li and Zhang suggested using two different neighborhood structures for balancing exploitation and exploration [8]. Zhang et al [9] proposed a scheme for dynamically allocating computational

Input

- Biobjective Optimization Problem (1);
- A stopping criterion;
- N the number of the subproblems;
- T: The neighborhood size.

Output Approximation to the PF: $F(x^1), \ldots, F(x^N)$.

Step 1 Initialization

Step 1.1 Compute neighborhood
For each $i = 1, \ldots, N$, set $B(i) = \{i_1, \ldots, i_T\}$ where $r_{i_1}, \ldots, r_{i_T}$ are the T closest values to r_i.

Step 1.2 Initialize population
Generate $x^1, \ldots, x^N$ randomly from the feasible search region. Set $FV^i = F(x^i)$.

Step 1.3 Estimate the extreme points of CHIM:
Set F^k to be the point among $\{FV^1, \ldots, FV^N\}$ with the smallest f_k function value, $k = 1, 2$.

Step 2 Update

For each $i \in \{1, \ldots, N\}$, do

Step 2.1: Reproduction:
Randomly choose several distinct indexes $p_1, \ldots, p_j$ from $B(i)$, perform a genetic operator on $x^{p_1}, \ldots, x^{p_j}$ to generate a new solution y. Repair y if it is an infeasible solution.

Step 2.2 Re-estimate of the extreme points in CHIM:
For each $k = 1, 2$, if $f_k(y)$ is the smaller that the f_k value in F^k, then replace F^k by $F(y)$.

Step 2.3 Update neighboring solutions:
For each index $j \in B(i)$, if $g^{(tn)}(y|r^{(i)}, \lambda) \leq g^{(tn)}(x^j|r^{(j)}, \lambda)$, then set $x^j = y$ and $FV^j = F(y)$.

Step 3 Stopping condition

If the stopping criteria is satisfied, then stop and output $F(x^1), \ldots, F(x^N)$. Otherwise, go to **Step 2**.

Fig. 2. MOEA/D with NBI-style Tchebycheff approach

effort to different procedures in MOEA/D in order to reduce the overall cost and improve the algorithm's performance; this implementation of MOEA/D is efficient and effective and has won the CEC'09 MOEA competition. Nebro and Durillo developed a thread-based parallel version of MOEA/D [10], which can be executed on multi-core computers. Palmers et al. proposed an implementation of MOEA/D in which each procedure record more than one solutions [11]. Ishibuchi et al. proposed using different aggregation functions at different search stages [12]. The work in this paper represents an attempt to develop a simple and yet efficient decomposition method in the framework of MOEA/D.

III. Constrained Biobjective Portfolio Optimization Problem

We consider the following constrained biobjective portfolio optimization problem:
Given

- W: the budget.
- n: the number of the assets available.
- c_i: the unit price of asset i.
- r_i: the expected return rate of asset i.
- σ_{ij}: the covariance between assets i and j.
- $g_i : N \rightarrow R^+$: a cost function for investing in asset i. i.e., the overall cost for the purchase of k units of asset i is $g_i(k)$. A major factor associated with this function is the transaction cost.
- r_s: the safe rate for investing in a settlement account.
- K: the maximal number of assets allowed to invest.

The decision variables are

- x_i: the amount of money used for buying asset i.

The constraints are

- x_i is the exact cost for buying an integer number of units of asset i. i.e., there exists an integer k_i such that

$$x_i = g_i(k_i) \quad (6)$$

- The overall money spent cannot excess the budget.

$$\sum_{i=1}^{n} x_i \leq W \quad (7)$$

- The total number of assets to invest, i.e., the number of x_i with positive values, can not be larger than K.

The goal is to maximize the expected return rate:

$$R = \frac{1}{W}[(W - \sum_{i=1}^{n} x_i)(1 + r_s) + \sum_{i=1}^{n}(1 + r_i)k_i c_i] - 1 \quad (8)$$

and minimize the variance:

$$V = \sum_{i=1}^{n}\sum_{i=1}^{n}(k_i c_i)(k_j c_j)\sigma_{ij} \quad (9)$$

where $(1 + r_i)k_i c_i$ in R is the return generated by k_i units of asset i, and $(W - \sum_{i=1}^{n} x_i)(1 + r_s)$ is the return by investing the remainder in a settlement account. $k_i c_i$ in V is the actual volume of asset i in the portfolio.

This problem, first presented in [13], was among the first attempts to investigate transaction costs and integer constraints at the same time. Like other portfolio optimization problems with real-life constraints, this problem cannot be solved directly by traditional quadratic programming methods since it involves integer variables. Very often, the two objectives are with very difference scales; the risk is often smaller than 1 while the return can be in the scale of 10,000 in our test instances.

This is the major reason why we use MOEA/D with the NBI-style Tchebycheff approach for solving this problem.

IV. Simulation Results

A. Test Instances

8 test instances have been constructed. The characteristics of these instances are given in Table I. The data in these instances (i.e., price of the unit of each stock and daily returns of 1000 days) is based on the German stock index DAX, which is available from the authors upon request. CPO1-4 are

TABLE I

PORTFOLIO OPTIMIZATION TEST INSTANCES

Instance Name	n	K	Scale
CPO1	30	30	small
CPO2	50	50	small
CPO3	100	100	large
CPO4	150	150	large
CPO5	30	10	small
CPO6	50	10	small
CPO7	100	30	large
CPO8	150	30	large

four instances with $K = n$, i.e., they have no constraint on the number of the assets to invest. In the other four instances $K < n$. Four instances are of small size, i.e., $n \leq 50$ and the other four ones are of large scale with $n \geq 100$.

we have considered the following three types of transaction costs in defining the $g(\cdot)$:

- zero transaction cost;
- fixed transaction cost; which is 50 pounds for each asset.
- proportional cost, the cost is 1% of the values of the assets.

The initial budget is set to be 50,000 pounds. The safe rate of investing money in settlement account r_s is 2.5%.

B. Experimental Settings

In MOEA/D with the NBI-style Tchebycheff approach, the setting is as follows:

1) Reproduction Operator in Step 2.1: Differential evolution (DE) mutation is used in this paper. Three distinct indexes p_1, p_2, p_3 are randomly selected from $B(i)$, y is created in the following:

$$y_i = x_i^{p_1} + F \cdot (x_i^{p_2} - x_i^{p_3}) \tag{10}$$

where F is a control parameter, $F = 0.5$ in our experiments.

2) Repair Operator: If a solution y violates the constraints, we repair it with the following procedure:

1. If any component of y is negative, set it to be zero.
2. If the number of nonzero components in y is larger than K, then randomly select and set some components to zero such that the number of nonzero components in y is K.
3. For each component y_i in y, set $y_i = \frac{y_i}{\sum_{i=1}^{n} y_i} W$.
4. For each component y_i in y, find k_i such that $g(k_i) \leq y_i < g(k_i + 1)$ and then set $y_i = g(k_i)$.
5. Randomly choose i with $y_i > 0$ and then invest $W - \sum_{i=1}^{n} y_i$ on asset i as much as possible.

3) Parameter Settings:

- Population size $N = 50$ for small scale instances, and $N = 100$ for large scale instances
- $T = N/2$ in MOEA/D.
- The algorithm stops after 100 generations.

20 runs were made on each algorithm on each instance.

4) Algorithm in Comparison: For comparison, we also tested NSGA-II on these instances, the settings of population size and the stopping condition are the same as in MOEA/D. NSGA-II also used the same reproduction operators.

C. Performance Metrics

In our experiments, the following performance indexes are used.

- **Set Coverage (C-metric):** Let A and B be two approximations to the PF of a 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.,

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

  $C(A, B)$ is not necessarily equal to $1 - C(B, A)$. $C(A, B) = 1$ 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.
- **Inverted Generational Distance (IGD-metric):** Let P^* be a set of well representative points along the PF. Let A be an approximation to the PF, the average distance from P^* to A is defined as:

$$D(A, P^*) = \frac{\sum_{v \in P^*} d(v, A)}{|P^*|}$$

  where $d(v, A)$ is the minimum Euclidean distance between v and the points in A. If $|P^*|$ is large enough to represent the PF very well, $D(A, P^*)$ could measure both the diversity and convergence of A in a sense. To have a low value of $D(A, P^*)$. The set A must be very close to the PF and cannot miss any part of the whole PF.

In our experiments, we do not know the actual PF. For each instances, we set P^* to be the set of non-dominated solutions obtained from all the runs of two algorithms.

D. Experimental Results

1) Results in the case of zero transaction cost: Tables II and III show the mean and standard deviation (std) values of C-metric and IGD-metric found by MOEA/D and NSGA-II on the instances with zero transaction cost. From these results, it is clear that MOEA/D performs better than NSGA-II on all the instances except CPO7. However, neither of these two algorithms can find solutions dominating more than 50% of those found by the other. The non-dominated fronts found by MOEA/D and NSGA-II in the best run with the smallest IGD value among 20 runs on each instance are plotted in Fig. 3 and 4. Fig. 3 clearly shows that the approximations obtained by MOEA/D are better than those by NSGA-II in CPO1-4. Fig .4 suggests that there is not much difference visually between the best approximations obtained by two algorithms for CPO5-8.

2) Results in the case of fixed transaction cost: Tables IV and V present the mean and std values of C-metric and IGD-metric found by MOEA/D and NSGA-II on CPO1-4 with fixed transaction costs. It can be observed from Table IV that, in terms of C-metric, MOEA/D performs worse than NSGA-II on two small instances - CPO1 and CPO2, but better on two large instances - CPO3 and CPO4. The results

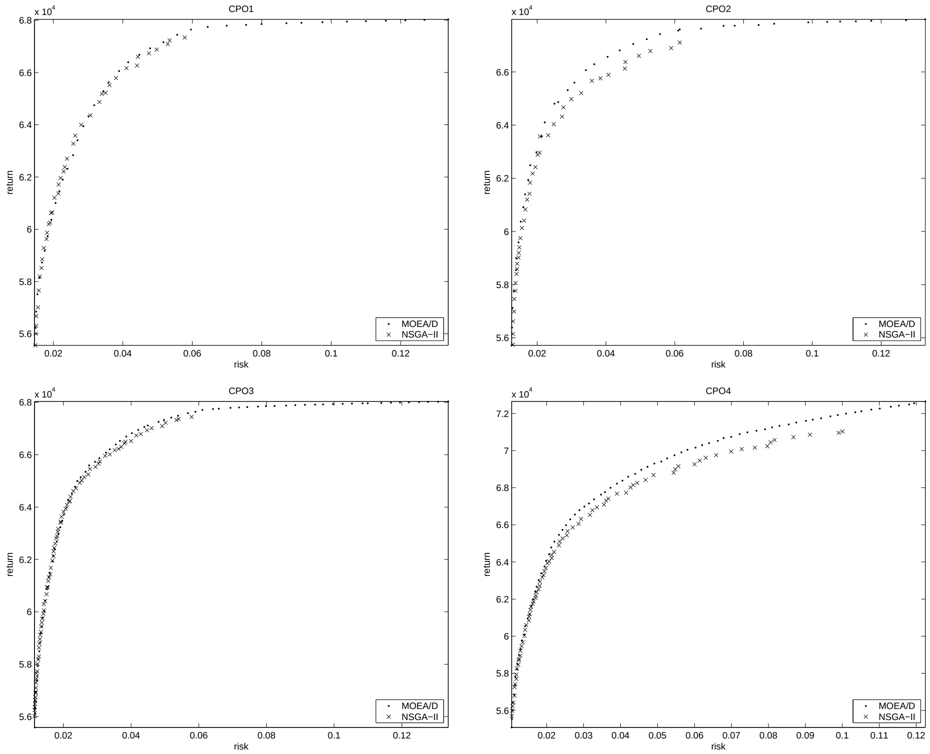

Fig. 3. Plots of the best approximations in terms of the IGD-metric found by MOEA/D and NSGA-II for CPO1-CPO4 with zero transaction cost

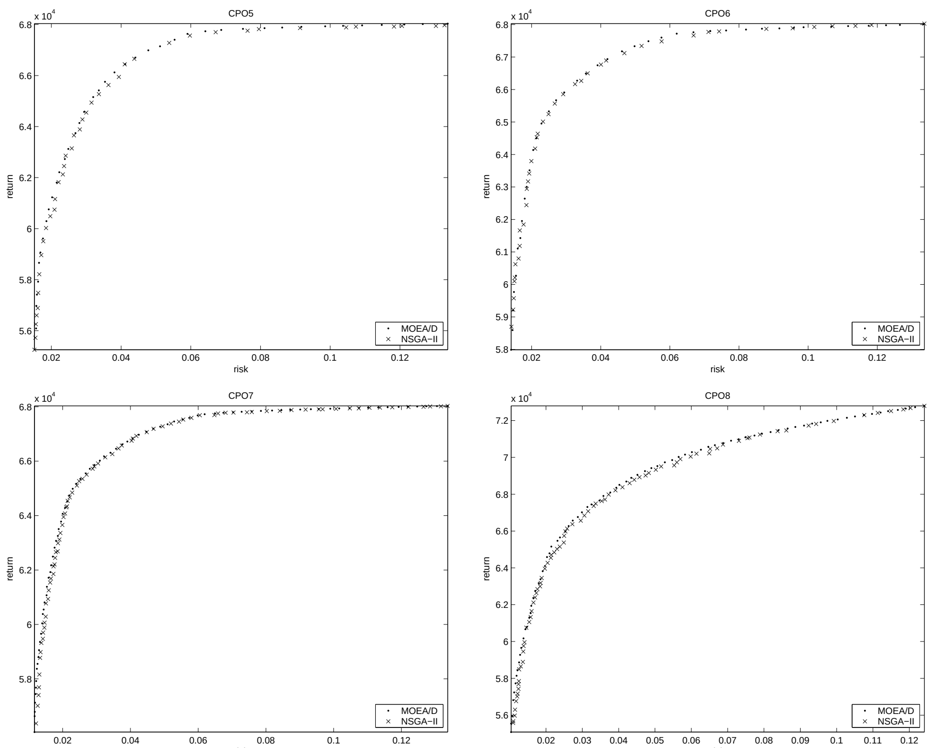

Fig. 4. Plots of the best approximations in terms of the IGD-metric found for CPO5-CPO8 with zero transaction cost

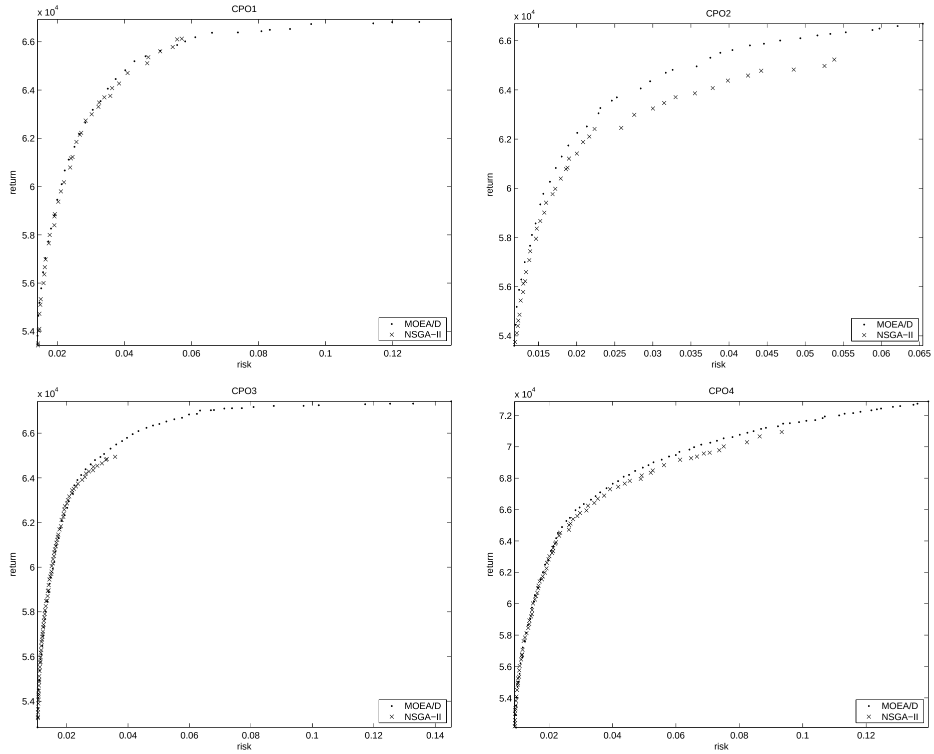

Fig. 5. Plots of the best approximations in terms of the IGD-metric by MOEA/D and NSGA-II for CPO1-4 with fixed transaction cost

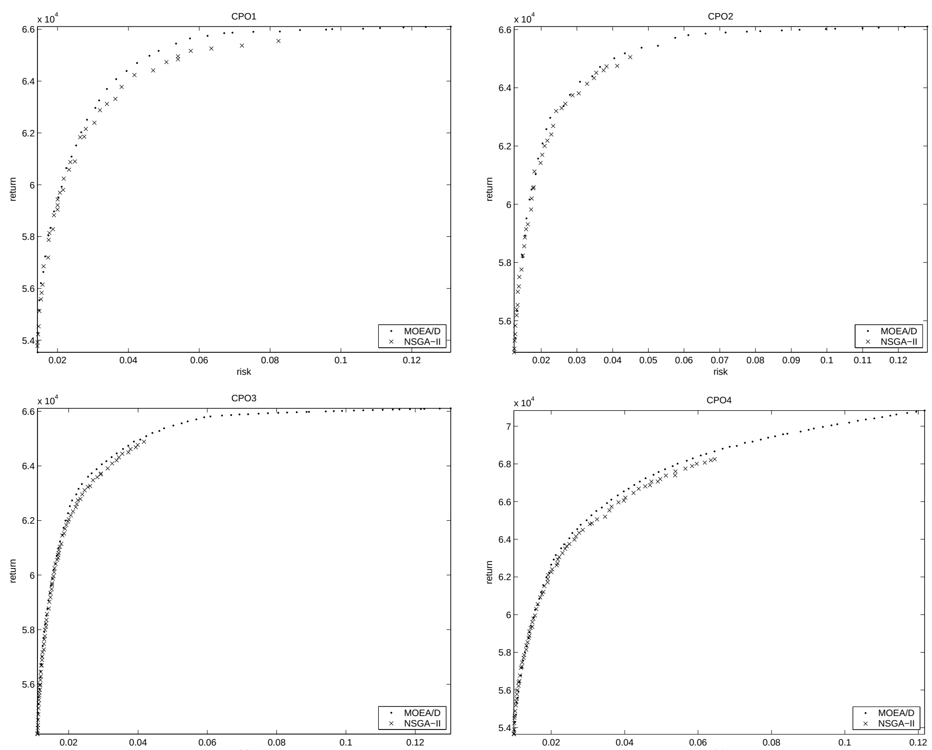

Fig. 6. Plots of the best approximations in terms of the IGD-metric by MOEA/D and NSGA-II for CPO1-4 with proportional transaction cost

TABLE II

C-METRIC OF MOEA/D AND NSGA-II ON CPO1-8 WITH ZERO TRANSACTION COST

C-metric	C(MOEA/D,NSGA-II)	C(NSGA-II, MOEA/D)
CPO1	11.50%	29.95%
CPO2	16.49%	28.63%
CPO3	19.25%	29.73%
CPO4	20.19%	40.74%
CPO5	14.06%	25.95%
CPO6	15.22%	26.31%
CPO7	37.11%	25.14%
CPO8	25.36%	29.76%

TABLE III

IGD-METRIC OF MOEA/D AND NSGA-II ON CPO1-8 WITH ZERO TRANSACTION COST

IGD-metric	MOEA/D	NSGA-II
CPO1	0.0066(0.0015)	0.0087(0.0010)
CPO2	0.0042(0.0012)	0.0083(0.0032)
CPO3	0.0030(0.0008)	0.0091(0.0032)
CPO4	0.0057(0.0023)	0.0125(0.0041)
CPO5	0.0025(0.0027)	0.0074(0.0062)
CPO6	0.0024(0.0025)	0.0075(0.0072)
CPO7	0.0058(0.0016)	0.0053(0.0012)
CPO8	0.0059(0.0026)	0.0189(0.0143)

in Table V indicates that MOEA/D outperforms NSGA-II on all four instances in terms of the IGD value. Fig. 5 shows the best approximations produced by MOEA/D are much better than those by NSGA-II on CPO2-4, and the two algorithms performs very similarly on CPO1.

3) Results in the case of proportional transaction cost: Tables VI and VII give the mean and std values of C-metric and IGD-metric found by MOEA/D and NSGA-II on CPO1-4 with proportional transaction cost. Table VI shows that MOEA/D performs better than NSGA-II on CPO2-4 and poorer on CPO1. Table VII show that MOEA/D performs better in terms of the ICD-metric on four instances. Fig. 6 shows the the best approximations obtained by MOEA/D are much better than those by NSGA-II on all the four instances.

TABLE IV

C-METRIC OF MOEA/D AND NSGA-II ON CPO1-4 WITH FIXED TRANSACTION COST

C-metric	C(MOEA/D,NSGA-II)	C(NSGA-II, MOEA/D)
CPO1	18.63%	10.87%
CPO2	19.24%	9.92%
CPO3	22.50%	36.68%
CPO4	17.55%	58.74%

TABLE V

IGD-METRIC OF MOEA/D AND NSGA-II ON CPO1-4 WITH FIXED TRANSACTION COST

IGD-metric	MOEA/D	NSGA-II
CPO1	0.0021 (0.0004)	0.0057 (0.0020)
CPO2	0.0039 (0.0018)	0.0161 (0.0072)
CPO3	0.0119 (0.0061)	0.0179 (0.0043)
CPO4	0.0092 (0.0032)	0.0154 (0.0058)

TABLE VI

C-METRIC OF MOEA/D AND NSGA-II ON CPO1-4 WITH PROPORTIONAL TRANSACTION COST

C-metric	C(MOEA/D,NSGA-II)	C(NSGA-II, MOEA/D)
CPO1	20.20%	10.27%
CPO2	15.55%	21.14%
CPO3	18.02%	42.17%
CPO4	16.56%	60.45%

TABLE VII

IGD-METRIC OF MOEA/D AND NSGA-II ON CPO1-4 WITH PROPORTIONAL TRANSACTION COST

IGD-metric	MOEA/D	NSGA-II
CPO1	0.0023(0.0011)	0.0054 (0.0005)
CPO2	0.0034 (0.0016)	0.0153 (0.0041)
CPO3	0.0032 (0.0011)	0.0088 (0.0022)
CPO4	0.0036 (0.0012)	0.0075 (0.0028)

V. CONCLUSION

In this paper, we proposed the NBI-style Tchebycheff approach for MOEA/D. In this approach, all the subproblems have the same weight vector. Their reference points are evenly distributed along the CHIM. MOEA/D with the NBI-style Tchebycheff approach can deal with MOPs with disparately scaled objectives. We tested this version of MOEA/D on a biobjective optimization problem for portfolio management, which has disparately scaled objectives. Our experimental results have shown that it is very promising.

In the future, we aim to investigate and improve the performances of MOEA/D with the NBI-style Tchebycheff approach on other complicated MOPs.

REFERENCES

[1] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. Chichester: Wiley, 2001.

[2] C. A. C. Coello, G. Lamont, and D. A. V. Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*. New York: Springer, 2007.

[3] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007.

[4] E. J. Elton and M. J. Gruber, *Modern Portfolio Theory and Investment Analysis*. New York: Wiley, 1995.

[5] H. M. Markowitz, *Portfolio Selection: Efficient Diversification of Investments*. Oxford: Blackwell, 1991.

[6] S. A. Zenios, *Financial Optimization*. Cambridge University Press, 1993.

[7] I. Das and J. E. Dennis, "Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems," *SIAM J. on Optimization*, vol. 8, no. 3, pp. 631–657, 1998.

[8] H. Li and Q. Zhang, "Multiobjective Optimization Problems with Complicated Pareto Sets, MOEA/D and NSGA-II," *IEEE Trans. on Evolutionary Computation*, vol. 12, no. 2, pp. 284–302, 2009.

[9] Q. Zhang, W. Liu, and H. Li, "The performance of a new version of moea/d on cec09 unconstrained mop test instances," in *IEEE Congress on Evolutionary Computation (CEC '09)*, 18-21 May 2009, pp. 203 – 208.

[10] J. Durillo and A. J. Nebro, "A study on parallelization of the multiobjective metaheurstics MOEA/D," in *LION 2010*, 2010.

[11] P. Palmers, T. McConnaghy, M. Steyaert, and G. Gielen, "Massively multitopology sizing of analog integrated circuits," in *DATE2009*, 2009, pp. 706–711.

[12] H. Ishibuchi, Y. Sakane, N. Tsukamoto, and Y. Nojima, “Adaptation of scalarizing functions in MOEA/D: An adaptive scalarizing function-based multiobjective evolutionary algorithm,” in *EMO 2009*, 2009, pp. 438–452.
[13] D. Maringer, “portfolioselektion bei transaktionskosten und ganzzahligkeitsbeschrnkungen,” *Zeitschrift fr Betriebswirtschaft*, vol. 72, no. 11, pp. 1155–1176, 2002.