Knowledge-Based Systems 107 (2016) 271–288

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Multiobjective differential evolution algorithm based on decomposition for a type of multiobjective bilevel programming problems

Hong Li[a,*], Qingfu Zhang[b], Qin Chen[c], Li Zhang[a], Yong-Chang Jiao[d]

[a] School of Mathematics and Statistics, Xidian University, Xi'an 710071, Shaanxi, China
[b] School of Computer Science and Electronic Engineering, University of Essex, Colchester CO4 3SQ, UK
[c] Facility Design and Instrument Institute, China Aerodynamic Research & Development Center, Mianyang 621000, Sichuan, China
[d] School of Electronic Engineering, Xidian University, Xi'an 710071, Shaanxi, China

ARTICLE INFO

Article history:
Received 6 December 2015
Revised 9 June 2016
Accepted 13 June 2016
Available online 18 June 2016

Keywords:
Multiobjective bilevel programming problem
Pareto optimality
Weighted sum approach
Tchebycheff approach
MOEA/D
NSGA-II

ABSTRACT

This paper considers the multiobjective bilevel programming problem (MOBLPP) with multiple objective functions at the upper level and a single objective function at the lower level. By adopting the Karush-Kuhn-Tucker (KKT) optimality conditions to the lower level optimization, the original multiobjective bilevel problem can be transformed into a multiobjective single-level optimization problem involving the complementarity constraints. In order to handle the complementarity constraints, an existing smoothing technique for mathematical programs with equilibrium constraints is applied. Thus, a multiobjective single-level nonlinear programming problem is formalized. For solving this multiobjective single-level optimization problem, the scalarization approaches based on weighted sum approach and Tchebycheff approach are used respectively, and a constrained multiobjective differential evolution algorithm based on decomposition is presented. Some illustrative numerical examples including linear and nonlinear versions of MOBLPPs with multiple objectives at the upper level are tested to show the effectiveness of the proposed approach. Besides, NSGA-II is utilized to solve this multiobjective single-level optimization model. The comparative results among weighted sum approach, Tchebycheff approach, and NSGA-II are provided.

1. Introduction

Bilevel programming problem (BLPP) is a complicated mathematical model with a hierarchical structure involving two decision makers in the decision process [1]. BLPPs have a wide domain of applications, particularly in urban traffic and transportation, resource assignment, supply chain planning, structural optimization, engineering design, game playing strategies, and others. For example, Chiou [2] established a bilevel model with link capacity expansion for a normative road network design with uncertain travel demand in order to simultaneously reduce travel delay to road users and mitigate the vulnerability of the road network. Dempe et al. [3] developed a linear bilevel model for a natural gas cash-out problem between a natural gas shipping company and a pipeline operator. A penalty function method was proposed to solve the model. Hesamzadeh and Yazdani [4] proposed a mixed-integer linear bilevel model with multi-follower for transmission planning in an imperfect competition environment of the electricity supply industry, and the model was solved using Kuhn-Tucker optimality conditions and a binary mapping approach. Gao et al. [5] proposed two nonlinear bilevel pricing models for pricing problems between the vendor and the buyer in a two-echelon supply chain. A PSO-based algorithm was developed to solve these bilevel pricing models. Lots of instances in application have been summarized in [15–21].

In view of the fact that the applications of BLPPs are more and more extensive and diverse, efficient solution strategies are of critical importance for solving these BLPP models. Till now, many studies on solution strategies including classical methods and heuristic algorithms have been done for all types of BLPPs. Especially, a variety of heuristic algorithms has been employed to solve BLPPs successfully [6–14], which have numerous advantages, such as simplicity, efficiency, flexibility and robustness. Compared with classical methods, heuristic algorithms are suitable for either large-scale BLPPs or BLPPs with weak features. The reviews, monographs, and surveys on the models, algorithms and applications of BLPPs may refer to [1,15–21].

* Corresponding author.
E-mail address: lihong@mail.xidian.edu.cn (H. Li).

http://dx.doi.org/10.1016/j.knosys.2016.06.018
0950-7051/

Table 1
The average CPU time (in seconds) used by MOEA/D-DE with weighted sum approach and MOEA/D-DE with Tchebycheff approach.

Instance	Weighted sum approach	Tchebycheff approach
1	288.8705	269.1355
2	290.4008	305.5410
3	1153.4208	1169.1095
4	1514.5671	1652.7553
5	316.4801	381.0618
6	344.4778	256.3112
7	487.5152	579.9305
8	235.3970	219.2462
9	608.9422	550.5114
10	674.3858	691.0868
11	1060.9020	1102.1321

Table 2
The *C*-metric values between MOEA/D with weighted sum approach (*A*) and Tchebycheff approach (*B*). Mean denotes the mean value of *C*-metric values, and SD means the standard deviation of *C*-metric values in ten independent runs.

Instance	$C(A, B)$		$C(B, A)$	
	Mean	SD	Mean	SD
1	0.0099	0.0072	0.0251	0.0449
2	0	0	0	0
3	0.0106	0.1979	0.0053	0.0107
4	0.0078	0.0068	0.0086	0.0055
5	0.0040	0.0034	0.0043	0.0035
6	0	0	0.0205	0.0649
7	0.0120	0.0400	0.0205	0.0679
8	0.0517	0.0182	0.0788	0.0148
9	0.0503	0.0214	0.0768	0.0154
10	0.0009	0.0008	0	0
11	0.0022	0.0034	0.0702	0.0433

Multiobjective bilevel programming problem (MOBLPP) involving multiple objectives either at a certain level or at both levels has great significance in application, for example transportation system planning and management [22], network flow problem in a large-scale construction project [23]. However, in contrast with the vast literature on the BLPPs, little work has been conducted on MOBLPPs, either in algorithm or in application [25,26]. MOBLPPs can be classified into three categories: 1) MOBLPP with multiple objectives at the upper level [24–27], 2) MOBLPP with multiple objectives at the lower level [28,29], and 3) MOBLPP with multiple objectives at both levels [30–36]. Such multiobjective bilevel models are difficult to solve due to their intrinsic nonconvexity and many objectives even in one level. This paper centers on the solution methodology for MOBLPP in first category.

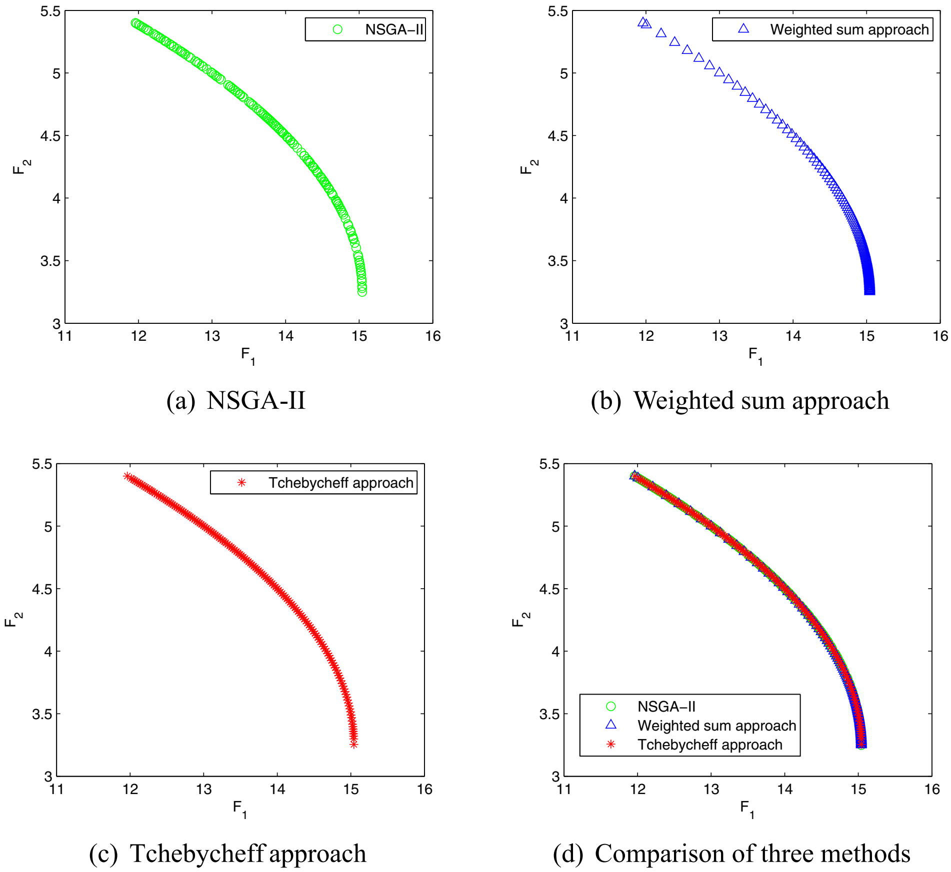
With respect to some recent studies on MOBLPP with multiple objectives at the upper level, most of the work focused on linear MOBLPPs. Ye [24] derived necessary optimality conditions by considering a combined problem, with both the value function and the Karush-Kuhn-Tucker (KKT) conditions of the lower-level problem involved in the constraints. Alves [25] proposed a multiobjective particle swarm optimization (MOPSO) algorithm to solve linear multiobjective bilevel programming problems with multiple objectives at the upper level. In MOPSO algorithm, each particle of the swarm is composed by two different parts, i.e. the upper level variable updated according to the principles of PSO, and the lower level variable given afterwards through the resolution of the lower-level optimization problem for the fixed upper level variable. Alves, Dempe, and Júdice [26] analyzed the linear bilevel programming problem with bi-objective on the upper level and a single objective at the lower level. However the problem considered in the paper has no lower level variables in the upper level constraints. The original problem was reformulated as a multiobjective mixed 0-1 linear programming problem. An existing interactive reference point procedure for multiobjective mixed-integer linear programming was employed to compute Pareto optimal solutions to the original problem. Calvete and Galé [27] considered general bilevel problems with many objectives at the upper level, when all objective functions are linear and constraints at both levels define polyhedra. This problem can be reformulated as a multiobjective problem with linear objective functions over a feasible region which is implicitly defined by a linear optimization problem and, in general, is non-convex. The weighted sum scalarization methods and scalarization methods were used to obtain efficient solutions. In above-mentioned literature, the solution algorithms were proposed only for linear version of MOBLPP with multiple objectives at the upper level. The aim of this paper is the development of solution methodology for both linear and nonlinear versions of MOBLPPs with multiple objectives at the upper level.

Regarding solution methodology of MOBLPP with multiple objectives at the upper level, there exist two ways to be chosen. One way is to transform the two-level structure to a single-level formation by adopting the optimality conditions or other techniques to the lower level optimization problem, and then utilize multiobjective evolutionary algorithms (MOEAs) or scalarization approaches to solve the single-level transformation model. For example, the way was used in [26]. The other way is to keep the two-level structure of original problem, and then apply MOEAs or scalarization approaches to the upper level optimization, while use the classical optimization techniques or heuristic algorithms to solve interactively the lower level optimization for each given upper level variable. For example, the way was used in [25]. The first way has a good efficiency, yet the lower level optimization must satisfy a certain optimality. In contrast, the second way is time-consuming, but the lower level optimization may have weak property. This paper aims at the first way in designing the algorithm for MOBLPP with multiple objectives at the upper level.

Based on above consideration, a solution approach for MOBLPP with multiple objectives at the upper level is presented. When the KKT optimality conditions are satisfied for the lower level optimization, the original multiobjective bilevel formulation can be converted into a multiobjective single-level nonlinear optimization problem with the complementarity constraints. Subsequently, an existing smoothing technique is applied to deal with the complementarity constraints. Thus, a constrained multiobjective single-level nonlinear optimization problem is formalized. For solving the reformulation of the original problem as a constrained multiobjective single-level programming problem, the scalarization approaches based on weighted sum approach and Tchebycheff approach are used respectively, and a constrained multiobjective differential evolution algorithm based on decomposition is presented, which is a modification to MOEA/D [39]. In addition, NSGA-II [41] is also utilized to solve reformulation of the original problem. By comparison of different MOEAs, we try to find which MOEA is more suitable for the reformulation of MOBLPP.

The main contributions of this work can be summarized as follows.

(1) The transformation model of MOBLPP is constructed to reduce its computational complexity. A multiobjective single level optimization is formed by using the KKT optimality conditions in the lower level programming, and then adopting the smoothing technique for the complementarity constraints.
(2) A constrained multiobjective differential evolution algorithm based on decomposition is developed for solving the transformation model of MOBLPP. For obtaining a uniform distribution of solutions in objective space, an adaptive weight



(a) NSGA-II (b) Weighted sum approach

(c) Tchebycheff approach (d) Comparison of three methods

Fig. 1. The Pareto front produced by three approaches for Problem 1.

Table 3
The mean value (Mean) and standard deviation (SD) of *S*-metric values obtained by MOEA/D-DE with weighted sum approach and MOEA/D-DE with Tchebycheff approach.

Instance	Weighted sum approach		Tchebycheff approach	
	Mean	SD	Mean	SD
1	0.0548	0	0.0049	0.0002
2	5.0453	0	2.1998	0.0005
3	6.8440	0.8751	2.3175	0.8817
4	79.0883	55.6920	3.9485	1.8983
5	5.6040	0.0179	0.4120	0.0777
6	0.3017	0	0.0079	0.0009
7	0.5669	0	0.0919	0.0000
8	0.0006	0	0.0018	0.0001
9	0.0006	0	0.0018	0.0002
10	10.1320	1.3739	0.4069	0.0001
11	0.0136	0.0004	0.0095	0.0016

vector adjustment is utilized in Tchebycheff approach in the final stage of evolution.

(3) Different frameworks of MOEAs, decomposition-based method and nondomination-based method are used to solve the transformed single level multiobjective model respectively for comparing which framework is more effective. We tested two decomposition-based methods, i.e. weighted sum approach and Tchebycheff approach, also tested a nondomination-based method, i.e. NSGA-II.

The remainder of the paper is organized as follows. MOBLPP with multiple objectives at the upper level is described in the next section. Reformulation of the original problem as a multiobjective single-level programming problem is provided in Section 3. A constrained multiobjective differential evolution algorithm based on decomposition is presented in Section 4. Experimental results and comparison are provided in Section 5. Section 6 concludes this paper.

2. MOBLPP with multiple objectives at the upper level

We consider the following minimization problem for bilevel programming in which multiple objectives only arise in the upper level programming.

$$\begin{aligned} \min_{x,y} \quad & F(x,y) = \Big(F_1(x,y), F_2(x,y), \cdots, F_k(x,y)\Big) \\ \text{s. t.} \quad & G(x,y) \le 0 \\ \min_{y} \quad & f(x,y) \\ \text{s. t.} \quad & g(x,y) \le 0 \end{aligned} \tag{1}$$

where $x \in R^n$ is the upper level variable or the leader, $y \in R^m$ is the lower level variable or the follower, F: $R^{n\times m} \longrightarrow R^k$, f: $R^{n\times m} \longrightarrow R$, G: $R^{n\times m} \longrightarrow R^p$, g: $R^{n\times m} \longrightarrow R^q$.

The related sets of problem (1) is defined as follows.

Ω is the constraint region of problem (1), which is defined as:

$\Omega = \{(x,y) : G(x,y) \le 0,\ \ g(x,y) \le 0\}$.

The projection of Ω onto the leader's decision space is denoted by

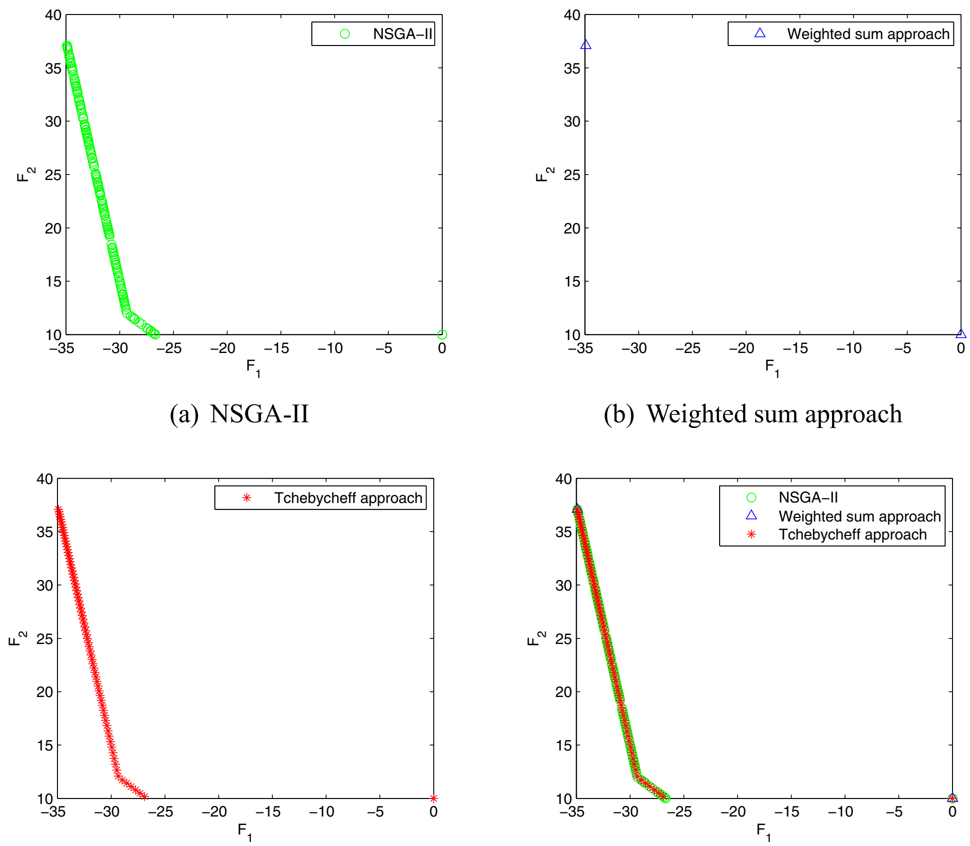
$I = \{x : \exists y,\ \text{such that } (x,y) \in \Omega\}$.

The follower's rational reaction set for $x \in I$ is defined as

$R(x) = \{y : y \in \operatorname{argmin}\{f(x,\bar{y}) : g(x,\bar{y}) \le 0\}$.

The inducible region is denoted by

$IR = \{(x,y) : (x,y) \in \Omega,\ y \in R(x)\}$.



(a) NSGA-II (b) Weighted sum approach (c) Tchebycheff approach (d) Comparison of three methods

Fig. 2. The Pareto front produced by three approaches for Problem 2.

In terms of the above definitions, the MOBLPP can be equivalently written as:

$$\begin{aligned} \min_{x,y} \quad & F(x,y) = \Big(F_1(x,y), F_2(x,y), \cdots, F_k(x,y)\Big) \\ \text{s. t.} \quad & (x,y) \in IR \end{aligned} \tag{2}$$

We assume that $F(x, y)$ and $G(x, y)$ may be discontinuous, nonconvex, even nondifferentiable, $f(x, y)$ and $g(x, y)$ are differentiable and convex in y for x fixed.

We aim at finding a set of Pareto optimal solutions of the MOBLPP. Therefore, it is necessary to introduce some related well-defined concepts of multiobjective optimization in [38]. In problem (2), IR is the decision variable space, and $\{F(x, y): (x, y) \in IR\}$ is called the objective space.

Let (x^1, y^1), $(x^2, y^2) \in IR$ be two decision variables, (x^1, y^1) is said to dominate (x^2, y^2) if $F_i(x^1, y^1) \le F_i(x^2, y^2)$ for all $i = 1, 2, \cdots, k$, and $F(x^1, y^1) \ne F(x^2, y^2)$. A point $(x^*, y^*) \in IR$ is called (globally) Pareto optimal if there is no $(x, y) \in IR$ such that (x, y) dominates (x^*, y^*). The set of all the Pareto optimal solutions, denoted by PS, is called the Pareto set. The set of all the Pareto objective vectors, $PF = \{F(x,y) : (x,y) \in PS\}$, is called the Pareto front [38–40].

3. Reformulation of the original problem as a multiobjective single-level programming problem

Problem (2) is an implicit formulation of the MOBLPP (1), so it cannot be solved directly by MOEAs. For simplifying computation, the original MOBLPP is converted into the following multiobjective single-level programming problem with the complementarity constraints by using the KKT conditions to lower level optimization problem.

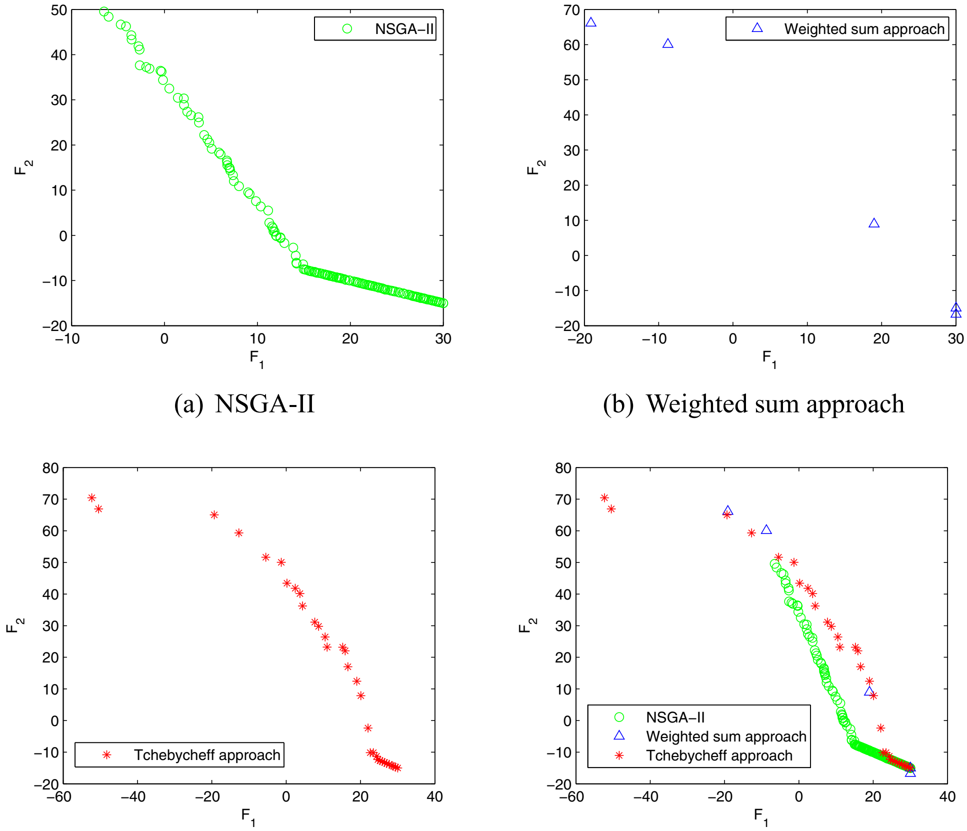
$$\begin{aligned} \min_{x,y,\gamma} \quad & F(x,y) = \Big(F_1(x,y), F_2(x,y), \cdots, F_k(x,y)\Big) \\ \text{s. t.} \quad & G(x,y) \le 0 \\ & \nabla_y L(x,y,\gamma) = 0 \\ & \gamma^\top g(x,y) = 0 \\ & g(x,y) \le 0 \\ & \gamma \ge 0 \end{aligned} \tag{3}$$

where Lagrange function $L(x,y,\gamma) = f(x,y) + \gamma^\top g(x,y)$, and $\gamma \in R^q$ are Lagrange multipliers. Thus, $\nabla_y L(x,y,\gamma) = \nabla_y f(x,y) + \gamma^\top \nabla_y g(x,y)$.

In order to deal with the complementarity constraints of problem (3), we use the smoothing function $\phi_\mu(a,b) = a + b - \sqrt{(a-b)^2 + 4\mu^2}$ of the Chen-Mangasarian function in [37], which has the following important property: $\phi_\mu(a,b) = 0$ if and only if $a \ge 0, b \ge 0, ab = \mu^2$ for every μ. For $\mu = 0$, $\phi_\mu(a,b) = 2\min(a,b)$, while, for every $\mu \ne 0$, $\phi_\mu(a, b)$ is a $\mathcal{C}^\infty$ function. Furthermore, for every (a, b), $\lim_{\mu \to 0} \phi_\mu(a,b) = 2\min(a,b)$ [37].

In our previous work [55], this smoothing function has been used successfully to cope with the complementarity constraints. Now it is adopted again for handling the complementarity constraints in problem (3). By applying the above-mentioned smoothing function $\phi_\mu(a,b) = a + b - \sqrt{(a-b)^2 + 4\mu^2}$ for complementarity constraints, problem (3) is approximated by:

$$\begin{aligned} \min_{x,y,\gamma} \; & F(x,y) = \Big(F_1(x,y), F_2(x,y), \cdots, F_k(x,y)\Big) \\ \text{s. t.} \quad & G(x,y) \le 0, \\ & \nabla_y L(x,y,\gamma) = 0, \end{aligned}$$



(a) NSGA-II (b) Weighted sum approach

(c) Tchebycheff approach (d) Comparison of three methods

Fig. 3. The Pareto front produced by three approaches for Problem 3.

$$\phi_\mu(\gamma_i, -g_i(x,y)) = 0, \;\; i = 1, 2, \cdots, q \tag{4}$$

where $\gamma = (\gamma_1, \gamma_2, \cdots, \gamma_q)^\top, g(x,y) = (g_1(x,y), g_2(x,y), \cdots, g_q(x,y))^\top$, and

$\phi_\mu(\gamma_i, -g_i(x,y)) = \gamma_i - g_i(x,y) - \sqrt{(\gamma_i + g_i(x,y))^2 + 4\mu^2}$. The parameter $\mu \in R$ is a small positive number.

Let $\omega = (y, \gamma)$, and introduce the function $\mathcal{H} : R^{n+m+q} \to R^{m+q}$, defined as

$$\mathcal{H}(x,\omega) := \mathcal{H}(x,y,\gamma) := \begin{bmatrix} \nabla_y L(x,y,\gamma) \\ \phi_\mu(\gamma_1, -g_1(x,y)) \\ \phi_\mu(\gamma_2, -g_2(x,y)) \\ \vdots \\ \phi_\mu(\gamma_q, -g_q(x,y)) \end{bmatrix}.$$

Let $\mathcal{F}_1(x,\omega) = F_1(x,y), \mathcal{F}_2(x,\omega) = F_2(x,y), \cdots, \mathcal{F}_k(x,\omega) = F_k(x,y)$, and $\mathcal{G}(x,\omega) = G(x,y)$.

Problem (4) is reformulated more compactly as

$$\begin{aligned} \min_{x,\omega} \quad & \mathcal{F}(x,\omega) = \Big(\mathcal{F}_1(x,\omega), \mathcal{F}_2(x,\omega), \cdots, \mathcal{F}_k(x,\omega)\Big) \\ \text{s. t.} \quad & \mathcal{G}(x,\omega) \le 0 \\ & \mathcal{H}(x,\omega) = 0 \end{aligned} \tag{5}$$

It is more difficult to solve directly problem (5) in that both equality and inequality constraints arise in problem (5) with many variables including the upper level and lower level variables, as well as a newly introduced vector, i.e. Lagrange multipliers. For simplifying effectively problem (5), the following interactive approach is performed with iteration.

At first, for a given upper-level variable x, the following system of equations is solved, and the related solution $\omega = (y, \gamma)$ is obtained:

$$\mathcal{H}(x,\omega) = 0. \tag{6}$$

Then, the following multiobjective optimization problem only with inequality constraints only on the upper-level variable x is solved by using a constrained MOEA:

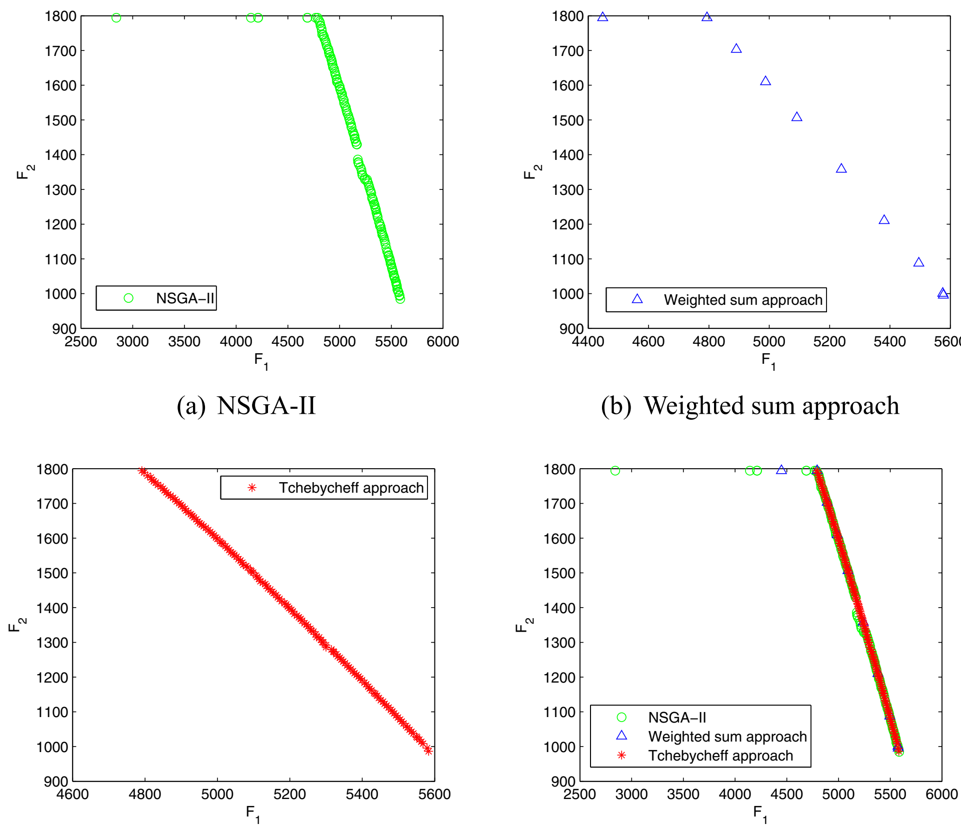
$$\begin{aligned} \min_{x} \quad & \mathcal{F}(x,\omega) = \Big(\mathcal{F}_1(x,\omega), \mathcal{F}_2(x,\omega), \cdots, \mathcal{F}_k(x,\omega)\Big) \\ \text{s. t.} \quad & \mathcal{G}(x,\omega) \le 0 \end{aligned} \tag{7}$$

4. Constrained multiobjective differential evolution algorithm based on decomposition

In recent decades much significant progress has been made in the development of MOEAs [39–53]. MOEAs can find a set of representative Pareto optimal solutions in a single run. MOEA/D [39] is an effective MOEA based on decomposition, which simultaneously optimizes a number of single objective optimization subproblems. A constrained MOEA/D combined with differential evolution (DE) [56,57] is used to solve multiobjective optimization problem (7), denoted by MOEA/D-DE.

In the constrained MOEA/D-DE, two distinct sets of weight vectors are applied in the different evolutionary periods. In the first phase, the set of customary weight vectors in [39] is adopted to reach the approximate Pareto front. In the last phase, a set of modified weight vectors based on the obtained approximate Pareto front is used to improve the uniformity of the approximate Pareto front. Additionally, the penalty function method is utilized to handle inequality constraints in problem (7).

Constrained MOEA/D-DE utilizes weighted sum approach and Tchebycheff approach respectively for converting problem (7) into



(a) NSGA-II (b) Weighted sum approach

(c) Tchebycheff approach (d) Comparison of three methods

Fig. 4. The Pareto front produced by three approaches for Problem 4.

a number of single objective optimization subproblems in order to investigate which approach is suitable for solving problem (7).

A single objective optimization subproblem in weighted sum approach is

$$\begin{aligned} \min_{x} \quad & \mathcal{S}^{ws}((x,\omega)|\lambda) = \sum_{i=1}^{k} \lambda_i \mathcal{F}_i(x,\omega) \\ \text{s. t.} \quad & \mathcal{G}(x,\omega) \le 0 \end{aligned} \tag{8}$$

where $\lambda = (\lambda_1, \lambda_2, \cdots, \lambda_k)$ is a weight vector, i.e. $0 \le \lambda_i \le 1$ for all $i = 1, 2 \cdots, k$ and $\sum_{i=1}^{k} \lambda_i = 1$.

Another single objective optimization subproblem in Tchebycheff approach is

$$\begin{aligned} \min_{x} \quad & \mathcal{S}^{te}((x,\omega)|\lambda, z^*) = \max_{1\le i\le k}\{\lambda_i |\mathcal{F}_i(x,\omega) - z_i^*|\} \\ \text{s. t.} \quad & \mathcal{G}(x,\omega) \le 0 \end{aligned} \tag{9}$$

where $\lambda = (\lambda_1, \lambda_2, \cdots, \lambda_k)$ is a weight vector. $z^* = (z_1^*, z_2^*, \cdots, z_k^*)$ is the reference point, i.e. $z_i^* = \min\{\mathcal{F}_i(x,\omega) : \mathcal{G}(x,\omega) \le 0\}$ for each $i = 1, 2 \cdots, k$.

4.1. Individual coding and initial population

For solving problem (7), a vector $x = (x_1, x_2, \cdots, x_n)$ is used to represent an individual, which is an upper level decision variable.

Let lb_i be the lower bounds and ub_i be the upper bounds of each x_i of upper level variable x, i.e. $lb_i \le x_i \le ub_i$, $i = 1, 2, \cdots, n$.

$$lb_i = \min\{x_i : x \in I\}, \quad ub_i = \max\{x_i : x \in I\}.$$

Generate randomly NP individuals to constitute an initial population x^1, x^2, $\cdots$, x^{NP}, where $x^t = (x_1^t, x_2^t, \cdots, x_n^t)$, $t = 1, 2, \cdots, NP$, such that $x_i^t \in [lb_i, ub_i]$. Herein, NP is the number of optimization subproblems.

4.2. Fitness function

For every individual x^t $(t = 1, 2, \cdots, NP)$ in the population, the solution ω^t is obtained by solving the system of Eq. (6). The fitness function $fit(x^t|\lambda)$ is expressed as:

$$fit(x^t|\lambda) = \begin{cases} \mathcal{S}(x^t,\omega^t) + \mathcal{P}(x^t,\omega^t), & \text{if } \omega^t \text{ exists} \\ +\infty, & \text{otherwise} \end{cases} \tag{10}$$

where $\mathcal{S}(x^t,\omega^t) = \mathcal{S}^{ws}((x^t,\omega^t)|\lambda)$ for weighted sum approach, or $\mathcal{S}(x^t,\omega^t) = \mathcal{S}^{te}((x^t,\omega^t)|\lambda, z^*)$ for Tchebycheff approach, $\mathcal{P}(x^t,\omega^t) = \Theta \cdot \sum_{i=1}^{p} \max\{0, \mathcal{G}_i(x^t,\omega^t)\}$, and $\Theta > 0$ denotes a penalty parameter.

4.3. Mutation operation

Two mutation operators are adopted to generate new population for maintaining the diversity of population. The population of Tth generation is first generated by the following DE/rand/1 mutation operator:

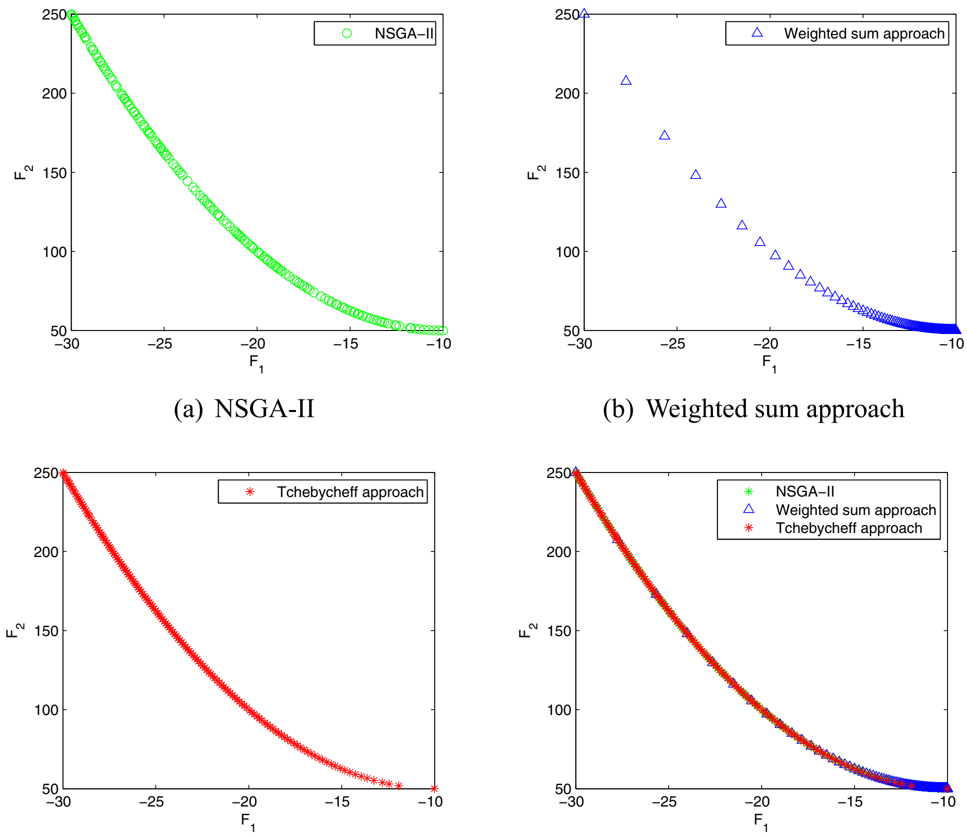
$$\bar{v}^t(T) = x^{r_1}(T) + SF \cdot (x^{r_2}(T) - x^{r_3}(T)) \tag{11}$$

where $t = 1, 2, \cdots, NP$, r_1, r_2, r_3 are mutually distinct indexes not equal to t. $SF \in (0, 1)$ denotes a scaling factor.

The new population is then generated by the following Gaussian mutation operator with probability p_m:

$$v^t(T) = \bar{v}^t(T) + \Delta \tag{12}$$

where $t = 1, 2, \cdots, NP$, $\Delta \sim \mathcal{N}(0, \delta^2)$, i.e., Δ obeys an n-dimensional normal distribution with mean value $0 = (0, 0, \cdots, 0)$ and deviation $\delta = (\delta_1, \delta_2, \cdots, \delta_n)$. If $rand(0,\ 1) < 0.5$ ($rand(0,\ 1)$ denotes a random number in the range $(0,\ 1)$), $\delta_i = \frac{ub_i - lb_i}{20}$; otherwise, $\delta_i = 1$ $(i = 1, 2, \cdots, n)$.



(a) NSGA-II (b) Weighted sum approach

(c) Tchebycheff approach (d) Comparison of three methods

Fig. 5. The Pareto front produced by three approaches for Problem 5.

4.4. Crossover operation

For each pair of individuals constituted by the parent individual $x^t(T) = (x_1^t(T), x_2^t(T), \cdots, x_n^t(T))$ and mutation individual $v^t(T) = (v_1^t(T), v_2^t(T), \cdots, v_n^t(T))$, the crossover individual $u^t(T) = (u_1^t(T), u_2^t(T), \cdots, u_n^t(T))$ is generated by the following binomial crossover of DE.

$$u_i^t(T) = \begin{cases} v_i^t(T), & \text{if } R_i \le CR \text{ or } i = rn(t) \\ x_i^t(T), & \text{otherwise} \end{cases} \tag{13}$$

where $R_i \in (0, 1)$ is a random number, $rn(t) \in \{1, 2, \cdots, n\}$ is the randomly selected index chosen once for each t, and $CR \in (0, 1)$ is a real-valued crossover rate constant.

4.5. Selection operation

By comparing the trial vector $u^t(T)$ and the parent vector $x^t(T)$, $t = 1, 2, \cdots, NP$, the individual $x^t(T+1)$ of the $(T+1)$th generation is decided as follows:

$$x^t(T+1) = \begin{cases} u^t(T), & \text{if } fit(u^t(T)|\lambda) < fit(x^t(T)|\lambda) \\ x^t(T), & \text{otherwise} \end{cases} \tag{14}$$

4.6. Weight vector adjustment for Tchebycheff approach

The common weight vectors used in MOEA/D are preseted uniformly according to the following form.

All the weight vectors $\lambda^1, \lambda^2, \cdots, \lambda^{NP}$ in weighted sum approach and Tchebycheff approach are controlled by a parameter h, where $h = NP - 1$, and each individual weight takes a value from $\{\frac{0}{h}, \frac{1}{h}, \cdots, \frac{h}{h}\}$ [39]. For example, the weight vectors for biobjective optimization are listed as: $\lambda^1 = (\frac{0}{h}, \frac{h}{h}), \lambda^2 = (\frac{1}{h}, \frac{h-1}{h}), \cdots, \lambda^{NP} = (\frac{h}{h}, \frac{0}{h})$.

The distribution of solutions in objective space cannot be determined only by the Euclidean distance of weights in the weight vector space. To get uniform distribution of solutions in objective space, it is necessary to adjust the weight vector in the last stage of evolution. For the sake of simplicity, we analyze the weight vector adjustment on biobjective optimization problems using Tchebycheff approach.

Let the reference point be (z_1^*, z_2^*), and the weight vector be $\lambda^i = (\lambda_1^i, \lambda_2^i)$ for the ith optimization subproblem. The decomposition-based single objective optimization subproblem is expressed as follows.

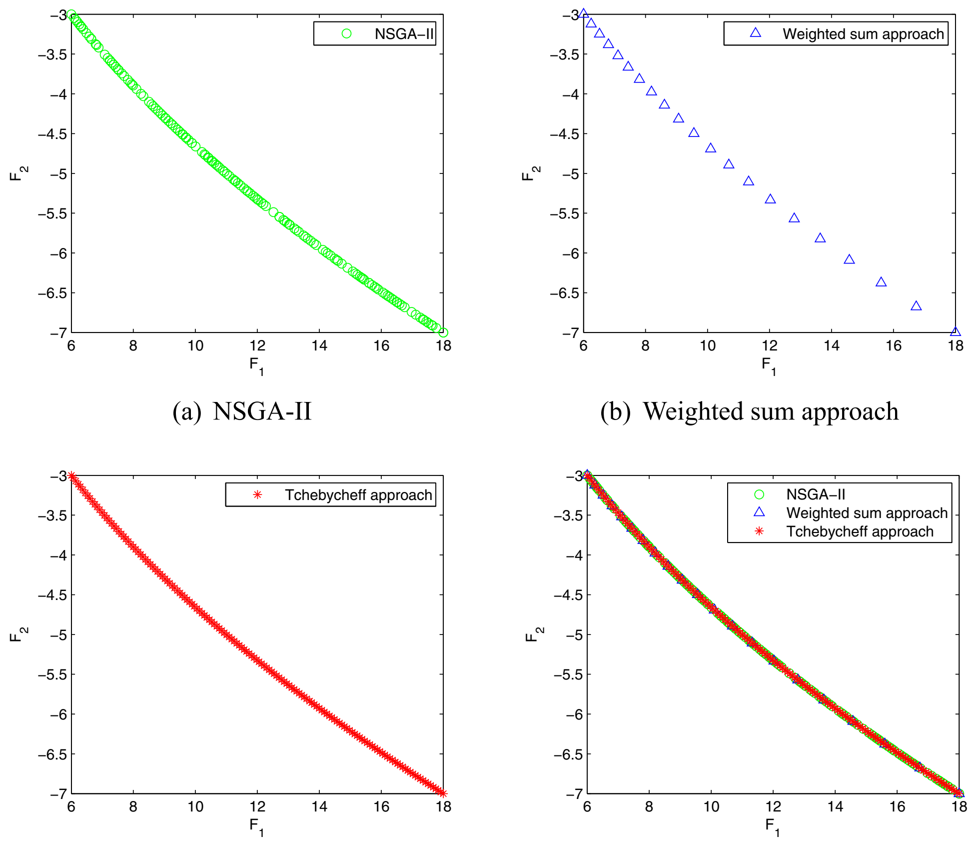
$$\min_{x \in X} \left(\max_{1 \le j \le 2} \{\lambda_j^i |\mathcal{F}_j(x, \omega) - z_j^*|\} \right) \tag{15}$$

where $X = \{x | \mathcal{G}(x, \omega) \le 0\}$, and ω is a solution of the system of equations $\mathcal{H}(x, \omega) = 0$ for a given $x \in X$.

Theorem 1. *If there exists $x^* \in PS$ such that $\lambda_1^i |\mathcal{F}_1(x^*, \omega^*) - z_1^*| = \lambda_2^i |\mathcal{F}_2(x^*, \omega^*) - z_2^*|$, then x^* is a solution of the problem (15).*

Proof. If $\lambda_1^i = 0$, then $\lambda_2^i = 1$. Thus, $\mathcal{F}_2(x^*, \omega^*) = z_2^*$. It's obvious that x^* is a solution of the problem (15). When $\lambda_1^i = 1$ and $\lambda_2^i = 0$, there is a similar result.

For $\lambda_j^i \neq 0$, $j = 1, 2$, we assume that x^* is not a solution of problem (15). If there exists $\bar{x}$ which is a solution of problem (15), then we get $\lambda_1^i |\mathcal{F}_1(\bar{x}, \bar{\omega}) - z_1^*| < \lambda_1^i |\mathcal{F}_1(x^*, \omega^*) - z_1^*|$, $\lambda_2^i |\mathcal{F}_2(\bar{x}, \bar{\omega}) - z_2^*| < \lambda_2^i |\mathcal{F}_2(x^*, \omega^*) - z_2^*|$. Because $\lambda_j^i \neq 0$, and $\mathcal{F}_1(\bar{x}, \bar{\omega}) \ge z_1^*$, $\mathcal{F}_1(x^*, \omega^*) > z_1^*$, $\mathcal{F}_2(\bar{x}, \bar{\omega}) \ge z_2^*$, $\mathcal{F}_2(x^*, \omega^*) > z_2^*$, we



(a) NSGA-II (b) Weighted sum approach (c) Tchebycheff approach (d) Comparison of three methods

Fig. 6. The Pareto front produced by three approaches for Problem 6.

have $\mathcal{F}_1(\bar{x}, \bar{\omega}) < \mathcal{F}_1(x^*, \omega^*)$, $\mathcal{F}_2(\bar{x}, \bar{\omega}) < \mathcal{F}_2(x^*, \omega^*)$. So this is totally contradictory to the condition $x^* \in PS$. Thus x^* is a solution of problem (15). □

Theorem 1 could be easily extended to tri-objective or many-objective optimization problems.

For Tchebycheff approach, each weight vector determines a search direction with the equation as follows:

$$\mathcal{F}_2 = \frac{\lambda_1^i}{\lambda_2^i}(\mathcal{F}_1 - z_1^*) + z_2^*. \tag{16}$$

In terms of Theorem 1, each subproblem tries to obtain the solution defined by the intersection of the direction and true Pareto front. Based on the approximation to Pareto front obtained after some evolutionary generations, we can adjust the weight vectors to get the better distribution of solutions.

According to above analysis, we present Algorithm 1 to adjust the weight vectors as follows.

Algorithm 1. Weight vectors adjustment based on the approximation to Pareto front (*PF*) and a set of used weight vectors

Step 1. Calculate the cardinality of *PF*: $NP = |PF|$;
Estimate reference point according to approximate *PF*: $z^* = (z_1^*, z_2^*)$;
Compute neighbouring Euclid distance in objective space: $d_i = \|\bar{\mathcal{F}}_{i+1} - \bar{\mathcal{F}}_i\|_2$, where $\bar{\mathcal{F}}_i = (\mathcal{F}_{i,1}, \mathcal{F}_{i,2}), i = 1, 2, \cdots, NP - 1$;
Compute average targeting distance: $\bar{d} = \frac{\sum_{i=1}^{NP-1} d_i}{NP-1}$;
Let $(\lambda^*)^i = \lambda^i$, where $\lambda^i = (\frac{i-1}{h}, \frac{h-i+1}{h})$, $h = NP - 1$, $i = 1, 2, \cdots, NP$.
Step 2. For $i = 2, 3, \cdots, NP - 1$ do
Based on curve fitting, estimate the local *PF* segment $\bar{g}$: $\bar{f}_1 \to \bar{f}_2$ around $\bar{\mathcal{F}}_i = (\mathcal{F}_{i,1}, \mathcal{F}_{i,2}) \in PF$;
Denote $\bar{\mathcal{F}}_{i-1}$ as $(\mathcal{F}_{i-1,1}, \mathcal{F}_{i-1,2})$ for $\bar{\mathcal{F}}_{i-1} \in PF$;
Solve $|\mathcal{F}_{i-1,2} - \bar{g}(\bar{f}_1)|^2 + |\mathcal{F}_{i-1,1} - \bar{f}_1|^2 = \bar{d}^2$, where $(\mathcal{F}_{i-1,1} - \bar{f}_1)(\mathcal{F}_{i-1,1} - \mathcal{F}_{i,1}) > 0$;
Let $\bar{f}_2 = \bar{g}(\bar{f}_1)$, $\bar{\mathcal{F}}_i = (\bar{f}_1, \bar{f}_2)$;
According to Theorem 1, obtain a revised weight $(\lambda^*)^i = \left(\frac{|\bar{f}_2 - z_2^*|}{\sum_{i=1}^{2}|\bar{f}_i - z_i^*|}, \frac{|\bar{f}_1 - z_1^*|}{\sum_{i=1}^{2}|\bar{f}_i - z_i^*|}\right)$.
Step 3. Output $(\lambda^*)^i$, $i = 1, 2, \cdots, NP$.

In conclusion two sets of weight vectors are adopted in MOEA/D with Tchebycheff approach. In the first stage, a set of above-mentioned preseted weight vectors is used to make the solutions approach to approximate Pareto front. In the last stage, a set of the modified weight vectors according to Algorithm 1 is adopted to improve the uniform distribution of solutions in objective space.

4.7. Constrained multiobjective differential evolution algorithm based on decomposition

Now the algorithmic procedure of constrained MOEA/D-DE algorithm with Tchebycheff approach is presented as follows.

Algorithm 2. The constrained MOEA/D-DE algorithm with Tchebycheff approach

Input:

- Problem (7);
- *NP*: the number of the subproblems;

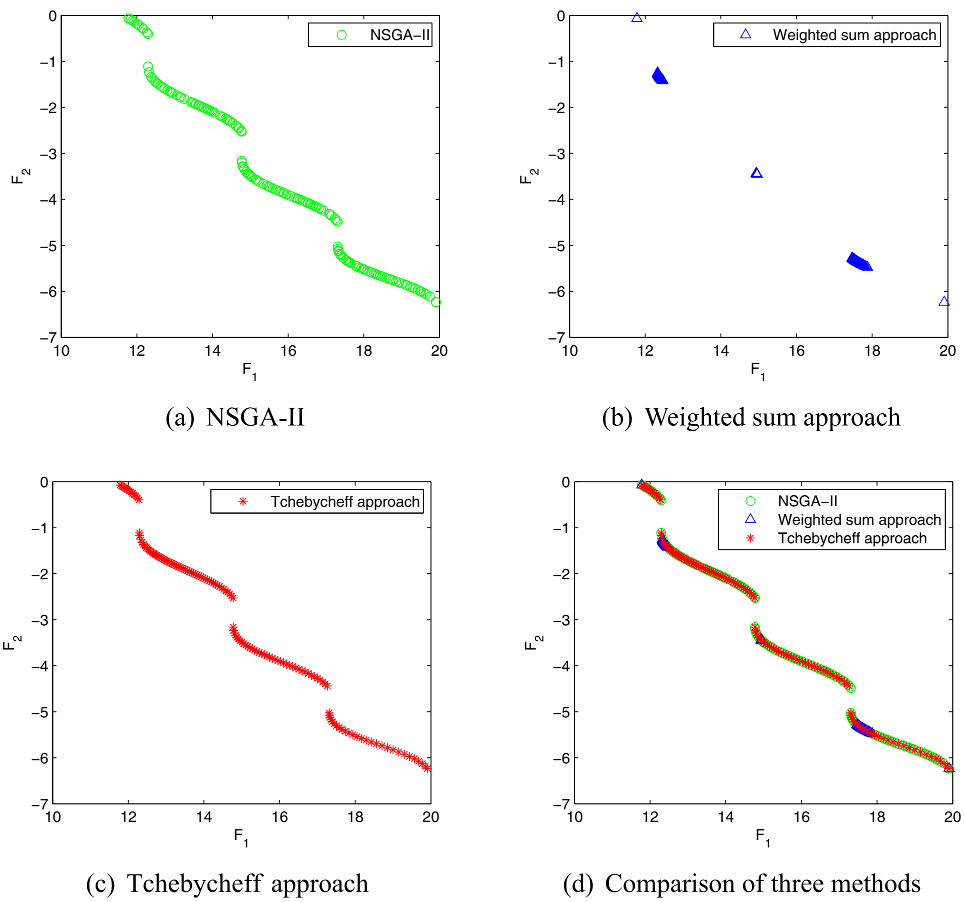


Fig. 7. The Pareto front produced by three approaches for Problem 7.

- $\lambda^1, \cdots, \lambda^{NP}$: a set of NP weight vectors;
- N: the number of the weight vectors in the neighborhood of each weight vector;
- μ: a small positive parameter in the smoothing function of problem (4);
- SF: a scaling factor in DE operator;
- CR: a real-valued crossover rate constant in DE operator;
- p_m: a Gaussian mutation probability;
- Θ: a penalty parameter in the fitness function;
- $MaxG$: maximum number of generations.

Output:

- Approximation to the Pareto set: $\{(x^1, \omega^1), \cdots, (x^{NP}, \omega^{NP})\}$;
- Approximation to the Pareto front: $\{\mathcal{F}(x^1, \omega^1), \cdots, \mathcal{F}(x^{NP}, \omega^{NP})\}$.

Step 1. **Initialization**

Set $T = 0$.

Step 1.1 Compute the Euclidean distances between any two weight vectors and then work out the N closest weight vectors to each weight vector. For each $i = 1, \cdots, NP$, set $B(i) = \{i_1, \cdots, i_N\}$, where $\lambda^{i_1}, \cdots, \lambda^{i_N}$ are the N closest weight vectors to λ^i.

Step 1.2 Generate randomly an initial population $\{x^t(T) = (x_1^t(T), \cdots, x_n^t(T)) : x_i^t(T) \in [lb_i, ub_i], i = 1, 2, \cdots, n, t = 1, 2, \cdots, NP\}$.

For every individual $x^t(T)$ $(t = 1, 2, \cdots, NP)$, solve the system of Eq. (6), and obtain the optimal solution $\omega^t(T)$.

For each weight vector λ^i $(i = 1, 2, \cdots, NP)$, evaluate the fitness values $fit(x^{i_1}(T)|\lambda^i)$, $fit(x^{i_2}(T)|\lambda^i)$, $\cdots$, $fit(x^{i_N}(T)|\lambda^i)$, and select a best individual $x^i(T)$ with minimum fitness value from N individuals with index set $B(i)$ for ith optimization subproblem.

Step 1.3 Initialize $z(T) = (z_1(T), \cdots, z_k(T))$ by setting $z_j(T) = \min\limits_{1 \le t \le NP} \{\mathcal{F}_j(x^t(T), \omega^t(T)) + \mathcal{P}(x^t(T), \omega^t(T))\}$.

Step 2. Weight vector adjustment

If $T \le 0.9 \times MaxG$, then the set of initial weight vectors keeps unchanged. Otherwise, adjust the set of initial weight vectors based on the obtained approximation to the Pareto front according to Algorithm 1.

Step 3. Mutation operation

For $t = 1, 2, \cdots, NP$, select randomly $r_1, r_2, r_3 \in B(t)$ such that $r_1 \ne r_2 \ne r_3 \ne t$, and generate a mutant individual $\bar{v}^t(T)$ according to Eq. (11); Then, generate a new mutant individual $v^t(T)$ with probability p_m according to Eq. (12).

Step 4. Crossover operation

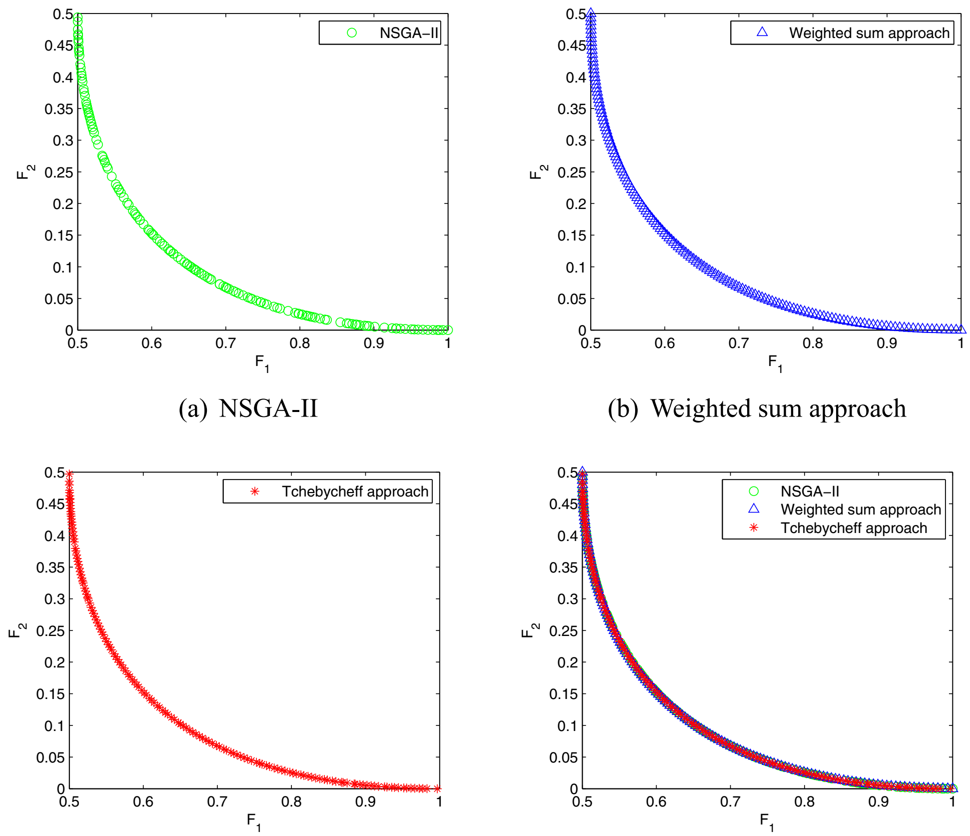
For $t = 1, 2, \cdots, NP$, produce a crossover individual $u^t(T)$ by using Eq. (13).

For each crossover individual $u^t(T)$ $(t = 1, 2, \cdots, NP)$, solve the system of Eq. (6), and obtain the optimal solution $\bar{\omega}^t(T)$.

For each weight vector λ^i $(i = 1, 2, \cdots, NP)$, evaluate the fitness values $fit(u^{i_1}(T)|\lambda^i)$, $fit(u^{i_2}(T)|\lambda^i)$, $\cdots$, $fit(u^{i_N}(T)|\lambda^i)$, and select a best individual $u^i(T)$ with minimum fitness value from N individuals with index set $B(i)$ for ith optimization subproblem.

Calculate $\bar{z}(T) = (\bar{z}_1(T), \cdots, \bar{z}_k(T))$, where $\bar{z}_j(T) = \min\limits_{1 \le t \le NP} \{\mathcal{F}_j(u^t(T), \bar{\omega}^t(T)) + \mathcal{P}(u^t(T), \bar{\omega}^t(T))\}$.

Step 5. Selection operation



(a) NSGA-II (b) Weighted sum approach

(c) Tchebycheff approach (d) Comparison of three methods

Fig. 8. The Pareto front produced by three approaches for Problem 8.

According to selection rule, form the next population $x^t(T+1)$ $(t=1,2,\cdots,NP)$.
Update $z(T+1)=(z_1(T+1),\cdots,z_k(T+1))$ according to the following formula:
For $j=1,\cdots,k$

$$z_j(T+1)=\begin{cases}\bar{z}_j(T), & \text{if } \bar{z}_j(T)<z_j(T)\\ z_j(T), & \text{otherwise}\end{cases}.$$

Step 6. Stopping criterion
If $T\geq MaxG$, then stop, and output the approximation to the Pareto set and approximation to the Pareto front. Otherwise, let $T=T+1$ and go to Step 2.

Additionally, the algorithmic procedure of constrained MOEA/D-DE algorithm with weighted sum approach is same as above algorithm 2 without Step 2.

5. Experimental studies

5.1. Test examples and experimental settings

To evaluate and compare the performances of MOEA/D-DE with weighted sum approach, MOEA/D-DE with Tchebycheff approach, and NSGA-II for problem (7), we have tested some MOBLPPs, where instances 4, 5, 7–10 were reconstructed from the examples in the literature, and instance 11 was constructed by us ($K=10$ was set in the experiments). These test instances are shown in Appendix.

In order to solve constrained multiobjective optimization (7) using NSGA-II, same penalty function method and same penalty parameter in MOEA/D-DE were utilized in NSGA-II. During the experiments, we adopted same population size and maximum number of generations in three algorithms. In NSGA-II, we adopted other control parameters provided in the literature [41].

The following parameter settings were used for MOEA/D-DE with weighted sum approach and Tchebycheff approach:

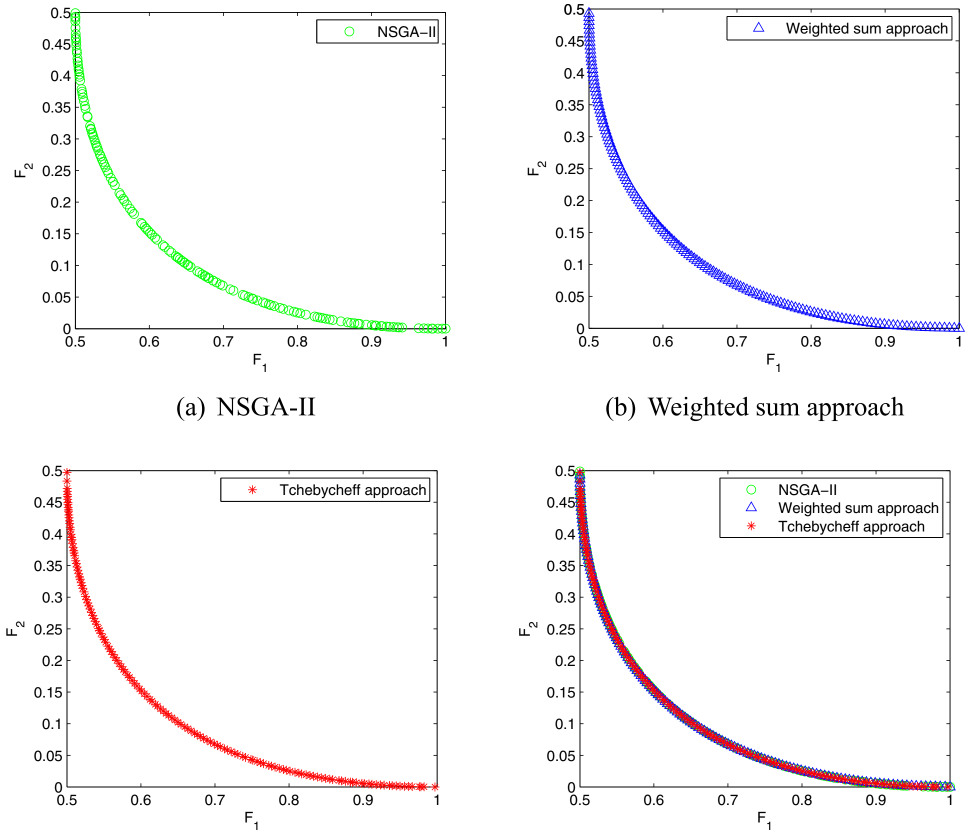
- The number of the subproblems: $NP=150$;
- The number of the weight vectors in the neighborhood of each weight vector: $N=20$;
- Scaling factor: $SF=0.5$;
- Crossover rate: $CR=0.6$;
- Gaussian mutation probability: $p_m=0.5$;
- A small positive number: $\mu=0.0001$;
- Penalty factor: $\Theta=10000$;
- Maximum number of generations: $MaxG=300$.

All experiments were performed on a Lenovo-PC with Intel Core i5-4200U CPU 1.60GHz processor and 8.00 GB of RAM using MATLAB software. Each algorithm was independently run ten times for every test instance. For solving the system of Eq. (6), we used the corresponding function calls in the optimization toolbox of MATLAB.

5.2. Performance metrics

Due to the nature of multiobjective optimization problem (MOP), multiple performance indexes should be used for comparing the performances of different algorithms [48]. In our experiments, the following performance measures were used.

(1) *C*-metric (Coverage measure)[44]: Let *A* and *B* be two approximations to the Pareto front of a MOP, *C*(*A*, *B*) is defined as the



(a) NSGA-II (b) Weighted sum approach

(c) Tchebycheff approach (d) Comparison of three methods

Fig. 9. The Pareto front produced by three approaches for Problem 9.

percentage of the solutions in B that are dominated by at least one solution in A, i.e.,

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

where $|B|$ denotes the number of the elements in the set 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. Generally, $C(A,B)\neq 1-C(B,A)$. If $C(A,\ B)>C(B,\ A)$, the Pareto front of A is better than that of B.

(2) S-metric[54]: This metric is used to calculate the uniformity in the distribution of Pareto front. For the most uniformly spread-out Pareto front, the numerator of S would be zero, making the metric take a value zero. Thus a smaller S indicator value is preferable.

$$S(A)=\sqrt{\frac{1}{|A|-1}\sum_{i=1}^{|A|-1}(d_i-\bar{d})^2}$$

where $d_i=\min_{j\neq i}\{\sum_{l=1}^{k}|F_l(x^i,y^i)-F_l(x^j,y^j)|\}$, $i,j=1,2,\cdots,|A|-1$. $\bar{d}$ is the average of all distances d_i, $i=1,2,\cdots,|A|-1$, assuming that there are $|A|$ solutions on the best Pareto front.

5.3. MOEA/D-DE with weighted sum approach vs. MOEA/D-DE with Tchebycheff approach

Table 1 presents average CPU time used by MOEA/D-DE with weighted sum approach and Tchebycheff approach for each test instance in ten independent runs.

Table 2 provides the mean value and standard deviation of C-metric values of the final approximations between MOEA/D-DE with weighted sum approach and MOEA/D-DE with Tchebycheff approach for each test instance in ten independent runs.

Table 3 shows the mean value and standard deviation of S-metric values obtained by MOEA/D-DE with weighted sum approach and MOEA/D-DE with Tchebycheff approach for each instance in ten independent runs.

With the same population size and same number of generations, it is evident from Table 1 that computational time required by MOEA/D-DE with weighted sum approach is similar to that by MOEA/D-DE with Tchebycheff approach on average. From Table 2, it is clear that MOEA/D-DE with Tchebycheff approach achieves the better final Pareto optimal solutions than MOEA/D-DE with weighted sum approach except instances 2, 3, and 10. Table 3 clearly indicates that the uniformity of final Pareto front obtained by MOEA/D-DE with Tchebycheff approach is better than that obtained by MOEA/D-DE with weighted sum approach in terms of S-metric, for all the test instances except instances 8 and 9.

The difference between the approximations of Pareto front obtained by MOEA/D-DE with Tchebycheff approach and weighted sum approach on all instances can be visually detected from Figs. 1–11. It is clear that the final Pareto fronts obtained by MOEA/D-DE with Tchebycheff approach are much better than those of MOEA/D-DE with weighted sum approach in terms of diversity and distribution of solutions for all test instances except instances 8 and 9.

Overall, MOEA/D-DE with Tchebycheff approach performs better than MOEA/D-DE with weighted sum approach in terms of the convergence and distribution of obtained Pareto front. It suggests

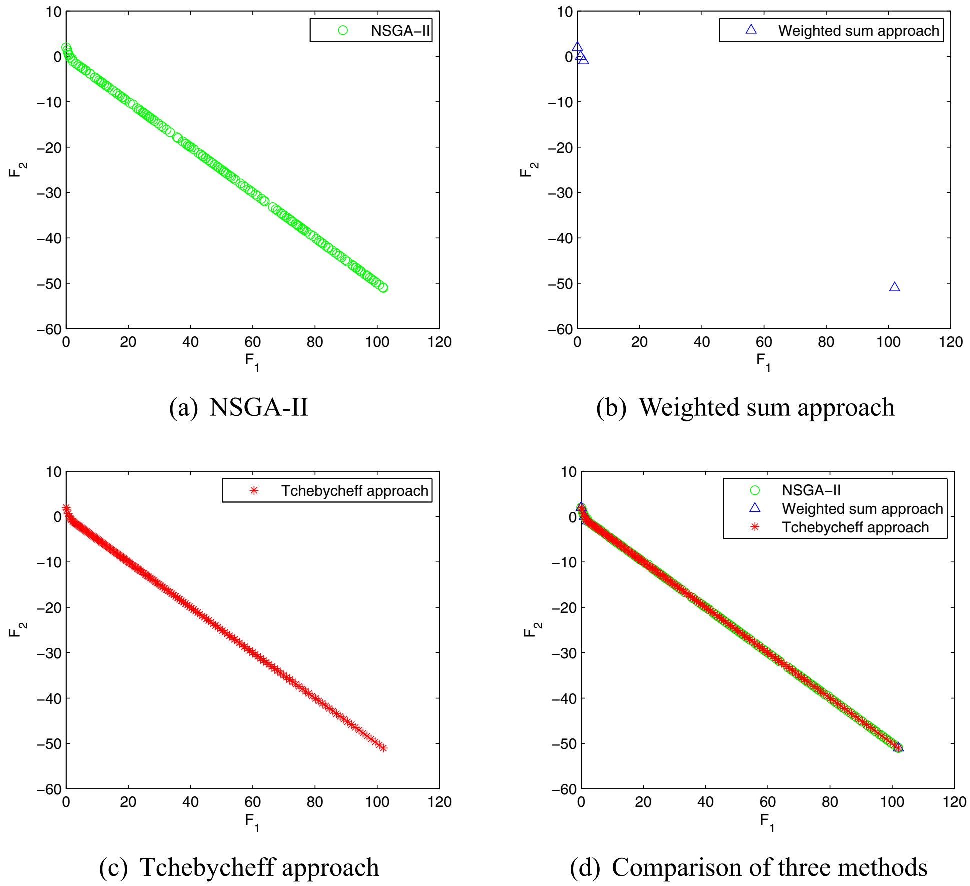


Fig. 10. The Pareto front produced by three approaches for Problem 10.

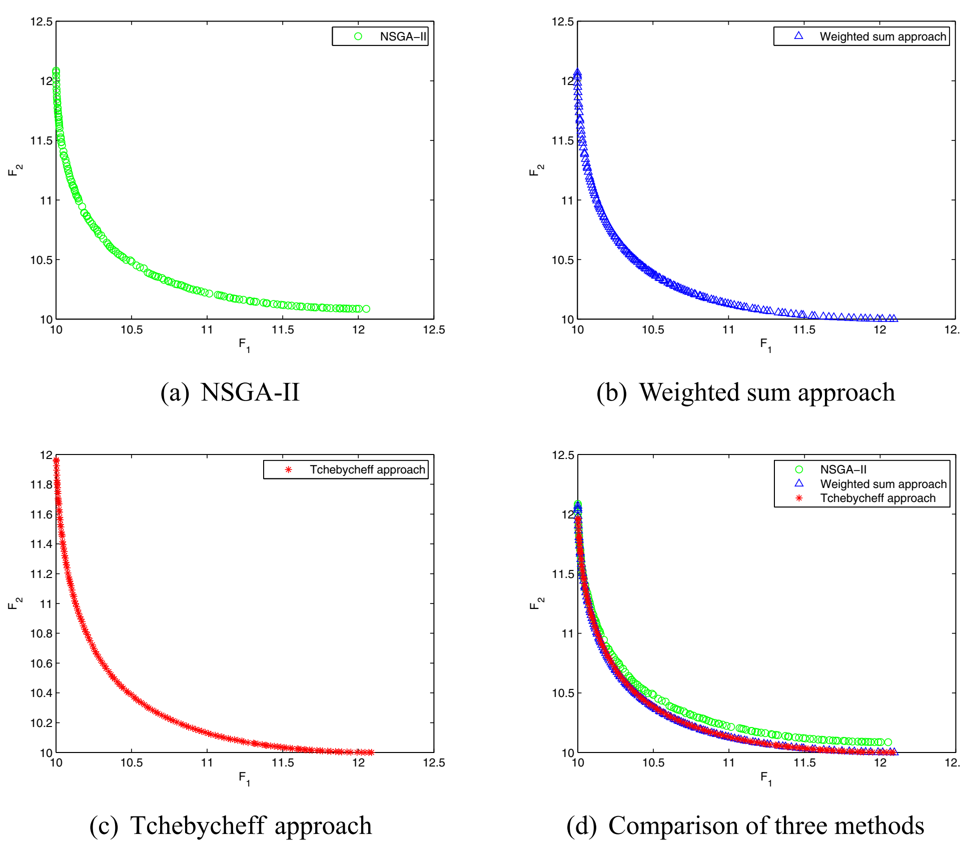


Fig. 11. The Pareto front produced by three approaches for Problem 11.

Table 4
The average CPU time (in seconds) used by NSGA-II and MOEA/D-DE with Tchebycheff approach.

Instance	NSGA-II	Tchebycheff approach
1	344.3431	269.1355
2	439.8860	305.5410
3	1041.2726	1169.1095
4	1385.9343	1652.7553
5	458.8090	381.0618
6	287.1288	256.3112
7	552.0388	579.9305
8	246.5130	219.2462
9	568.2115	550.5114
10	665.9153	691.0868
11	1064.1987	1102.1321

Table 5
The *C*-metric values between NSGA-II (*A*) and MOEA/D-DE with Tchebycheff approach (*B*). Mean denotes the mean value of *C*-metric values, and SD means the standard deviation of *C*-metric values in ten independent runs.

Instance	$C(A, B)$		$C(B, A)$	
	Mean	SD	Mean	SD
1	0.0026	0.0034	0.0080	0.0052
2	0.0304	0.0560	0.0020	0.0032
3	0.1834	0.2658	0.4379	0.3067
4	0.1019	0.0869	0.2446	0.0809
5	0	0	0.0007	0.0021
6	0.0013	0.0028	0.0146	0.0042
7	0.0265	0.0342	0.0160	0.0035
8	0.0165	0.0134	0.0327	0.0131
9	0.0106	0.0100	0.0413	0.0160
10	0.0046	0.0045	0.0307	0.0141
11	0.0371	0.0179	0.6840	0.1865

Table 6
The mean value (Mean) and standard deviation (SD) of *S*-metric values in NSGA-II and MOEA/D-DE with Tchebycheff approach.

Instance	NSGA-II		Tchebycheff approach	
	Mean	SD	Mean	SD
1	0.0210	0.0015	0.0049	0.0002
2	2.1716	0.0441	2.1998	0.0005
3	1.6865	0.7132	2.3175	0.8817
4	71.2698	66.0196	3.9485	1.8983
5	1.0374	0.0176	0.4120	0.0777
6	0.1936	0.0017	0.0079	0.0009
7	0.0906	0.0022	0.0919	0.0000
8	0.0042	0.0003	0.0018	0.0001
9	0.0038	0.0002	0.0018	0.0002
10	0.6129	0.0470	0.4069	0.0001
11	0.0154	0.0051	0.0095	0.0016

that weighted sum approach may be unsuitable for MOBLPPs with multiobjective optimization at the upper level due to the intrinsic non-convexity of MOBLPPs, which has been also demonstrated by numerical experiments.

5.4. NSGA-II vs. MOEA/D-DE with Tchebycheff approach

Table 4 shows the average runtime used by NSGA-II and MOEA/D-DE with Tchebycheff approach for each instance in ten independent runs.

Table 5 provides the mean value and standard deviation of *C*-metric values of the final approximations between NSGA-II and MOEA/D-DE with Tchebycheff approach for each instance in ten independent runs.

Table 6 presents the mean value (Mean) and standard deviation (SD) of *S*-metric values in NSGA-II and MOEA/D-DE with Tchebycheff approach for each instance in ten independent runs.

From Table 4, it is clear that the runtime required by MOEA/D-DE with Tchebycheff approach is similar to that by NSGA-II on average. It is evident from Table 5, the final Pareto optimal solutions obtained by MOEA/D-DE with Tchebycheff approach are better than those generated by NSGA-II in terms of *C*-metric for all the test instances except instances 2 and 7. The uniformity of Pareto front obtained by MOEA/D-DE with Tchebycheff approach is better than that of NSGA-II in terms of *S*-metric from Table 6, for most of test instances except instances 2, 3 and 7.

Figs. 1–11 display the Pareto fronts achieved by NSGA-II and MOEA/D-DE with Tchebycheff approach in the last generation of a typical run, respectively. It can be observed that MOEA/D-DE with Tchebycheff approach outperforms NSGA-II in terms of the convergence and distribution of solutions.

In conclusion, we can claim that MOEA/D-DE with Tchebycheff approach can produce better approximations of Pareto front than NSGA-II and MOEA/D-DE with weighted sum approach on most of test instances.

5.5. Sensitivity analysis of parameters

In this subsection, we discussed the impacts of different parameters on MOEA/D-DE with Tchebycheff approach to illustrate the rationality of the selected parameters.

(1) Impacts of *SF* and *CR* on MOEA/D-DE with Tchebycheff approach: $SF \in [0.5, 1)$ and $CR \in (0, 1)$ are two control factors in the DE operators for generating new individuals. To investigate the impacts of *SF* and *CR* on the convergence of algorithm, we have tested 30 combinations of six values of *SF* (i.e., 0.5, 0.6, 0.7, 0.8, 0.9, and 1) and five values of *CR* (i.e., 0.1, 0.3, 0.5, 0.7, and 0.9) on Problem 1. All the other parameters remain the same as in Section 5.1. Each combination of (*SF*, *CR*) has been tested ten times.
Table 7 shows the mean C-metric values between the original setting of combination $(SF, CR) = (0.5, 0.6)$ and 30 different combinations (*SF*, *CR*) in MOEA/D-DE with Tchebycheff approach. It is evident that the convergence of MOEA/D-DE with Tchebycheff approach adopted the setting $(SF, CR) = (0.5, 0.6)$ is equal to or better than those used other combinations (*SF*, *CR*).
(2) Impacts of penalty factor Θ on the feasibility of solutions: Penalty factor is a key parameter in penalty function method for handling constraints. For $\Theta = 100$, $\Theta = 1000$, $\Theta = 10000$, and $\Theta = 100000$, we have counted the number of feasible solutions in the population on Problem 1 when evolutionary generations reach 50, 100, 150, 200, 250, and 300. The statistical results are displayed in Fig. 12.
It can be observed that MOEA/D-DE is less sensitive to the penalty factor Θ. It also suggests that $\Theta = 1000$ or $\Theta = 10000$ is the best choice for MOEA/D-DE.
(3) Impacts of weight vector adjustment on the distribution of solutions: We compared the commonly used changeless weight vectors with the adjustable weight vectors adopted in MOEA/D-DE with Tchebycheff approach on Problems 1, 5, 6, 8, 9 and 11 in terms of *S*-metric. Fig. 13 displays the box plots of the *S*-metric values among Tchebycheff approach with the changeless weight vectors and Tchebycheff approach with the adjustable weight vectors for Problems 1, 5, 6, 8, 9 and 11. Figs. 14–19 visually show the distributions of the final populations when the commonly changeless weight vectors and the adjustable weight vectors were adopted in MOEA/D-DE with Tchebycheff approach on Problems 1, 5, 6, 8, 9 and 11.

Table 7
The average values of C-metric between the original combination of control factors $(SF, CR) = (0.5, 0.6)$ denoted by A and the other combinations of control factors (SF, CR) denoted by A_i, B_i, D_i, E_i, F_i, G_i $(i = 1, 2, \cdots, 5)$ in MOEA/D-DE with Tchebycheff approach.

	$A_1 = (0.5, 0.1)$	$A_2 = (0.5, 0.3)$	$A_3 = (0.5, 0.5)$	$A_4 = (0.5, 0.7)$	$A_5 = (0.5, 0.9)$
$C(A_i, A)$	0.0149	0.0143	0.0120	0.0364	0.0152
$C(A, A_i)$	0.0158	0.0147	0.0150	0.0391	0.0166
	$B_1 = (0.6, 0.1)$	$B_2 = (0.6, 0.3)$	$B_3 = (0.6, 0.5)$	$B_4 = (0.6, 0.7)$	$B_5 = (0.6, 0.9)$
$C(B_i, A)$	0.0093	0.0113	0.0091	0.0139	0.0219
$C(A, B_i)$	0.0097	0.0180	0.0152	0.0205	1
	$D_1 = (0.7, 0.1)$	$D_2 = (0.7, 0.3)$	$D_3 = (0.7, 0.5)$	$D_4 = (0.7, 0.7)$	$D_5 = (0.7, 0.9)$
$C(D_i, A)$	0.0146	0.0265	0.0192	0.0245	0.0152
$C(A, D_i)$	0.0146	0.0291	1	1	1
	$E_1 = (0.8, 0.1)$	$E_2 = (0.8, 0.3)$	$E_3 = (0.8, 0.5)$	$E_4 = (0.8, 0.7)$	$E_5 = (0.8, 0.9)$
$C(E_i, A)$	0.0079	0.0188	0.0152	0.0099	0.0179
$C(A, E_i)$	0.0185	0.0188	0.0172	0.0132	0.0212
	$F_1 = (0.9, 0.1)$	$F_2 = (0.9, 0.3)$	$F_3 = (0.9, 0.5)$	$F_4 = (0.9, 0.7)$	$F_5 = (0.9, 0.9)$
$C(F_i, A)$	0.0141	0.0162	0.0053	0.0139	0.0079
$C(A, F_i)$	0.0149	0.0169	0.0172	0.0179	0.0179
	$G_1 = (1, 0.1)$	$G_2 = (1, 0.3)$	$G_3 = (1, 0.5)$	$G_4 = (1, 0.7)$	$G_5 = (1, 0.9)$
$C(G_i, A)$	0.0119	0.0107	0.0126	0.0144	0.0139
$C(A, G_i)$	0.0159	0.0116	0.0212	0.0168	0.0199

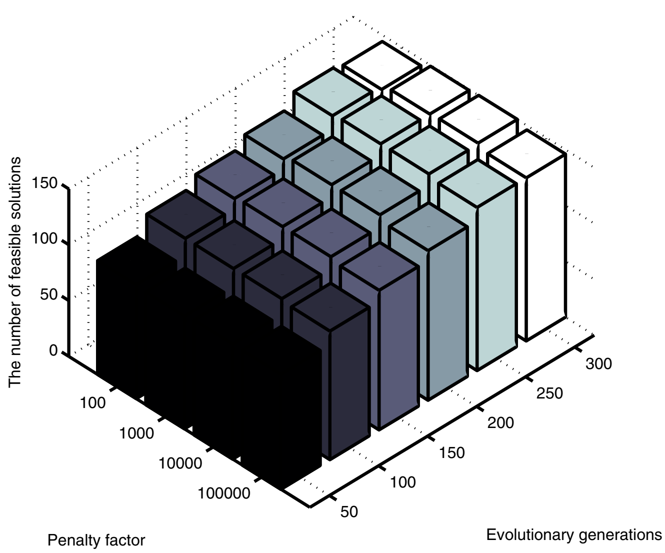


Fig. 12. The number of feasible solutions in the population for different values of penalty factor with different evolutionary generations.

It is clear from Figs. 13–19 the adjustable weight vectors utilized in MOEA/D-DE with Tchebycheff approach can achieve a significantly better distribution in the final Pareto front than the commonly used changeless weight vectors for each instance.

6. Conclusion and future work

For multiobjective bilevel programs with multiple objective functions at the upper level, when the KKT optimality conditions of the lower level optimization problem are satisfied, the original multiobjective two-level programming problem is converted into a multiobjective single-level optimization with the complementarity constraints, and then the smoothing technique is applied to deal with the complementarity constraints. Thus, a multiobjective single-level nonlinear programming problem is obtained. To solve this constrained multiobjective optimization problem, a constrained multiobjective differential evolution algorithm based on decomposition is presented.

The numerical results show the transformation model is simple and easy to solve via the MOEAs. Moreover, MOEA/D-DE with Tchebycheff approach and NSGA-II are effective and robust. Furthermore, the experimental results indicate that MOEA/D-DE with Tchebycheff approach outperforms NSGA-II for most of instances. Also, it is concluded that weighted sum approach may be unsuitable for solving multiobjective optimization at the upper level, even for linear cases.

The proposed solution method is suitable for a wide range of multiobjective bilevel programs with multiple objective functions at the upper level, where the lower level programming problem belongs to linear programming, quadratic programming, differentiable nonlinear programming.

On the basis of the transformation model in this paper, we can further cope with the most complex formulation of multiobjective bilevel programming, i.e. multiple objectives arise in both levels of bilevel programming using these similar strategies in the future.

Acknowledgements

This work was supported by Natural Science Basic Research Plan in Shaanxi Province of China (Program No. 2013JM1022), and the National Natural Science Foundation of China (Grant No. 61401322). The first author gratefully acknowledges the support from China Scholarship Council (Grant No. 201306965006). The authors also gratefully acknowledge the helpful comments and suggestions of the referees, which have improved the presentation.

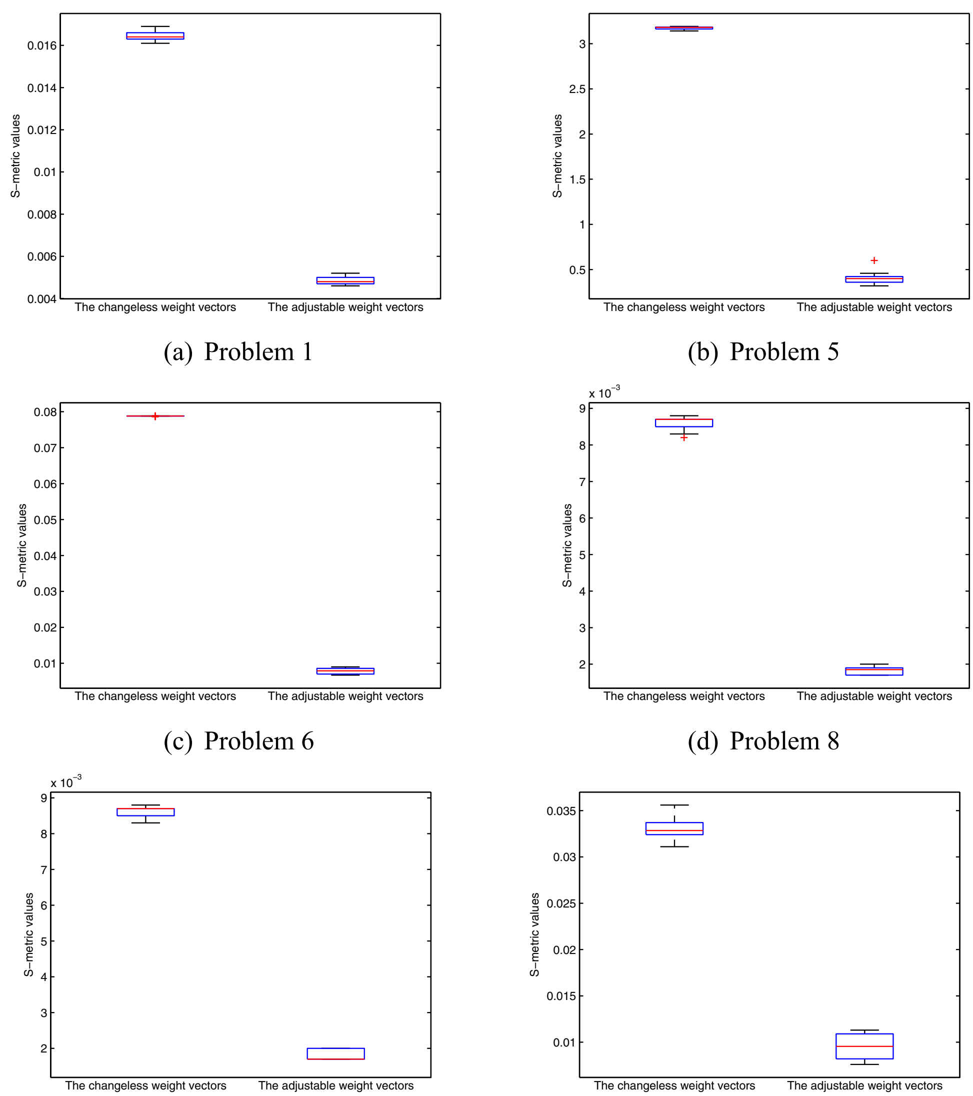
Appendix. MOBLPPs with multiple objectives at the upper level

Problem 1 [58]:.

$$
\begin{array}{ll}
\max\limits_{x} & F(x,y) = \Big((x+2y_2+3)(3y_1+2),\ \ (2x+y_1+2)(y_2+1) \Big) \\
\text{s. t.} & 3x+y_1+2y_2 \le 5,\ \ y_1+y_2 \le 3 \\
\max\limits_{y} & f(x,y) = (y_1+1)(x+y_1+y_2+3) \\
\text{s. t.} & x+2y_1+y_2 \le 2,\ \ 3y_1+2y_2 \le 6, \\
 & x \ge 0,\ \ y_1 \ge 0,\ \ y_2 \ge 0
\end{array}
$$

Problem 2 [26]:.

$$
\begin{array}{ll}
\max\limits_{x} & F(x,y) = \Big(-2x,\ \ -x+5y \Big) \\
\max\limits_{y} & f(x,y) = -y \\
\text{s. t.} & x-2y \le 4,\ \ 2x-y \le 24,\ \ 3x+4y \le 96, \\
 & x+7y \le 126,\ \ -4x+5y \le 65,\ \ x+4y \ge 8, \\
 & x \ge 0,\ \ y \ge 0
\end{array}
$$



(a) Problem 1 (b) Problem 5

(c) Problem 6 (d) Problem 8

(e) Problem 9 (f) Problem 11

Fig. 13. Box plots of the *S*-metric values between different settings of weight vectors..

Problem 3 [26]**:.**

$$
\begin{array}{ll}
\max\limits_{x} & F(x,y) = \Big(2x_1 - 4x_2 + y_1 - y_2,\ \ -x_1 + 2x_2 - y_1 + 5y_2\Big) \\
\max\limits_{y} & f(x,y) = 3y_1 + y_2 \\
\text{s. t.} & 4x_1 + 3x_2 + 2y_1 + y_2 \le 60,\ \ 2x_1 + x_2 + 3y_1 + 4y_2 \le 60, \\
 & x_1,\ x_2,\ y_1,\ y_2 \ge 0
\end{array}
$$

Problem 4 It is modified from the literature [30]**.**

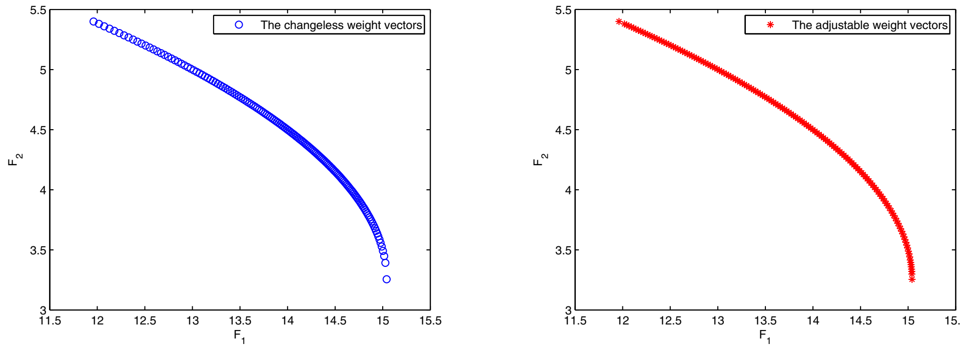
$$
\begin{array}{ll}
\max\limits_{x} & F(x,y) = \Big((y_1 + y_3)(200 - y_1 - y_3),\ (y_2 + y_4) \\
 & (160 - y_2 - y_4)\Big) \\
\text{s. t.} & x_1 + x_2 + x_3 + x_4 \le 40 \\
 & 0 \le x_1 \le 10,\ \ 0 \le x_2 \le 5,\ \ 0 \le x_3 \le 15,\ \ 0 \le x_4 \le 20 \\
\min\limits_{y_1, y_2} & f_1(x,y) = (y_1 - 4)^2 + (y_2 - 13)^2 \\
\text{s. t.} & 0.4y_1 + 0.7y_2 - x_1 \le 0,\ \ 0.6y_1 + 0.3y_2 - x_2 \le 0 \\
 & 0 \le y_1 \le 20,\ \ 0 \le y_2 \le 20 \\
\min\limits_{y_3, y_4} & f_2(x,y) = (y_3 - 35)^2 + (y_4 - 2)^2 \\
\text{s. t.} & 0.4y_3 + 0.7y_4 - x_3 \le 0,\ \ 0.6y_3 + 0.3y_4 - x_4 \le 0 \\
 & 0 \le y_3 \le 40,\ \ 0 \le y_4 \le 40
\end{array}
$$

Problem 5 It is modified from the literature [59]**.**

$$
\begin{array}{ll}
\min\limits_{x} & F(x,y) = \Big(-x - y,\ \ x^2 + (y - 10)^2\Big) \\
\text{s. t.} & 0 \le x \le 15 \\
\min\limits_{y} & f(x,y) = y(x - 30) \\
\text{s. t.} & y - x \le 0,\ \ 0 \le y \le 15
\end{array}
$$

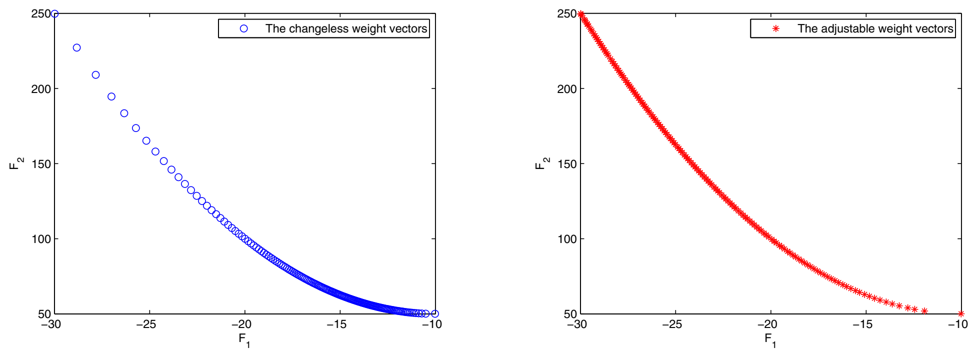
Problem 6 [60]**:.**

$$
\begin{array}{ll}
\min\limits_{x} & F(x,y) = \Big(8x + 4y_1^2 - 2,\ \ 4x - 8y_2 + 1\Big) \\
\text{s. t.} & 1 \le x \le 2 \\
\min\limits_{y_1} & f_1(x,y) = 2y_1^3 - x + 7 \\
\text{s. t.} & x - y_1 - 1 \le 0 \\
\min\limits_{y_2} & f_2(x,y) = x^2 + 2x + y_2^2 - 5 \\
\text{s. t.} & x - y_2 \le 0
\end{array}
$$



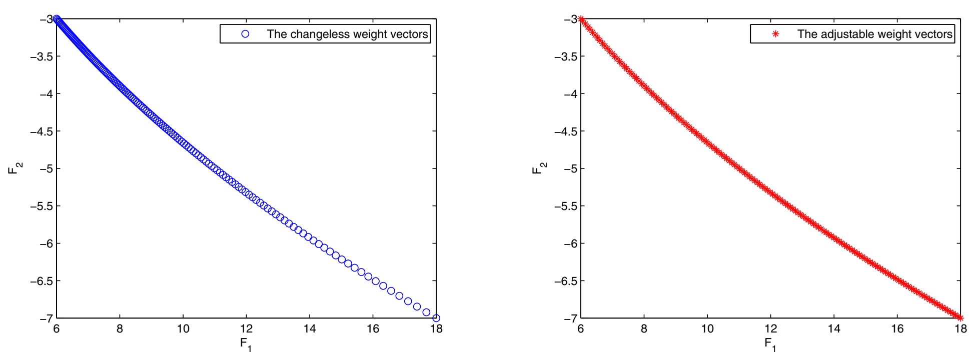
(a) The changeless weight vectors (b) The adjustable weight vectors

Fig. 14. The Pareto front produced by Tchebycheff approach with different settings of weight vectors for Problem 1.



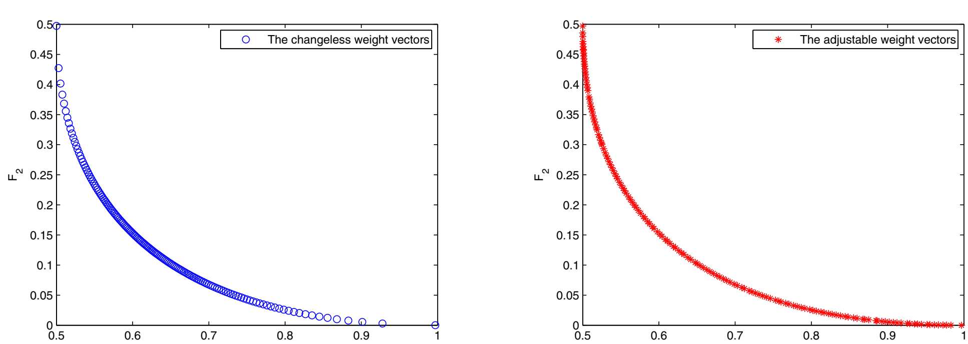
(a) The changeless weight vectors (b) The adjustable weight vectors

Fig. 15. The Pareto front produced by Tchebycheff approach with different settings of weight vectors for Problem 5.



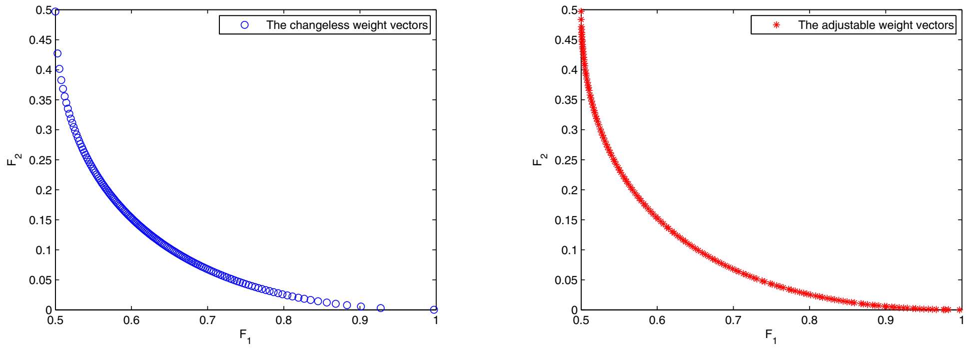
(a) The changeless weight vectors (b) The adjustable weight vectors

Fig. 16. The Pareto front produced by Tchebycheff approach with different settings of weight vectors for Problem 6.



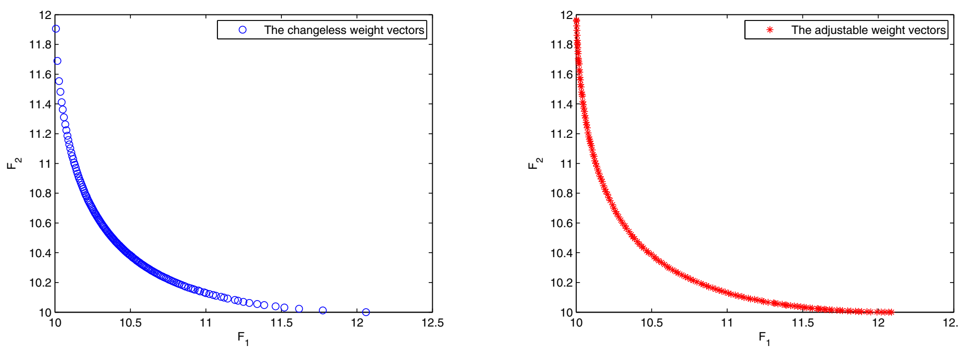
(a) The changeless weight vectors (b) The adjustable weight vectors

Fig. 17. The Pareto front produced by Tchebycheff approach with different settings of weight vectors for Problem 8.



(a) The changeless weight vectors (b) The adjustable weight vectors

Fig. 18. The Pareto front produced by Tchebycheff approach with different settings of weight vectors for Problem 9.



(a) The changeless weight vectors (b) The adjustable weight vectors

Fig. 19. The Pareto front produced by Tchebycheff approach with different settings of weight vectors for Problem 11.

Problem 7 It is modified from the literature [35].

$$\begin{aligned}
\min_x \quad & F(x,y) = \Big(y_1 + y_2^2 + x + \sin^2(y_1+x),\ \cos(y_2)(0.1+x) \\
& \exp\Big(-\frac{y_1}{0.1+y_2}\Big)\Big) \\
\text{s. t.} \quad & 0 \le x \le 10 \\
\min_y \quad & f(x,y) = \frac{(y_1-2)^2+(y_2-1)^2}{4} + \frac{y_2x+(5-x)^2}{16} + \sin\frac{y_2}{10} \\
& + \frac{y_1^2+(y_2-6)^4-2y_1x-(5-x)^2}{80} \\
\text{s. t.} \quad & y_1^2 - y_2 \le 0,\ 5y_1^2+y_2 \le 10,\ y_2 + \frac{x}{6} \le 5,\ y_1 \ge 0
\end{aligned}$$

Problem 8 It is modified from the literature [30].

$$\begin{aligned}
\min_x \quad & F(x,y) = \Big((y_1-1)^2+y_2^2+x^2,\ (y_1-1)^2+y_2^2+(x-1)^2\Big) \\
\min_y \quad & f(x,y) = (y_1-x)^2+y_2^2 \\
\text{s. t.} \quad & -1 \le x, y_1, y_2 \le 2
\end{aligned}$$

Problem 9 It is modified from the literature [30].

$$\begin{aligned}
\min_x \quad & F(x,y) = \Big((y_1-1)^2 + \sum_{i=1}^{13} y_{i+1}^2 + x^2,\ (y_1-1)^2 + \sum_{i=1}^{13} y_{i+1}^2 \\
& +(x-1)^2\Big) \\
\min_y \quad & f(x,y) = (y_1-x)^2 + \sum_{i=1}^{13} y_{i+1}^2 \\
\text{s. t.} \quad & -1 \le x,\ y_i \le 2,\ i = 1,2,\cdots,14
\end{aligned}$$

Problem 10 It is modified from the literature [31].

$$\begin{aligned}
\min_x \quad & F(x,y) = \Big((1-x_1)(1+x_2^2+x_3^2)y,\ x_1(1+x_2^2+x_3^2)y\Big) \\
\min_y \quad & f(x,y) = (1-x_1)(1+x_4^2+x_5^2)y \\
\text{s. t.} \quad & (1-x_1)y + \tfrac{1}{2}x_1y - 1 \ge 0, \\
& -1 \le x_1 \le 1,\ 1 \le y \le 2 \\
& -5 \le x_i \le 5,\ i = 2,3,4,5
\end{aligned}$$

Problem 11.

$$\begin{aligned}
\min_x \quad & F(x,y) = \Big(\sum_{i=1}^{K} \exp(-\tfrac{x_i}{1+|y_i|}) + \sum_{i=1}^{K} \sin(\tfrac{x_i}{1+|y_i|}),\ \sum_{i=1}^{K} \\
& \exp(-\tfrac{y_i}{1+|x_i|}) + \sum_{i=1}^{K} \sin(\tfrac{y_i}{1+|x_i|})\Big) \\
\text{s. t.} \quad & -1 \le x_i \le 1,\ i = 1,2,\cdots,K \\
\min_y \quad & f(x,y) = \sum_{i=1}^{K} \cos(|x_i|y_i) + \sum_{i=1}^{K} \sin(x_i - y_i) \\
\text{s. t.} \quad & x_i + y_i \le 1, i = 1,2,\cdots,K \\
& -1 \le y_i \le 1,\ i = 1,2,\cdots,K
\end{aligned}$$

References

[1] J.F. Bard, Practical Bilevel Optimization: Algorithms and Applications, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998.

[2] S.W. Chiou, A bi-level decision support system for uncertain network design with equilibrium flow, Decis. Support Syst. 69 (2015) 50–58.

[3] S. Dempe, V.V. Kalashnikov, G.A. Pérez-Valdés, N.I. Kalashnykova, Natural gas bilevel cash-out problem: convergence of a penalty function method, Eur. J. Oper. Res. 215 (2011) 532–538.

[4] M.R. Hesamzadeh, M. Yazdani, Transmission capacity expansion in imperfectly competitive power markets, IEEE Trans. Power Syst. 29 (2014) 62–71.

[5] Y. Gao, G. Zhang, J. Lu, H.M. Wee, Particle swarm optimization for bi-level pricing problems in supply chains, J. Global Optim. 51 (2011) 245–254.

[6] B. Liu, Stackelberg-nash equilibrium for multilevel programming with multiple followers using genetic algorithms, Comput. Math. Appl. 36 (1998) 79–89.
[7] H. Li, Y.C. Jiao, F.S. Zhang, L. Zhang, An efficient method for linear bilevel programming problems based on the orthogonal genetic algorithm, Int. J. Innov. Comput. Inf. Control 5 (2009) 2837–2846.
[8] S.R. Hejazi, A. Memariani, G. Jahanshahloo, M.M. Sepehri, Linear bilevel programming solution by genetic algorithm, Computers & Operations Research 29 (2002) 1913–1925.
[9] V. Oduguwa, R. Roy, Bi-level optimization using genetic algorithm, in: Proceedings of IEEE International Conference on Artificial Intelligence Systems, 2002, pp. 123–128.
[10] Y. Wang, Y.C. Jiao, H. Li, An evolutionary algorithm for solving nonlinear bilevel programming based on a new constraint-handing scheme, IEEE Trans. Syst. Man Cybern.-Part C 35 (2005) 221–232.
[11] X. Li, P. Tian, X. Min, A hierarchical particle swarm optimization for solving bilevel programming problems, in: Artificial Intelligence and Soft Computing-ICAISC 2006, in: Lecture Notes in Computer Science, 4029, 2006, pp. 1169–1178.
[12] Y. Wang, H. Li, C. Dang, A new evolutionary algorithm for a class of nonlinear bilevel programming problems and its global convergence, INFORMS J. Comput. 23 (2011) 618–629.
[13] J.S. Angelo, E. Krempser, H.J.C. Barbosa, Differential evolution for bilevel programming, in: In 2013 IEEE Congress on Evolutionary Computation (CEC),, 2013, pp. 470–477.
[14] H. Li, L. Zhang, A differential evolution with two mutation strategies and a selection based on an improved constraint-handling technique for bilevel programming problems, Math. Prob. Eng. 2014 (2014) 1–16.
[15] S. Dempe, Foundations of Bilevel Programming, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2002.
[16] L.N. Vicente, P.H. Calamai, Bilevel and multilevel programming: a bibliography review, J. Global Optim. 5 (1994) 291–306.
[17] S. Dempe, Annotated bibliography on bilevel programming and mathematical programs with equilibrium constraints, Optimization 52 (2003) 333–359.
[18] B. Colson, P. Marcotte, G. Savard, Bilevel programming: a survey, 4OR 3 (2005) 87–107.
[19] B. Colson, P. Marcotte, G. Savard, An overview of bilevel optimization, Ann. Oper. Res. 153 (2007) 235–256.
[20] G. Zhang, J. Lu, Y. Gao, Multi-Level Decision Making: Models, Methods and Applications, Springer, 2015.
[21] J. Lu, J. Han, Y. Hu, G. Zhang, Multilevel decision-making: a survey, Inf. Sci. 346 (2016) 463–487.
[22] Y. Yin, Multiobjective bilevel optimization for transportation planning and management problems, J. Adv. Transp. 36 (2002) 93–105.
[23] J. Xu, Y. Tu, Z. Zeng, A nonlinear multiobjective bilevel model for minimum cost network flow problem in a large-scale construction project, Math. Prob. Eng. 2012 (2012) 1–40.
[24] J.J. Ye, Necessary optimality conditions for multiobjective bilevel programs, Math. Oper. Res. 36 (2011) 165–184.
[25] M.J. Alves, Using MOPSO to solve multiobjective bilevel linear problems, in: Swarm Intelligence, in: Lecture Notes in Computer Science, 7461, 2012, pp. 332–339.
[26] M.J. Alves, S. Dempe, J.J. Júdice, Computing the pareto frontier of a bi-objective bi-level linear problem using a multiobjective mixed-integer programming algorithm, Optimization 61 (2012) 335–358.
[27] H.I. Calvete, C. Galé, Linear bilevel programs with multiple objectives at the upper level, J. Comput. Appl. Math. 234 (2010) 950–959.
[28] Z. Ankhili, A. Mansouri, An exact penalty on bilevel programs with linear vector optimization lower level, Eur. J. Oper. Res. 197 (2009) 36–41.
[29] H.I. Calvete, C. Galé, On linear bilevel problems with multiple objectives at the lower level, Omega 39 (2011) 33–40.
[30] K. Deb, A. Sinha, Solving bilevel multi-objective optimization problems using evolutionary algorithms, in: 5th International Conference on Evolutionary Multi-Criterion Optimization, EMO 2009, in: Lecture Notes in Computer Science, 5467, 2009a, pp. 110–124.
[31] K. Deb, A. Sinha, Constructing test problems for bilevel evolutionary multi-objective optimization, in: IEEE Congress on Evolutionary Computation, CEC'09, 2009b, pp. 1153–1160.
[32] Y. Gao, G. Zhang, J. Ma, J. Lu, A λ-cut and goal-programming-based algorithm for fuzzy-linear multiple-objective bilevel optimization, IEEE Trans. Fuzzy Syst. 18 (2010) 1–13.
[33] G. Zhang, J. Lu, Fuzzy bilevel programming with multiple objectives and cooperative multiple followers, J. Global Optim. 47 (2010) 403–419.
[34] I. Nishizaki, M. Sakawa, Stakelberg solutions to multiobjective two-level linear programming problems, J. Optim. Theor. Appl. 103 (1999) 161–182.
[35] G. Eichfelder, Multiobjective bilevel optimization, Math. Prog. 123 (2010) 419–449.
[36] T. Zhang, T. Hu, Y. Zheng, X. Guo, An improved particle swarm optimization for solving bilevel multiobjective programming problem, J. Appl. Math. 2012 (2012) 1–13.
[37] F. Facchinei, H. Jiang, L. Qi, A smoothing method for mathematical programs with equilibrium constraints, Math. Prog. 85 (1999) 107–134.
[38] K. Miettinen, Nonlinear multiobjective optimization, MAA:Kluwer, Boston, 1999.
[39] Q. Zhang, H. Li, MOEA/D: a multiobjective evolutionary algorithm based on decomposition, IEEE Trans. Evol. Comput. 11 (2007) 712–731.
[40] H. Li, Q. Zhang, Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II, IEEE Trans. Evol. Comput. 13 (2009) 284–302.
[41] K. Deb, S. Agrawal, A. Pratap, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2002) 182–197.
[42] Q. Zhang, A. Zhou, Y. Jin, RM-MEDA: A regularity model-based multiobjective estimation of distribution algorithm, IEEE Trans. Evol. Comput. 21 (2008) 41–63.
[43] Y.W. Leung, Y. Wang, Multiobjective programming using uniform design and genetic algorithm, IEEE Trans. Syst. Man Cybern.-Part C 30 (2000) 293–304.
[44] E. Zitzler, L. Thiele, Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach, IEEE Trans. Evol. Comput. 3 (1999) 257–271.
[45] C.A.C. Coello, An updated survey of GA-based multiobjective optimization techniques, ACM Comput. Surv. 32 (2000) 109–143.
[46] K.C. Tan, T.H. Lee, E.F. Khor, Evolutionary algorithm with dynamic population size and local exploration for multiobjective optimization, IEEE Trans. Evol. Comput. 5 (2001) 565–588.
[47] H. Lu, G.G. Yen, Rank-density-based multiobjective genetic algorithm and benchmark test function study, IEEE Trans. Evol. Comput. 7 (2003) 325–343.
[48] E. Zitzler, L. Thiele, M. Laumanns, C.M. Fonseca, V.G.d. Fonseca, Performance assessment of multiobjective optimizers: An analysis and review, IEEE Trans. Evol. Comput. 7 (2003) 117–132.
[49] P.A.N. Bosman, D. Thierens, The balance between proximity and diversity in multiobjective evolutionary algorithms, IEEE Trans. Evol. Comput. 7 (2003) 174–188.
[50] Y.Y. Tan, Y.C. Jiao, H. Li, X.K. Wang, MOEA/D-SQA: a multi-objective memetic algorithm based on decomposition, Eng. Optim. 44 (2012a) 1095–1115.
[51] Y.Y. Tan, Y.C. Jiao, H. Li, X.K. Wang, A modification to MOEA/D-DE for multiobjective optimization problems with complicated pareto sets, Inf. Sci. 213 (2012b) 14–38.
[52] Y.Y. Tan, Y.C. Jiao, H. Li, X.K. Wang, MOEA/D+uniform design: A new version of MOEA/d for optimization problems with many objectives, Comput. Oper. Res. 40 (2013) 1648–1660.
[53] Y.P. Wang, C.Y. Dang, Improving multiobjective evolutionary algorithm by adaptive fitness and space division, in: ICNC05, August, in: LNCS 3612, 27–29, 2005, pp. 392–398. Changsha,China
[54] J. Schott, Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA, 1995 Master Thesis.
[55] H. Li, L. Zhang, Y.C. Jiao, An interactive approach based on a discrete differential evolution algorithm for a class of integer bilevel programming problems, Int. J. Syst. Sci. 47 (2016) 2330–2341.
[56] R. Storn, K. Price, Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces,, J. Global Optim. 11 (1997) 341–359.
[57] S. Das, P.N. Suganthan, Differential evolution: a survey of the state-of-the-art, IEEE Trans. Evol. Comput. 15 (2011) 4–31.
[58] S.R. Arora, Ritu arora, indefinite quadratic bilevel programming problem with multiple objectives at both levels, Int. J. Optim. 1 (2009) 318–327.
[59] C.X. Teng, L. Li, H.B. Li, A class of genetic algorithms on bilevel multiobjective decision making problem, J. Syst. Sci. Syst. Eng. 9 (2000) 290–296.
[60] F. Xu, Y. Guo, A feasible direction method of multiobjective nonlinear two-level optimization problems with leader-follower structure, J. Southwest Jiaotong Univ. 31 (1996) 433–439.