Journal of Fundamental and Applied Sciences

ISSN 1112-9867

Available online at http://www.jfas.info

Research Article

Special Issue

FINDING NON-DOMINATED VECTORS IN MULTI-OBJECTIVE LINEAR PROGRAMMING PROBLEMS

S. A. Gheidari

Master of Applied Mathematics, Operations Research orientation Karaj Branch, Islamic Azad University, Karaj, Iran

Published online: 1 February 2018

ABSTRACT

In this article, we aim to propose an algorithm to generate non-dominated vectors in multi-objective linear programming problems. In other words, in the case where all variables are integer, we introduce an algorithm to produce a series of non-dominated vectors. Then in the next section, the algorithm is improved. Therefore, two modified versions of the algorithm are introduced. The two versions by reducing number of constraints and binary variables result in better computational performance.

Keyword: Integer programming, multi-objective programming, parametric programming

INTRODUCTION

Generally, finding all non-dominated vectors in multi-objective combinatorial optimization problems (MOCO) is a very complex and difficult task. In this section, by fitting smooth super-surfaces, we try to approximate non-dominated boundaries in MOCO problems. For a given problem, we fit the mentioned super-surface using a non-dominated reference vector. Using such approximation and by brief calculations, we can find a neighborhood of the preferred area, which includes non-dominated vectors. Further arithmetic operations can be found within this area, i.e. where the decision maker prefers to continue and obtain non-dominated vectors.

Author Correspondence, e-mail: abdi.sami28@yahoo.com

doi: http://dx.doi.org/10.4314/jfas.v10i2s.56

Of course, finding all non-dominated vectors, because of the large number of them, can be very difficult. One way to solve this problem is combining this method with an interactive decision-making method so as to achieve a desirable non-dominated vector. However, this interactive method can also lead to finding a significant number of non-dominated vectors and therefore, this method can be inefficient too. But if the introducedL_p-surface determines well the non-dominated boundaries then we can use an interactive surface to find the best decision-making solution on thisL_p-surface.

So, first we choose a discrete set on L_p-surface and after we found the best solution on L_p-surface, we solve the combinatorial optimization problem and this way we generate several real non-dominated vectors that are close to this best theoretical solution located on L_p-surfaces. The resulting non-dominated vectors can be presented to decision-makers so that after the final examination, the desired solution to be selected.

The fitting technique of a super-surface to approximate a non-dominated boundary

Consider the following multi-objective optimization problem:

$$(VM) \qquad Max \qquad \{z_1(x), z_2(x), \dots, z_q(x)\} \qquad s.t. \qquad x \in X.$$

Suppose$(z_1^{IP}, z_2^{IP}, \dots, z_q^{IP})$represents the ideal point corresponding to the problem *VM*, i.e.:

$$z_i^{IP} = \max_{x \in X} z_i(x).$$

Suppose also that $(z_1^{NP}, z_2^{NP}, \dots, z_q^{NP})$ is the corresponding point for the problem *VM*, which is defined as follows:

$$z_i^{NP} = \min_{x \in E} z_i(x),$$

Where *E*is the set of efficient solutions. Finding $z_i^{NP}$is not an easy task and instead we can use a lower bound that results from solving$\min_{x \in X} z_i(x)$.

Corresponding to values of$(z_1, z_2, \dots, z_q)$, we obtained the following scaled values

$$\left(z_1', z_2', \dots, z_q'\right) = \left(\frac{z_1 - z_1^{IP}}{z_1^{NP} - z_1^{IP}}, \frac{z_2 - z_2^{IP}}{z_2^{NP} - z_2^{IP}}, \dots, \frac{z_q - z_q^{IP}}{z_q^{NP} - z_q^{IP}}\right)$$

So that for each for$i = 1, \dots, q$, we have $0 \le z_i' \le 1$. Thetarget scaled values are better values, because $z_i$approaches to the ideal target value, i.e.z_i^{IP}, whilez_i'closes to zero. So we minimize the target scaled values. These considerations are valid both for minimization and maximization problems.

Non-dominated vectors have the feature that improving each of the objective functions requires that at least deteriorate one other objective functions. Due to this feature, we assume a set of non-dominated vertex vectors as follows:

$$S = \{(0,1,\dots,1),(1,0,1,\dots,1),\dots,(1,1,\dots,1,0)\}.$$

Then we fit a super-surface on the scaled vectors of set *S*. In case of having two objectives, each vector in set $S = \{(0,1),(1,0)\}$corresponds to a real non-dominated vector. But for more than two objectives, it is only an approximation and vectors of set *S* are not necessarily non-dominated vectors.

The super-page passing from of all vectors of the set *S* is defined as follows:

$$\left(1 - z_1^{'}\right)^p + \left(1 - z_2^{'}\right)^p + \cdots + \left(1 - z_q^{'}\right)^p = 1, \qquad p > 0.$$

This super-surface is called L_p-surface. If we can find an appropriate value for p so that L_p-surface is close enough to the non-dominated vectors, then we can approximate the non-dominated boundary using this super-surface. Let $(r_1, r_2, \dots, r_q)$be the non-dominated scaled vector that we have chosen it as the reference vector. If you find a value for *p* satisfying the following relation

$$(1 - r_1)^p + (1 - r_2)^p + \cdots + \left(1 - r_q\right)^p = 1$$

, then the L_p-surface will pass through this reference point.

To select a central reference vector, we solve an expanded weighted T-Chebyshev problem. In other words, we solve the following problem:

$$\min_{x \in X}\left(\max_{i=1,2,\dots,q} \left(z_i^{IP} - z_i(x)\right) - \epsilon \sum_{i=1}^{q} z_i(x)\right),$$

Where *ε* is a positive and enough small value. This problem is equivalent to:

$$\left(P_{ref}\right) \quad Min \quad \alpha - \epsilon \sum_{i=1}^{q} z_i(x) \quad s.t. \quad z_i^{IP} - z_i(x) \leq \alpha, \quad \forall i \quad x \in X.$$

As we know the expanded weighted T-Chebyshev problem, i.e.$\left(P_{ref}\right)$,will produce a non-dominated vector.

Whenever we have a reference vector$\left(r_1, r_2, \dots, r_q\right)$, then we will be looking for a value for *p* so that satisfying the relation$\sum_{i=1}^{q}(1 - r_i)^p = 1$. If solving the intended combinatorial problem is not difficult, we can produce several reference vector. This realizes by modifying problem ($\mathrm{P_{ref}}$) and solving problem(P_w) as follows:

$$(P_w) \qquad Min \quad \alpha - \epsilon \sum_{i=1}^{q} z_i(x) \;\; s.t. \quad w_i(z_i^{IP} - z_i(x)) \le \alpha, \quad \forall i \quad x \in X,$$

where$w = (w_1, w_2, \dots, w_q)$ arethe weight vectors assigned to the objective functions. Examining several different weight vectors and solving problem (P_w)for each of these vectors, we can obtain the non-dominated vectors from different parts of non-dominated boundary.

2-1 A way to find the set of non-dominated vectors in integer multi-objective linear problems

2-1-1 Basic definitions and propositions

Generally, an integer multi-objective linear problem is defined in the following forum:

$$(P): \quad Max\,\{Cx: Ax = b, x \ge 0, x \in Z^n\},$$

Where$C \in Z^{p\times n}$, ‹$A \in R^{m\times n}$and $b \in R^m$. In this problem, *Cx* represents the objective function *p*. *Ax* = *b* contains *m* linear constraints and *x* represents *n*integer variable. Feasible region of problem *P*is shown with *F* (*P*).

Usually since objective functions are in conflict with each other, there is no optimal (maximum) solution for the above problem. Therefore, for such problems, non-dominated vectors are defined.

Definition 3-1. Feasible solution x^* for problem P is called an efficient solution if there is no other feasible solution such as *x*such that$Cx \ge Cx^*$and at least one of the inequalities hold as strict. In this case, vector $Cx^*$is called a non-dominated vector.

Proposition 3-2. If x^* is an optimal solution for the following single objective problem

$$Max\ \{\lambda^T Cx: x \in S\}$$

Then there is a $\lambda \in R^p$ (λ>0), for which x^* is an efficient solution to the following problem

$$Max\ \{Cx: x \in S\}$$

Efficient solutions that are optimal for the corresponding parametric problem (for a$\lambda \in R^p$and $\lambda > 0$), are called supported efficient solutions. Unlike multi-objective linear programming, in integer multi-objective linear programming, solutions are not necessarily supported and efficient. In other words, there are efficient solutions that are not optimal for any $\lambda > 0$.However, as we see in the next proposition, by eliminating the known efficient solutions and also solution get dominated by them, new efficient solutions (supported or non-supported) can be generated.

Proposition 3-3. Let$x^1, x^2, \dots, x^l$, are efficient solutions for problem (P) and

$$D_s = \{x \in Z^n: Cx \leq Cx^s\}.$$

If x^* is an efficient solution to the following integer multi-objective problem

$$(MOP_l): \quad Max\,\{Cx: x \in F(P) - \cup_{s=1}^{l} D_s\}$$

then x^* is an effective solution to problem P. In addition, if the problem $(MOP_l)$is infeasible, then $\{Cx^s\}_{s=1}^{l}$ is the set of all non-dominated vectors of problem P.

Proof. Let a solution $x' \in F(P)$exists such that $Cx^* \leq Cx'$and at least one of the inequalities is hold as strict. Since $x^*$is an efficient solution to the problem(MOP_l), then x' does not belong to the set $F(MOP_l)$andwe have$x' \in\cup_{s=1}^{l} D_s$.Thus, for a$k \in \{1, \dots, l\}$, we have $x' \in D_k$and according to definition ofD_k, we have $Cx' \leq Cx^s$. But since$Cx^* \leq Cx' \leq Cx^k$, thus we have$x^* \in D_k$, which is in contradiction with $x^* \in F(P) - \cup_{s=1}^{l} D_s$x.

On the other hand, if(MOP_l), is infeasible then $F(P) \subseteq\cup_{s=1}^{l} D_s$, and for each $x \in F(P)$, there exists a x^k so that $Cx \leq Cx^k$.Therefore either $Cx = Cx^k$, which results in $Cx = Cx^k$, or $Cx \leq Cx^k$ so that at least one of the inequalities hold as strict, which in this case, vector Cx is dominated by Cx^k.So, proof of the proposition is complete.

Result 3-4. Let $x^1, x^2, \dots, x^l$are efficient solutions to the problem (P) and

$$D_s = \{x \in Z^n: Cx \leq Cx^s\}.$$

If x^* for a $\lambda \in R^p$, which $(\lambda > 0)$, is the efficient solution of the following problem

$$(PN_l): \quad Max\,\{\lambda^T Cx: x \in F(P) - \cup_{s=1}^{l} D_s\}$$

thenx^* is the efficient solution to the problem (P).

In the next section, in order to implement the above ideas, alinear version of problem $(PN_l)$is considered.

2-1-2 Implementation method

The results obtained in the previous section can be used to create an algorithm for multi-objective integer-bounded linear problems. After selecting a weight vector such as $\lambda> 0$, the first step in the algorithm is solving the following integer linear problem:

$$(P_0): \quad Max\ \{\lambda^T Cx: Ax = b, x \geq 0, x \in Z^n\}.$$

If the problem is infeasible, then problem (*P*) is infeasible. Otherwise, an optimal solution like x^1can be found that according to Proposition 3.2 is an efficient solution to the problem (*P*). Then a series of problems (P_l)will be solved, in which gradually the number of constraints will be increased. After performing *l*steps of the algorithm, if problem $(P_{l-1})$is infeasible, then the algorithmwill stops; otherwise, a new efficient solution like$x^l$is obtained and the problem (P_l) creates by eliminating all the solution satisfying $Cx \leq Cx^l$in the feasible area of problem(P_{l-1}).Removal of a solutions realizes by adding the following constraints to Problem (P_{l-1}):

$$(Cx)_k \geq ((Cx^l)_k + 1)y_k^l - M_k(1 - y_k^l), \qquad k = 1,2, \dots, p,$$

$$\sum_{k=1}^{p} y_k^l \geq 1,$$

$$y_k^l \in \{0,1\}, \qquad k = 1,2, \dots, p,$$

where, $-M_k$ is a lower bound for the *k*-th objective function on the feasible area, e.g. if all the objective functions are non-negative, value of M_k for each k can be equal to 0. Adding these constraints is equivalent to removing area of D_l from the feasible area and thus Problem $(P_l)$is a linear version of(PN_l), and each optimal solution of the problem is also an efficient solution to the problem (*P*):

$$(P_l): \quad Max \quad \lambda^T Cx$$

$$s.t.$$

$$Ax = b,$$

$$(Cx)_k \geq ((Cx^s)_k + 1)y_k^s - M_k(1 - y_k^s), s = 1, \dots, l, \quad k = 1, \dots, p,$$

$$\sum_{k=1}^{p} y_k^s \geq 1, \qquad s = 1, \dots, l,$$

$$y_k^l \in \{0,1\}, \qquad s = 1, \dots l, \quad k = 1,2, \dots, p,$$

$$x \geq 0, \qquad x \in Z^n.$$

For large-scale problems, it may be infeasible to find all non-dominated vectors. Inthese cases, a subset of non-dominated vectors can be created as representative. To do so, problem (P_l)can be modified as follows:

$$(P_l):\quad Max \quad \lambda^T Cx$$
$$s.t. \quad Ax = b,$$
$$(Cx)_k \geq ((Cx^s)_k + f_k)y_k^s - M_k(1 - y_k^s), s = 1, \dots, l, \quad k = 1, \dots, p,$$
$$\sum_{k=1}^{p} y_k^s \geq 1, \qquad s = 1, \dots, l,$$
$$y_k^l \in \{0,1\}, \qquad s = 1, \dots l, \quad k = 1,2, \dots, p,$$
$$x \geq 0, \qquad x \in Z^n,$$

where, $f_k$represents the minimum benefit that must be applied on the k-objective function to the new non-dominated vector to be considered.

If $f_k = 1$for every k = 1, ...,p, and entries of the cost matrix Call are integer, then the generates all the non-dominated vectors, but does not generates necessarily all the efficient solutions. However, the algorithm can be used in the case that all entries of matrix are real numbers, provided that for the difference between each pair of the objective a lower bound is given.

2-1-3 Computational results

Results obtained for the allocation problem show that although the special structure of allocation problem is not used, the algorithm in this problem has a better performance than in the knapsack problem so that the number of simplex iterations as well as some other criteria have decreased compared to the number of non-dominated vectors.

2-1-4 Generating all the efficient solutions for binary problems

Although the algorithm proposed in Section 2.1.1 generates all the non-dominated vectors, but does not generate all of its efficient solutions. Whenever a new efficient solution like $x^*$is found, the set of all x that hold $Cx \leq Cx^*$,must be removed from the feasible area. With this action, not only inefficient solution will be deleted, but also efficient solutions such as $Cx = Cx^*$will also be deleted. Nevertheless, for each target non-dominated vector, an efficient solutionon behalf of a class of equivalent solutions generates. To generate all efficient solutions,more complex operations will be required. In the following, one way is presented for the case in which all the decision variables are binary.

For every efficient solution likex^*, we define the following sets:

$$K_0(x^*) = \{j: x_j^* = 0\}$$

, and

$$K_1(x^*) = \{j: x_j^* = 1\}.$$

Then, we solve the following problem:

$$\begin{aligned} (Q):\quad & Max \quad 0 \\ & s.t. \quad Ax = b, \\ & Cx = Cx^*, \\ & \sum_{j\in K_1(x^*)} x_j - \sum_{j\in K_0(x^*)} x_j \le |K_1(x^*)| - 1, \\ & x \ge 0, \quad x \in Z^n. \end{aligned}$$

Lemma 3-5. All the binary vectors satisfy the constraint

$$\sum_{j\in K_1(x^*)} x_j - \sum_{j\in K_0(x^*)} x_j \le |K_1(x^*)| - 1$$

except than x^*.

Result 3-6. To generate all the efficient solutions,we do as follows: in each iteration of the algorithm, after generating the efficient solution ofx^*, we solve Model (*Q*). Thus, a new efficient solution (from the same class) like x^{**}, if available, would be generated. Then we add theconstraint

$$\sum_{j\in K_1(x^{**})} x_j - \sum_{j\in K_0(x^{**})} x_j \le |K_1(x^{**})| - 1$$

to the Problem (*Q*), and continue the process until the model (*Q*) become infeasible. In this step,allthe efficient solutions like *x* that satisfy $Cx = Cx^*$ are obtained. Now we return to the next step of the algorithm (Section 2.1.1) and we run the next iterations.

2-2 Modified techniques of finding all non-dominated vectors in multi-objective integer linear problems

In this section, two exact algorithm are presented to findthe set of non-dominated vectors in multi-objective integer linear problems. The algorithms can be considered as modified versions of the algorithm presented in Section 2.1.

Consider the following the general multi-objective programming problem:

$$(P): \quad Max \ \{z_1(\mathbf{x}), z_2(\mathbf{x}), \dots, z_p(\mathbf{x})\}$$
$$\text{s.t.}\boldsymbol{x} \in X$$

where, z_i(x)are the functions, p is the number of criteria or objective functions, $x$is a decision variable, and X represents the feasible region. The set of all the non-dominated vectors of problem is called non-dominated boundary.

The algorithm presented in Section 2.1, with a positive weight vector such as λ, ran and solved the following problem:

$$(P_\lambda): \quad Max \ \sum_{j=1}^{p} \lambda_j \, z_j(\mathbf{x})$$

s.t.$\mathbf{x} \in$ X.

After finding a new non-dominated vector, Problem $(P_\lambda)$by adding a binary variable p and $p + 1$ new constraints become updated. If n non-dominated vector was found then Problem(P_λ^n),which was defined as follows, is solved and generates n+1-th non-dominated vector (assuming allz_{tj} for each$t = 1, \dots, n$ andj $= 1, \dots,$ p have integer values):

$$(P_\lambda^n): \quad Max \quad \sum_{j=1}^{p} \lambda_j \, z_j(\mathbf{x})$$

$$s.t. \quad Ax = b,$$

$$z_j(\mathbf{x}) \geq (z_{tj} + 1) y_{tj} - M(1 - y_{tj}), \qquad \forall j \quad \forall t,$$

$$\sum_{j=1}^{p} y_{tj} \geq 1, \qquad \forall t,$$

$$y_{tj} \in \{0,1\}, \qquad j = 1, \dots, p, \quad t = 1,2, \dots, n,$$

□ ∈ X.

The Problem(□$_{□}^{□}$), vector □$^{t} = (z_{t1}, z_{t2}, \ldots, z_{tp})$indicates *t*-thnon-dominated vector. *M* is a positive constant with enough large value that is a lower bound for □$_{□}$(□)and □$_{□□}$is a binary variable that taking value of 1 causes the constraint □$_{□}$(□) ≥ (□$_{□□}$ + *1*) to be satisfied. Constraints $\sum_{□=1}^{□}$ □$_{□□}$ ≥ *1*cause at least one of the entries of the new obtained target vectors to be greater in its *t*-thcorresponding entry than the non-dominated vector and this causes that the new obtained non-dominated vector to differ from all the previous non-dominated vectors. The mentioned algorithm continues adding new binary variables and constraints, i.e. updating Problem(□$_{□}^{□}$)), until the feasible region of the Problem gets empty.

2-2-1 The first modified algorithm

In this section, the algorithm presented in Section 2-1is modifiedby reducing the number of binary variables and new constraints that were used to update the Problem(□$_{□}^{□}$).

In this algorithm, first, one of the objective functions is selected randomly. Let, for example, the *m*-th objective function is selected; consider the following problem:

(□$_{□}^{0}$): □□□ □$_{□}$(□) + □ $\sum_{□≠□}$ □$_{□}$(□)

s.t.□ ∈ X,

where*ε* is a positive constant with a small enough value. The problem generates a non-dominated vector in which, *m*-th objective function is maximized. In addition, the positive value of *ε* prevents generation of a poor non-dominated vector.

Let □$^{1} = (z_{11}, z_{12}, \ldots, z_{1p})$ is the non-dominated vector corresponding to the optimal solution of Problem(□$_{□}^{0}$). According to the structure of the Problem, the *m*-th component of vector□^{1}, i.e. the□$_{1□}$, is equal to value of the *m*-th objective function on the feasible region. Assuming that now *n*-1 non-dominated vector are found, which the *m*-thcomponent of each of them is greater than or equal□$_{□□}$, let □$^{n} = (z_{n1}, z_{n2}, \ldots, z_{np})$is the non-dominated vector thatis obtained in *n*-th iteration. Vectors□^{t} for □ = *1*, …, □ − *1*include all of the non-dominated vectors that their *m*-component is strictly greater than□$_{□□}$, but it is feasible that it may not include all non-dominated vectors that their *m*-component is equal to□$_{□□}$.The rest of non-dominated vectors that their *m*-component is equal to□$_{□□}$ will be generated in the next iterations and before producing any of non-dominated vector that its *m*-thcomponent is smaller than of□$_{□□}$.Finally, all non-dominated vectors□^{n} for □ $= 1, \dots,$ □, with an ascending order with respect to the value of *m*-th objective function will be generated. The set of non-dominated vectors that are obtained in the first *n* iterations are shown by□$^{□}$ $= \{$□$^{t}: 1 \leq$ □ $\leq$ □$\}$.

By solving Problem(□$^{0}_{□}$), non-dominated vectors in which *m*-th objective function has their best value would obtain. The following proposition asserts that having *n*non-dominated vectors with the best *m*-th objective function value, we can obtain the non-dominated vector that provides *n*+1-th good value for the objective function.

Proposition 3-7. Let *ε* is a positive and small enough value and *M* is a positive and large enough value. If we have all the non-dominated vectors belonging to the set□$^{□}$ $= \{$□$^{t}: 1 \leq$ □ $\leq$ □$\}$, then Problem(□$^{□}_{□}$)is defined as follows so as to generate the non-dominated vector □$^{n+1}$ $= (z_{(n+1)1}, z_{(n+1)2}, \dots, z_{(n+1)p})$so that$z_{(n+1)m} \leq z_{tm}$for each □ $= 1, \dots,$ □.If Problem (□$^{□}_{□}$) is infeasible, then □$^{□}$ includes all non-dominated vectors of the main Problem.

(□$^{□}_{□}$): □□□ □$_{□}$(□) + □ $\sum_{□ \neq □}$ □$_{□}$(□)

s.t.□$_{□}$(□) $\geq$ (□$_{□□}$ + 1)□$_{□□}$ − □(1 − □$_{□□}$), ∀□ ≠ □ ∀□,

$\sum_{□ \neq □}$ □$_{□□}$ = 1, ∀□,

□$_{□□}$ ∈ {0,1}, □ = 1,2, …, □, □ = 1, …, □ □ ≠ □,

□ ∈ X.

Proof. Let $n = 1$, where we have only non-dominated vector □^{1} $= (z_{11}, z_{12}, \dots, z_{1p})$at our disposal. Since $\sum_{□ \neq □}$ □$_{1□}$ = 1, exactly one of *p*-1 constraints □$_{□}$(□) ≥ (□$_{1□}$ + 1)will hold as equality or would be active, and given the sufficiently large value of *M*, the rest of constraints will be redundant. Thus, in the new non-dominated vector, at least value of one of the objective functions in vector □^{1}will be strictly larger, which guarantees variety of non-dominated vectors obtained. Since our goal is to maximize*m*-th objective function we believe that one different non-dominated vector will be obtained, therefore we obtain the non-dominated vector□^{2}, which among the feasible solution of Problem(□$^{1}_{□}$), its *m*-thobjective function value is the highest. Since the feasible region of Problem (□$^{1}_{□}$) is the subset of the feasible region of Problem(□$^{0}_{□}$),so the optimal value of Problem$\left(□_{□}^{1}\right)$is smaller or equal to the optimal value of Problem$\left(□_{□}^{0}\right)$.Since ε is small enough, so$□_{2□} \leq □_{1□}$. If $\left(□_{□}^{1}\right)$is infeasible, then, we conclude that there is only one non-dominated vector. Now let$□ > 1$. In this case, similarly, the first and second class of constraints of Problem$(□_{□}^{□})$ guarantee that the generated efficient vector is different from all non-dominated vectors belonging to set$□_{□}$.Thus, the non-dominated vector $□^{n+1}$ will obtain that among the feasible solutions of Problem$(□_{□}^{□})$,its *m*-thobjective function value is the highest. Since the feasible region of Problem$(□_{□}^{□})$ is a subset of the feasible region of Problem$\left(□_{□}^{□-1}\right)$, thus the optimum of Problem$(□_{□}^{□})$is smaller or equal to the optimal value of Problem$\left(□_{□}^{□-1}\right)$andsimilar to before, since ε is small enough, thus, $□_{(□+1)□} \leq □_{□□}$. If $(□_{□}^{□})$infeasible, then, we conclude that the set $□^{□} = \{□^{t}: 1 \leq □ \leq □\}$ includes all non-dominated vectors of the main problem. This way, proof of the proposition is complete.

The above results indicate thatto generate all non-dominated vectors of the original Problem (*P*), it is enough to solve Problem $(□_{□}^{□})$in an iterative manner until the stop condition, i.e. infeasibility of the Problem, is established.

Proof. Let $n = 1$, where we have only non-dominated vector $□^{1} = \left(z_{11}, z_{12}, \dots, z_{1p}\right)$at our disposal. Since $\sum_{□\neq□} □_{1□} = 1$, exactly one of *p*-1 constraints $□_{□}(□) \geq (□_{1□} + 1)$will hold as equality or would be active, and given the enough large value of *M*, the rest of constraints will be redundant. Thus, in the new non-dominated vector, at least value of one of the objective functions in vector $□^{1}$will be strictly larger, which guarantees variety of non-dominated vectors obtained. Since our goal is to maximize *m*-th objective function we believe that one different non-dominated vector will be obtained, therefore we obtain the non-dominated vector$□^{2}$, which among the feasible solution of Problem$\left(□_{□}^{1}\right)$, its *m*-th objective function value is the highest. Since the feasible region of Problem $\left(□_{□}^{1}\right)$ is the subset of the feasible region of Problem$\left(□_{□}^{0}\right)$,so the optimal value of Problem$\left(□_{□}^{1}\right)$ is smaller or equal to the optimal value of Problem$\left(□_{□}^{0}\right)$. Since ε is small enough, thus, we have$□_{2□} \leq □_{1□}$. If $\left(□_{□}^{1}\right)$ is infeasible, then, we conclude that there is only one non-dominated vector. Now suppose that *n*> 1. In this case also, the first and second class of constraints of Problem $(□_{□}^{□})$guarantee that the generated efficient vector is different from all non-dominated vectors belonging to set$□_{□}$. Thus the non-dominated vector$□^{n+1}$ will be obtained so that among the feasible solutions of Problem$(□_{□}^{□})$, its *m*-the objective function value is the highest. Since the feasible region of Problem($□_{□}^{□}$) is a subset of the feasible region of Problem($□_{□}^{□-1}$), thus the optimum of Problem($□_{□}^{□}$) is smaller or equal to the optimal value of Problem($□_{□}^{□-1}$) and as before, since ε is small enough thus we have $□_{(□+1)□} \leq □_{□□}$. If($□_{□}^{□}$) is infeasible, then, we conclude that the set of $□^{□} = \{□^{t}: 1 \leq □ \leq □\}$ includes all non-dominated vectors of the main Problem. This way, the proof of the proposition is completed.

The above results indicate that in order to generate all non-dominated vectors of the main Problem (*P*), it is enough to solve Problem ($□_{□}^{□}$)in an iterative manner until the stop condition, i.e. infeasibility of the Problem, is established.

The first modified algorithm

Step zero. Choose one of the objective functions like *m* that its value through the algorithm become maximized. Let *n* equal to zero. If *X* is empty, then there is no non-dominated vector. Stop.

Step one. Solve Problem($□_{□}^{□}$). If ($□_{□}^{□}$)infeasible, then go to the second step. Otherwise, name the non-dominated vector corresponding to the optimal solution of ($□_{□}^{□}$)as $□^{□+1}$. Increase *n* by one and repeat the first step from the beginning.

Second step. Stop. The set $□_{□} = \{□^{t}: 1 \leq □ \leq □\}$includes all the *n* non-dominated vector of Problem (*P*). The above modified algorithm is an improved version of the algorithm introduced in Section 2-1. In fact, this algorithm in *n*-th iteration reduces number of new binary variables from *np* to *n*(*p*-1), and the number of new constraints from *n*(*p*+1) to *np*.

2-2-2 The second modified algorithm

As we have seen, in Problem($□_{□}^{□}$), the number of binary variables added is $□(□ - 1)$ and number of added constraints is equal to □□. But in fact, for each feasible solution, at most one constraints is enough to identify the region that compared to existing points for each of the $□ - 1$ benchmark (objective function) is non-dominated. According to this, the second modified algorithm determines the necessary constraints and to find the solution of Problem($□_{□}^{□}$), several models with a maximum of $□ - 1$ constraints (for the lower bound) is added. Without reducing the generality of Problem and in order to simplify the symptoms, we let $□ = □$. Let $□^{(□^{□})} =$

$(\square_1^{(\square^\square)}, \ldots, \square_\square^{(\square^\square)})$ is the optimal non-dominated solution of the following corresponding problem:

$$(\square^\square): \quad \square\square\square \quad \square_\square(\square) + \square \sum_{\square=1}^{\square-1} \square_\square(\square)$$

s.t.

$$\square_\square(\square) \geq \square_\square, \qquad \square = 1,2, \ldots, \square - 1,$$

$$x \in X,$$

, where $\square = (\square_1, \square_2, \ldots, \square_{\square-1})$is the lower bound of each of the first *p*-1 introduced criteria. If the Problem$(\square^\square)$ is infeasible, then we have$\square^{(\square^\square)} = (-\square, -\square, \ldots, -\square)$.

First we will prove that Problem $(\square_\square^\square)$ for *p* = 3 can be divided into several sub models. Then we will generalize the results for any *p*.

Problem $(\square_3^\square)$ is divided into *n*+1 sub models, i.e. $(\square^{\square^{\square,\square}})$ for $\square = 0,1, \ldots, \square$ where, different bounds for the first and second criteria are set with the help of non-dominated points existed in$\square^\square = \{\square^t: 1 \leq \square \leq \square\}$. $\square^{k,n} = \left(b_1^{k,n}, b_2^{k,n}\right)$imply the bound vector corresponding to each of sub models$(\square^{\square^{\square,\square}})$ for $\square = 0,1, \ldots, \square$; *k*> 0, which indicates that we have used from the *k*-th non-dominated point, i.e. $\square^\square$, in order to define a lower bound for the first criteria, i.e.$\square_1^{\square,\square} = \square_{\square 1} + 1$. By defining this lower bound for the first criterion removes the region by current non-dominated points, value of their first criterion is less than $b_1^{k,n}$. When $b_1^{k,n}$is defined, it is enough by considering the current non-dominated points, which value of their first criterion is greater than or equal to$b_1^{k,n}$, we define a bound for the second criterion. In other words, if we define:

$$\square_\square^\square = \left\{\square^t: \square_{\square 1} \geq b_1^{k,n}, \qquad \square^t \in \square_\square\right\},$$

Then we define:

$$b_2^{k,n} = \max_{\square^t \in \square_\square^\square} \{\square_{\square 2}\} + 1.$$

If$\square_\square^\square = \square$, then there is no need to define a bound for the second criterion.

If $\square = 0$, we will not have a lower bound for the first criterion and $\square_\square^0$includes all the available non-dominated points, i.e. $\square_\square^0 = \square_\square$.

Proposition 3-8. Let $p = 3, □^{k,n} = (b_1^{k,n}, b_2^{k,n}), □_□^□ = \{□^t: □_{□1} \geq b_1^{k,n},\ □^t \in □_□\}$ for every $□ = 0,1,\dots,□$, and $□^*$ is such that:

$$□_3^{\left(□^{□^{k^*,n}}\right)} = \max_{k=0,1,\dots,n} □_3^{\left(□^{□^{k,n}}\right)}$$

where

$$b_1^{k,n} = \begin{cases} -M, & k = 0 \\ z_{k1} + 1, & k > 0' \end{cases} \qquad b_2^{k,n} = \begin{cases} -M, & □_□^□ = □ \\ \max_{□^t \in □_□^□}\{□_{□2}\} + 1, & o.w. \end{cases}$$

If $□_3^{\left(□^{□^{k^*,n}}\right)} = -□$, then $□_□ = \{□^t: 1 \leq □ \leq □\}$ includes all non-dominated points of the main Problem of (P). Otherwise, we have:

$$□^{□+1} = □^{\left(□^{□^{k^*,n}}\right)}.$$

Proof. Recall from the previous proposition that we can obtain $□^{n+1} = (z_{(n+1)1}, z_{(n+1)2}, \dots, z_{(n+1)p})$ by solving the Problem($□_□^□$). A rewrite of the Problem($□_□^□$) for the case with three criteria and m = 3 will be as follows:

$\left(□_3^□\right)$: $□□□\quad □_3(□) + □□_1(□) + □□_2(□)$

s.t.

$$□_1(□) \geq (□_{□1} + 1)□_□ - □(1 - □_□), \quad \forall □,$$

$$□_2(□) \geq (□_{□2} + 1)(1 - □_□) - □□_□, \quad \forall □,$$

$$□_□ \in \{0,1\}, \quad □ = 1,2,\dots,□,$$

$$X \in X$$

where $□_□ = 1$ indicates that $□_1(□) \geq □_{□1} + 1$ and $□_□ = 0$ result in $□_2(□) \geq □_{□2} + 1$.

Based on the value of k, one of the following modes for the optimum non-dominated point occurs:

First mode: $k = 0$. In this case, for every $□ = 1,\dots,□$, we have $□_□ = 0$. So, we have no additional lower bound for the value of the first criteria and therefore we can write $b_1^{k,n} = -M$. Also in this mode, for every $□ = 1,\dots,□$, we have $□_2(□) \geq □_{□2} + 1$ and thus, we can define:

$$b_2^{k,n} = \max_{□^t \in □_□^□}\{□_{□2}\} + 1$$

This mode is equivalent with the mode in which $k = 0$, which results in $□_□^0 = □_□$, because for every $t = 1, \dots, n$, we have $□_{□1} \geq b_1^{k,n}$.

Second mode: In this mode we have $0 < □ \leq □$. In this case, for some values of $k(1 \leq □ \leq □)$, we have $□_{□} = 1$ and for all t that satisfy the condition $□_{□1} \geq □_{□1} + 1$, we have $□_{□} = 0$. This shows that $□_1(□) \geq □_{□1} + 1$ and we can put $□_1^{□} = □_{□1} + 1$. For all t that satisfy the condition $□_{□1} \leq □_{□1}$, we have $□_{□1} + 1 \leq □_{□1} + 1 \leq □_1(□)$ and we can write $□_{□} = 1$.

Since just for all t that satisfy $□_{□1} \geq □_{□1} + 1$, we have $□_{□} = 0$, therefore, for every t that satisfy condition $□_{□1} \geq □_1^{□,□}$, we have $□_2(□) \geq □_{□2} + 1$, i.e. $□^{t} \in S_n^k$. If $S_n^k \neq □$, then, lower bound for $□_2$ will be as follows:

$$b_2^{k,n} = \max_{□^t \in □_{□}^{□}} \{□_{□2}\} + 1.$$

If $S_n^k = □$, then $b_2^{k,n} = -M$.

For every $□ = 0,1,\ldots,□$, value of $□^{(□^{□k,n})}$, is obtained by solving the corresponding models so that we consider all the possible states one by one for every new solution. Since our goal is maximizing the third criterion, sub model $□^*$ that has the optimal solution with the highest value for the third criterion, i.e.:

$$□_3^{(□^{□k^*,n})} = \max_{k=0,1,\ldots,n} □_3^{(□^{□k,n})},$$

Results in the next non-dominated point. $□_3^{(□^{□k^*,n})} = -□$ corresponds to the mode in which all sub models are infeasible and therefore Problem$(□_3^{□})$ is infeasible and $□_{□} = \{□^t : 1 \leq □ \leq □\}$ includes all the non-dominated points. Otherwise, the corresponding sub model will result in new non-dominated point $□^{□+1} = □^{(□^{□k^*,n})}$.

Similar to the case mentioned for $p = 3$, for each p, Problem$(□_{□}^{□})$ can be divided into sub-problems$(□^{□k,n})$ for $□ = (□_1, □_2, \ldots, □_{□-2})$. We used existing non-dominated points to define lower bound for criteria $□ = 1,2,\ldots,□-1$. Index of non-dominated point that we used to define bound for the j-th criteria was considered to be $□_{□}$. The relation $□_{□} = 0$ indicates that no specific bound is considered for the j-th criteria.

At first, when $0 < □_1 \leq □$, the lower bound is $b_1^{□,n} = z_{k_1 1} + 1$ and if when $□_1 = 0$, we consider the lower bound of $b_1^{□,n} = -□$ for the first criteria. Now, similar to the case with three criteria, we used only those non-dominated points that value of their first criterion is greater than or equal to$b_1^{\square,n}$to define bound for other criteria. In other words, the first criterion of$\square^{\square\square}$, which will be used to define bound for the second criterion, will satisfy the relation $z_{k_2 1} \geq z_{k_1 1} + 1$, provided that $\square_1, \square_2 \neq 0$.

We will continue finding bound for each of the criteria *j* = 1, ...,*p*-2, which depends on the value of$\square_\square$, so that$z_{k_j i} \geq z_{k_i i} + 1$ for each non-zero$\square_\square$and$\square_\square$with $\square < \square$.

In order to define a lower bound for *p*-1-th, we define the below set:

$$\square_\square^\square = \left\{\square^\square : \square_{\square\square} \geq b_j^{\square,n}, \qquad \square = 1,2,\dots,\square - 2, \qquad \square^\square \in \square_\square\right\}$$

that included the existing non-dominated points that yet dominated regions are not removed by them. This way, we impose the following conditions on the Problem:

$$\square_{\square-1}(\square) \geq \square_{\square(\square-1)} + 1 \qquad \forall \square^\square \in \square_\square^\square.$$

So we have:

$$b_{p-1}^{\square,n} = \max_{\square^t \in \square_\square^\square} \left\{\square_{\square(\square-1)}\right\} + 1.$$

According to the above explanations, the previous proposition can be extended for more than three criteria.

Proposition 3-9. Let *K* is the set of all possible combinations of $\square = (\square_1, \square_2, \dots, \square_{\square-2})$that

$$\square_\square \in \{0,1,2,\dots,\square\} \qquad \forall \square = 1,2,\dots,\square - 2,$$

$$z_{k_j i} \geq z_{k_i i} + 1 \qquad \forall \square_\square, \square_\square \neq 0, \qquad \square < \square.$$

Now considering

$$\square^{\square,n} = \left(\square_1^{\square,n}, \square_2^{\square,n}, \dots, \square_{p-1}^{\square,n}\right)$$

and

$$\square_\square^\square = \left\{\square^\square : \square_{\square\square} \geq b_j^{\square,n}, \qquad \square = 1,2,\dots,\square - 2, \qquad \square^\square \in \square_\square\right\}$$

, let $\square^*$ is such that

$$\square_\square^{\left(\square^{\square k^*,n}\right)} = \max_{\square \in \square} \square_\square^{\left(\square^{\square k,n}\right)},$$

where,

$$□_j^{□,n} = \begin{cases} -M, & □_j = 0 \\ z_{k_j j} + 1, & o.w. \end{cases} \qquad j = 1,2,\dots,p-2,$$

$$□_{p-1}^{□,n} = \begin{cases} -M, & □_□^□ = □ \\ \max_{□^t \in □_□^□} \{□_{□(□-1)}\} + 1, & o.w. \end{cases}$$

If $□_□^{(□^{□^*,n})} = -□$, then $□_□ = \{□^t: 1 \le □ \le □\}$ includes all the non-dominated points of the main Problem (*P*). Otherwise, we have:

$$□^{□+1} = □^{(□^{□^*,n})}.$$

The second modified algorithm

Step zero. Set *n* equal to zero. If$(□_□^0)$is infeasible, then *X* is empty and there is no non-dominated vector. Stop.

Step one. Find $□_□^{(□^{□,n})}$for each *k*□*K*. Specify $□^*$ such that:

$$□_□^{(□^{□^*,n})} = \max_{□ \in □} □_□^{(□^{□,n})}.$$

If $□_□^{(□^{□^*,n})} = -□$ (all models are infeasible), go to the second step. Otherwise, new non-dominated point will be as follows:

$$□^{□+1} = □^{(□^{k^*,n})}.$$

Increase *n* by one and repeat the first step from the beginning.

Second step. Stop. Set$□_□ = \{□^t: 1 \le □ \le □\}$includes all *n* non-dominated vector for Problem (*P*).

If *N* is the number of all the non-dominated points of Problem (*P*), then at worst mode, number of models that must be solved to find *N* is equal to:

$$\sum_{□=0}^{□} \sum_{□ \in □} 1,$$

which of the order of $O(□^{□-1})$.

The number of models that must be solved can be reduced by maintaining some information in memory. Many of models results in the same solutions, since there are only *N* non-dominated point, while we have solved more than *N* model, each of which results in one non-dominated point. As is shown in the following proposition, by keeping vector of lower bounds *b* and the corresponding solution to it, i.e.$□^{(□^□)}$**,** we can detect whether the generated optimal solution is similar to each of the previous solutions or not.

Proposition 3-10. Let the bound vectors $□_1 = (b_1^1, b_2^1, \dots, b_{p-1}^1)$ and$□_2 = (b_1^2, b_2^2, \dots, b_{p-1}^2)$. If

$$□_□^1 \leq □_□^2 \leq □_□^{(□^□1)}, \qquad \forall □ = 1,2, \dots, □ - 1,$$

Then$□^{(□^□□)} = □^{(□^□□)}$**.**

Proof. Since $□_□^2 \leq □_□^{(□^□1)}$ for every $□ = 1,2, \dots, □ - 1$, thus, the non-dominated point$□^{(□^□□)}$ for Problem ($□^□□$) is also feasible. Assume ($□^□□$)has an optimal solution so that$□^{(□^□□)} \neq □^{(□^□□)}$. Since$□^{(□^□□)}$is not an optimal solution to Problem($□^□□$)**,** and both Problems seek to maximize the *p*-ti criteria, so we have$□_□^{(□^□1)} < □_□^{(□^□2)}$. In addition, to meet the feasibility condition, we can write:

$$□_□^2 \leq □_□^{(□^□2)}, \quad \forall □ = 1,2, \dots, □ - 1.$$

Since we know $□_□^1 \leq □_□^2$for each $□ = 1,2, \dots, □ - 1$, therefore, for each $□ = 1,2, \dots, □ - 1$, we will have $□_□^1 \leq □_□^{(□^□2)}$and thus $□^{(□^□□)}$is also infeasible for Problem($□^□□$). But we know $□_□^{(□^□1)} < □_□^{(□^□2)}$and therefore $□^{(□^□□)}$results in contradiction with optimality$□^{(□^□□)}$ for Problem($□^□□$). This contradiction shows that$□^{(□^□□)} = □^{(□^□□)}$**.**

In a similar way, we can maintain and use the lower bounds that have caused infeasibility. The following result makes possible to detect infeasibility of the problem in the next iterations, without solving the model.

Result. Consider bounds$□_1 = (b_1^1, b_2^1, \dots, b_{p-1}^1)$ and$□_2 = (b_1^2, b_2^2, \dots, b_{p-1}^2)$. If ($□^□□$) is infeasible and

$$□_□^1 \leq □_□^2, \qquad \forall □ = 1,2, \dots, □ - 1,$$

Then ($□^□□$) will also be infeasible.

REFERENCES

Corley, H.W.: Efficient spanning trees. J. Optim. Theory Appl. **45**, 481–485 (1985).

Gomes da Silva, C., Clímaco, J., Figueira, J.R.: Core problems in bi-criteria 0–1 knapsack problems. Comput. Oper. Res. **35**(7), 2292–2306 (2008).

Guerriero, F., Musmanno, R.:Labelcorrectingmethodstosolvemulticriteriashortestpathproblems.J.Optim. Theory Appl. **111**(3), 589–613 (2001).

Köksalan, M.M.: A heuristic approach to bicriteria scheduling. Nav. Res. Logist. **46**, 777–789 (1999).

Martins, E.Q.V.: On a multicriteria shortest path problem. Eur. J. Oper. Res. **16**, 236–245 (1984).

Prim, R.C.: Shortest connection networks and some generations. Bell Syst. Tech. J. **36**, 1389–1401 (1957).

Przybylski, A., Gandibleux, X., Ehrgott, M.: A two phase method for multi-objective integer programminganditsapplicationtotheassignmentproblemwiththreeobjectives.Discret.Optim. **7**(3), 149–165(2010).

Ramos, R.M., Alonso, S., Sicilia, J., Gonzáles, C.: The problem of the optimal biobjective spanning tree. Eur. J. Oper. Res. **111**, 617–628 (1998).

Steiner, S., Radzik, T.: Computing all efficient solutions of the biobjective minimum spanning tree problem. Comput. Oper. Res. **35**(1), 198–211 (2008).

How to cite this article:
Gheidari A S. Finding non-dominated vectors in multi-objective linear programming problems. J. Fundam. Appl. Sci., 2018, *10(2S), 783-802.*