Multi-criteria Integer Programming Problems

Thesis submitted in partial fulfillment of the requirement for

The award of the degree of

Masters of Science

In

Mathematics and Computing

Submitted By

Twinkle Goyal
301603025

Under the guidance of

Dr. Vikas Sharma

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

JULY 2018

School of Mathematics

Thapar Institute of Engineering and Technology,

Patiala-147004

PUNJAB, INDIA

Dedicated to God, My Parents and My Supervisor

Declaration

I hereby certify that the work, which is being presented in the thesis, entitled **"Multi-criteria Integer Programming Problems"** in partial fulfillment of the requirements for the award of the degree of **Masters of Science** in **Mathematics and Computing** and submitted to the School of Mathematics, Thapar Institute of Engineering and Technology, Patiala, is an authentic record of my own work carried out under the supervision of **Dr.Vikas Sharma,** Assistant Professor and other research work which is duly listed in the reference section.

The matter presented in this thesis has not been submitted elsewhere for the award of any other degree or diploma from any institution.

Date: 2-07-2018

Twinkle Goyal

Twinkle Goyal

Roll No. 301603025

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.

Date: 02-07-2018

Dr. Vikas Sharma

Assistant Professor

TIET, Patiala

Dr. Satish Kumar

Professor & Head

TIET, Patiala

Acknowledgment

Foremost, I would like to express my deep gratitude to my supervisor **Dr. Vikas Sharma** for his indispensable advice and encouragement at each and every step. Without his steady support and belief in me, I would not be able to complete this thesis. He helped me a lot during this project, having so much responsibility on his own yet he managed to give me required time for my work. He also provided me with valuable advices. It was a great honor to have mentor like him.

I take this opportunity to express my sincere thanks to **Dr. Satish Kumar, Head, SOM, TIET, Patiala,** for his valuable support.

I express my sincere and deep gratitude to my parents and family member for their constant belief in me and helping me to stand again when I am losing my hope.

Finally, I am also thankful to my friends, Rajenki, Akshay who always encouraged me to do something good, Lokesh, who devoted his valuable time and helped me with his skills and Aman, who always heard me when things went wrong and advised me with the best.

Twinkle Goyal

Abstract

Multi-criteria optimization is concerned with mathematical optimization problems having more than one objective function to be optimized simultaneously. In this thesis we have reviewed two different papers based on multi-criteria optimization.

In the second chapter we have studied bicriteria integer linear programming problem, where we aim to generate all the efficient solutions of a multi-criteria optimization problem. An algorithm discussed in this chapter makes use of a scalarization technique, where the bicriteria optimization problem is converted into a single criteria optimization by treating one of the objective function as constraint.

In the third chapter we have obtained optimal integer solution of linear fractional problem over the efficient set of multi-objective integer linear problem. An iterative algorithm is discussed which by making use of efficiency test at each stage an optimal solution of a constrained integer programming problem is obtained, finds an integer efficient solution of the given problem.

Contents

Chapter 1

Introduction

1.1 Introduction

Optimization can be stated as the activity of making the best or best utilization of a circumstance or asset. In this, we maximize or minimize a particular objective function with respect to some constraints or a set representing the available restrictions or choices for the specific problem. Hence, using this model we can find the best required solution for a particular problem. In real life it is used to minimize the cost, maximize the gain, minimize the error, minimize the distance, maximize the productivity with respect to supply etc. Optimization is used in various fields such as in studying tumor growth, production of certain enzyme, maximizing gain or minimizing loss etc. An optimization problem can be stated as

$$\text{Min } f(x)$$

subject to

$$g_i(x) \leq = \geq 0 \quad (i = 1, 2, ..., m)$$

where $x \in \mathbb{R}^n$ is the vector of unknown decision variables.

If all the functions f and g_i, $(i = 1, 2, ..., m)$ are linear functions of the decision variable x, then it is called a *linear optimization problem* or a *linear programming problem (LPP)*. However, if either the objective function f or at least one of the constraints g_i $(i = 1, 2, ..., m)$ is nonlinear function of the decision variable x, then it is called a *non-*

linear optimization problem or a *non-linear programming problems (NLP)*. Certain linear programming problems require that some or all the variables take only integer values, we call such problems as *integer linear programming problems* (ILP's). Integer LPP's are very common in real life applications, e.g. in a production problem, the items being produced may be in complete units (say items are T.V. sets of 21" and 29") and therefore, fractional number of items may not have any meaning. The concept of linear and non-linear programming problems involving optimization of a single objective function is has been widely studied in literature. In the present study we have discussed multi objective integer programming problems.

Multi-criteria and Single-criteria Optimization

When an optimization modeling consist of only one objective function, the task of finding the optimal solution is called *single objective optimization*, whereas if optimization problem involves more than one objective function then it is referred as *multi-objective optimization*. Since multi-objective optimization consists of multiple objectives therefore it can be seen that single-objective is a degenerate case of multi-objective optimization.

The single-objective optimization problems are relatively easier to tackle than the multi objective optimization problems hence they received more exposure as compared to multi-objective optimization techniques. But in real life we deal with problems having more than one objective functions. For example: Consider a decision making problem of buying a new cellphone. There are a lot of features like cost, battery backup, camera quality, RAM, internal capacity, processor version etc. A cheap cell phone may not have a good camera or battery. Therefore it can be seen that a decision of buying a particular cellphone is not only based on single criteria, of say cost alone, but many more criteria are equally vital in the final decision.

Formulation of Multi-objective optimization:

Multi-objective optimization can be formulated as:

$$\text{optimize } z = (f_1(x), f_2(x), ..., f_k(x))$$

$$\text{such that}$$

$$Ax \leq = \geq b;$$

$$x \geq 0$$

where z is the objective function, $(f_1(x), f_2(x), ..., f_k(x))$ are k no. of distinct objective functions. A is $m \times n$ matrix; $b, x \in \mathbb{R}^m$.

The solution to a multi-objective problem results in a number of points in the objective space referred to as Pareto optimal solutions. For a multi-objective problem with two objective functions (the first function is efficiency maximization and the second function is cost minimization). The Pareto optimal front is obtained using the principal of *domination*. In this concept, each solution is compared to check whether it dominates another solution or not.

A solution z^1 is said to dominate another solution z^2 if the following conditions are satisfied

1) The solution z^1 is no worse than z^2 in all objectives.

2) The solution z^1 is better than z^2 in at least one objective.

The points along the Pareto optimal front are referred to as non dominated solutions. This front can be concave, partially convex/concave or discontinuous. The trade-off between the objective functions defines the shape of the Pareto front.

Now, there are various methods for obtaining the non dominated solutions for a multi objective optimization problem. The weighted sum approach, ϵ - constraint method, goal programming, and utility function method are the common techniques for solving multi-objective problems.

In the weighted sum approach, different objectives are combined into a single objective function using different weights. This method is simple and easy to implement. However, it can locate one Pareto point in one optimization run using the gradient- based method.

In the ϵ-constraint method , one objective function is minimized and the remaining objective functions are transformed into constraints which are to be specified by the user. The transformed problem is then solved using the gradient-based method .The method can locate the Pareto fronts of the non convex problems .

In goal programming approach, a target is set for each of the objective functions and the optimizer aims to minimize the deviations from these set gaols.

In the utility function method, all the objectives are combined onto a single function which is then solved along with the constraints .

Linear Fractional Programming Problems

In mathematical optimization, linear-fractional programming (LFP) is a generalization of linear programming (LP). Whereas the objective function in a linear program is a linear function, the objective function in a linear-fractional program is a ratio of two linear functions .A linear program can be regarded as a special case of a linear-fractional program in which the denominator is the constant function.

The feasible region for both the linear programming and linear fractional programming is taken as polyhedron. Linear programming computes a policy delivering the best outcome, such as maximum profit or lowest cost. In contrast, a linear-fractional programming is used to achieve the highest ratio of outcome to cost, the ratio representing the highest efficiency.

A LFP can be stated as:

"Problems in which the objective function is the ratio of two affine functions and constraints are affine inequalities are called Linear Fractional Programming Problems."

$$\text{(LFP)} \qquad \text{Max } Z = \frac{c^T x + \alpha}{d^T x + \beta}$$

subject to

$$Ax = b, \qquad x \geq 0.$$

The denominator of the objective function is strictly positive for all feasible solutions. The objective function in a linear-fractional problem is both quasiconcave and quasiconvex (hence quasilinear) with a monotone property, pseudoconvexity, which is a stronger property than quasiconvexity. A linear-fractional objective function is both pseudoconvex and pseudoconcave, hence pseudolinear. We have different kinds of methods for solving LFP. The first method is Simplex Method and the other method is Charnes and Cooper, this describe another procedure of solving LFP by reducing the problem in to a Linear Problem by using non-linear variable transformation.

1.2 Literature Review

In MOLP problems, all objectives are linear and must be optimized over a convex polyhedron. MOLP problems are solved as sub problems for MOIP and MOMIP and all non-dominated solutions of MOLP are supported. MOLP problems are popular and there is a lot of literature that covers finding the efficient set, some of which are covered in this section. Benson (1998a) generates an outer approximation algorithm in the outcome space to mitigate the size of a problem. The main advantage of this algorithm is that there is no need for backtracking or bookkeeping which is needed when solving in the decision space, as in Benson (1997). This method is later implemented into Benson (1998b) which introduces a hybrid vector maximization approach which was first introduced by Kuhn and Tucker (1951). Benson incorporates the special simplified partitioning technique used by Ban (1983) and Tuy and Horst (1988) into the outcome space using outer approximation. Later in Benson (1998c) it was found that this algorithm also generated weakly efficient points in the outcome space. Benson and Sun (2000) proved that a feasible basis for the linear program LP can be decomposed into a finite union of subsets with a one-to-one correspondence between the weights and efficient extreme solutions in the outcome space. Using this result, Benson and Sun (2002) develop a weight set decomposition algorithm. Yamamoto((2002) studied optimization over the efficient set. The existing algorithms for solving linear multi-criteria problems could be classified into several groups such as adjacent vertex search algorithms, non-adjacent vertex search algorithms, branch and bound based algorithms, lagrangian relaxation based algorithms, dual approach and bisection algorithm. He compared algorithm from each group from computational point of view. Ida (2005) uses an extreme ray generation method to sequentially generate efficient points and rays. In the algorithm, objective values for each extreme ray are obtained and tested for efficiency. Krichen, Masri and Guitouni (2012) generate maximal efficient faces using adjacency between efficient extreme points. This algorithm explores efficient extreme points and uses simplex pivots to find adjacent vertices of the current extreme point.

The main difference between MOLP and MOIP problems is that MOIP objectives are discrete, not continuous. In bi-criteria problems, it is well known that sub problems must be solved to generate all efficient solutions. Klein and Hannan (1982) propose a sequential generation method for finding all non-dominated solutions in the decision space. This method solves a sequence of progressively more constrained single-objective integer problems. At each step a new constraint is added which excludes previously generated efficient points. This allows points which are dominated by the generated non-dominated solutions to be eliminated. A variation of this method is used by Sylva and Crema (2004) which sequentially solves weighted sum problems instead of single-objective problems and later, Sylva and Crema (2007) propose another variant that finds a well-dispersed subset of non-dominated points. An improvement of the algorithm by Sylva and Crema (2004) is developed by Lokman and Koksalan (2012) which decreases the number of additional constraints to be added at each step. Lemesre, Dhaenens and Talbi (2007) propose parallel partitioning method (PPM) to solve bi-objective programming problems. This method uses three stages to determine the entire Pareto front. Firstly, the problem is solved for extreme solutions to limit the search space. Next, the search space is divided up by searching the efficient solutions. Lastly, the solutions found from the previous stage are used to find any other efficient solutions. An extension of this method for any number of objectives is done by Dhaenens, Lemesre and Talbi (2010).

MOMIP problems are the hybrid of MOLP and MOIP problems. There are several types of problems within MOMIP itself due to the combination of continuous and integer variables. So far, there is no existing algorithm that can solve for mixed 0-1 integer programs p¿3 with objectives and no general algorithm to find all non-dominated solutions. Mavrotas and Diaokoulaki (1998) modify the single-objective branch and bound algorithm to find efficient solutions in mixed 0-1 MOLP problems in the decision space. Later, Mavrotas and Diaokoulaki (2005) further extend to find the efficient solutions of this problem using a vector maximization approach of the branch and bound method. Jozefowiez, Laporte and Semet (2012) propose a general multi-objective branch and bound method which does not iteratively solve single-objective problems. The lower and upper bounds are defined as sets of points in the objective spaces instead of being single values. Stidsen, Andersen and Dammann (2014) use branch and bound to find all non-dominated solutions for bi-objective mixed integer problem where all integers must be binary and only one of the objectives may be a continuous. This algorithm first solves the problem with all binary values as free variables. Przybylski, Gandibleux and Ehrgott (2010b) develop some additional properties for the weight space for MOMIP and develop their algorithm based on this. The algorithm utilizes the bi-objective algorithms of Cohon (1978) and Aneja and Nair (1979) and recursively reduces multi-objective problems into bi-objective problems which can then solved by the bi-objective algorithms. Zerdani and Moulai(2011) discussed about optimization over an efficient set of a multiple objective linear fractional problem. He proposed an algorithm which optimizes an arbitrary linear function over an integer efficient set of problem(MOLFP) without explicitly having to enumerate all the efficient points. The presented algorithm is based on simple selection technique that improves the linear objective value at each iteration.

1.3 Outlines of Present Work

In this thesis, we reviewed two papers on multi-objective integer problems.

In chapter 2, we have discussed bi-criteria integer programs. In this chapter we find all the efficient points of multi-objective problem. Here, we consider two different objectives and find all the corresponding efficient points. Basically, in this chapter we proposed an algorithm which generates all the efficient points by minimizing the first objective function and by taking the second objective function as a restriction over the first. Mathematically the problem can be stated as follows:

Let $c, d, x \in \mathbb{N}^n$ with $c = (c_1, c_2, ..., c_n)$, $d = (d_1, d_2, ...d_n)$, $x = (x_1, x_2, ..., x_n)$

and $b \in \mathbb{N}^m$ with $b = (b_1, b_2, ..., b_m)$ and $A \in \mathbb{N}^{n \times m}$

Then we minimize two linear objective functions $z_1(x_1, x_2, ..., x_n)$ and $z_2(x_1, x_2, ..., x_n)$ with m linear constraints.

$$\begin{aligned} \text{minimize} \qquad & z_1 = \sum_{i=1}^{n} c_i x_i \\ & z_2 = \sum_{i=1}^{n} d_i x_i \\ & Ax \leq=\geq b \end{aligned}$$

Let X be the set of feasible solutions then $X \subseteq \mathbb{N}^n$ and if there exist x which satisfies all the linear restrictions then $x \in X$.

Also, z_1 and z_2 is the optimal value corresponding to x. Thus, corresponding to value of x we have a pair (z_1, z_2) called feasible objective function vector.

It may be noted here,that only positive objective function of pair (z_1, z_2) is considered because all $x_i's \geq 0$ and therefore $z_i's \geq 0$ and also each element of pair represents an element of N^2. In our problem N^2 is objective or criteria space.

The algorithm discussed in this chapter is the first algorithm proposed for generating all the efficient points for this particular kind of problem given by P.Neumayer and D.Schweigert[20].

In chapter 3, we have discussed the method to find the optimal solutions of integer linear fractional problem over the efficient set of multi-objective integer linear problem. This method uses Branch and Bound technique and the continuous linear fractional programming program. The Branch and Bound method is strengthened by adding efficient cuts and efficiency tests to avoid large number of non-efficient solutions of multi-objective optimization. In general, the efficient solution set have large cardinality, made difficult for the decision maker to choose the best compromise. Though, the decision maker have its own predictions using one function other than those of the multi-objective integer linear problem.

We considered a MOILP defined as:

MOILP $\quad \max f_i(x) = \sum a^i x^i; \quad i \in \{1, 2, ..., s\}$

s.t.

$$x \in E = X \cap \mathbb{Z}^n$$

where the set of integers is denoted by Z and $s \geq 2$; $A \in \mathbb{R}^{m \times n}$; $b \in \mathbb{R}^n$ and $X = \{x \in R^n \mid Ax \leq b, x \geq 0\}$ and $X \neq \phi$ and X is convex- polyhedron set also $E \neq \phi$.

Now, a LFP is maximized by an efficient solution of MOLIP is defined as

(ILPE) $\quad \max \dfrac{cx + d}{px + q}$

s.t.

$$x \in X_E \subset E$$

where c, d are real n vectors, $\alpha, \beta \in \mathbb{R}$ be any real constants and X_E is efficient set of multi-objective integer linear problem. The denominator of the objective function of integer linear fractional problem is always strictly positive for all $x \in X$.

Now we obtain an optimal solution after solving ILPE over efficient solution set of MOLIP.

Two cases arise i.e.

1) If the solution is integer, then we check its efficiency by using Ecker snd Kouada(19750) efficiency test.

2) If the solution is non-integer, then we first apply Branch and Bound technique and obtain a integer solution, then we apply efficiency test to obtain integer efficient solution. Hence, in this chapter we presented an algorithm to find an optimal solution by maximizing a linear fractional problem over the non-convex set of efficient solutions of the multi-objective integer linear problem.

Chapter 2

Bi-criteria integer linear programs

The integer linear programming is consider only a single criteria for optimizing the total amount. But in real life we require two or more objectives. For example: Consider the case where we have to maximize the total cost and minimize the total consumption then in those situations we deal with more than one objective functions termed as multi-objective problems. But in some situations we require only integers solutions for example as in distribution of products, scheduling of production or quenching of machines and we generally require the optimized value of a function which is the combination of all these functions. Hence, multi-objective optimization combined with integer optimization is of great use.
Now, in this chapter the work of P.Nevmayer, D.Schweigert[20], has been reviewed, instead of using bi-criteria transportation optimization problem we have considered bicreteria integer optimization problem.

2.1 Formulation of bi-criteria integer linear program

Let $a, b, y \in \mathbb{N}^n$ with $a = (a_1, a_2, ..., a_n)$, $b = (b_1, b_2, ...b_n)$, $y = (y_1, y_2, ..., y_n)$
and $c \in \mathbb{N}^m$ with $c = (c_1, c_2, ..., c_m)$ and $A \in \mathbb{N}^{n \times m}$.
Now, we have to minimize both the linear objective functions $f_1(y_1, y_2, ..., y_n)$ and $f_2(y_1, y_2, ..., y_n)$ with m linear constraints.

$$\text{minimize } f_1 = \sum_{i=1}^{n} a_i y_i$$
$$f_2 = \sum_{i=1}^{n} b_i x_i$$
$$Ay \leq = \geq c$$

Let Y denotes be the set of feasible solutions then $Y \subseteq \mathbb{N}^n$ and if there exist y which satisfies all the linear restrictions then $y \in Y$.

Also, f_1 and f_2 is the optimal value corresponding to y. Thus, corresponding to value of y we have a pair (f_1, f_2) called feasible objective function vector.

It may be noted here,that only positive objective function of pair (f_1, f_2) is considered because all $y_i's \geq 0$ and therefore $f_i's \geq 0$ and also each element of pair represents an element of $\mathbb{N}^2$. In our problem $\mathbb{N}^2$ is objective or criteria space.

2.2 Definition and Assumptions

Consider a set F of all the feasible points in objective space $\mathbb{N}^2$ such that $F \subseteq \mathbb{N}^2$. For every feasible point $f \in F$ there is a feasible point $f = (f_1(x), f_2(x))$ in the objective space.

Dominated points:: Let $F \subseteq \mathbb{N}^2$ be a set of all the feasible points in an objective space. Let $f = (f_1, f_2) \in F$, $f' = (f'_1, f'_2) \in F$ the f dominated f' that is $f < f'$ if $f_1 \leq f'_1,\ f_2 \leq f'_2$ and $f \neq f'$. For example: Consider a set $F = \{(1,5),(5,1),(6,2)\}$ here $(6,2)$ is dominated point as $(5,1) < (6,2)$.

Efficient or non-dominated points: Let $F \subseteq \mathbb{N}^2$ be the set of all feasible points in an objective space. Then $f' \in F$ is called efficient or non-dominated if there is no $f \in F$ with $f < f'$ and $f \neq f'$. The set of all these efficient points is called E. For example: Consider a set $F = \{(1,5),(5,1),(6,2)\}$ here $(1,5),(5,1)$ are the efficient points and $E = \{(1,5),(5,1)\}$.

Extreme efficient points: Let $F \subseteq \mathbb{N}^2$ be a set of all the feasible points in an objective space. Then an efficient point $f' \in F$ is called extreme efficient if $f'_1 = \min\{f_1 \mid f_1 = \sum_{i=1} a_i y_i \textit{ and } y \in Y\}$ or $f'_2 = \min\{f_2 \mid f_2 = \sum_{i=1} b_i y_i \textit{ and } y \in Y\}$. For example: Consider a set $F = \{(1,5),(5,1),(3,3)\}$ here $(1,5),(5,1)(3,3)$ are the efficient points and $E = \{(1,5),(5,1),(3,3)\}$ and the points $(1,5),(5,1)$ are extreme efficient.

Ideal points: Let $F \subseteq \mathbb{N}^2$ be a set of all the feasible points in an objective space. Then $f' \in F$ is called an ideal point if $f'_1 = \min\{f_1 \mid f_1 = \sum_{i=1} a_i y_i \textit{ and } y \in Y\}$ and $f'_2 = \min\{f_2 \mid f_2 = \sum_{i=1} b_i y_i \textit{ and } y \in Y\}$.

Lexicographic Order: Let $f = (f_1, f_2) \in F,\ f' = (f'_1, f'_2) \in F$ we say f is lexicographically smaller than f', if either $f_1 < f'_1$ or $f_1 = f'_1 \textit{ and } f_2 < f'_2$.

Adjacent elements: Let $f', f'' \subset E$ with $f'_1 < f''_1$. Then if a set $S = \{f \mid f \in E \textit{ and } f'_1 < f_1 < f''_1\}$ is empty then elements f', f'' are called adjacent elements.

In the above definitions we considered non-dominance concept over two objective functions but in general it can be extended to n objective functions.

Now, a bi-criteria optimization problem can be described by a set of efficient solutions corresponding to the feasible solution set i.e. to solve the described problem we have to evaluate the set E of all the efficient points.

When the variables are continuous i.e. $y_i \in \mathbb{R}$, then there are well known methods to solve. In those cases feasible set F in an objective space is a convex polytope and each efficient point belongs to a convex hull of the feasible points in an objective space. Therefore, for finding all the efficient points a multi-criteria simplex algorithm (Zeleny) can be used.

Now consider the case when variables are discrete i.e $y_i \in \mathbb{N}$ then in this case efficient points can lie inside the convex hull and these efficient points which lie inside the convex hull can not be found by multi criteria simplex algorithm.

2.3 Example

Consider a bicriteria linear integer problem

Min $F = (f_1, f_2)$ where $f_1 = y_1 - 2y_2$ and $f_2 = -3y_1 + 2y_2$

s.t. $\quad -y_1 + y_2 \leq 2$

$\quad y_1 + y_2 \leq 4$

$\quad y_1, y_2 \geq 0$

This problem has four basic feasible solutions $(4,0),(1,3),(0,2),(0,0)$.

At point $(4, 0), y_1 = 4, y_2 = -12$

At point $(1, 3), y_1 = -5, y_2 = 3$

At point $(0, 2), y_1 = -4, y_2 = 4$

At point $(0, 0), y_1 = 0, y_2 = 0$

For each of the feasible solution we have a pair (y_1, y_2) of values of an objective function called the feasible objective function or point or point vector $f = f(y_1, y_2, ..., y_n)$ i.e.

At point $(4, 0), f = (4, -12)$

At point $(1, 3), f = (-5, 3)$

At point $(0, 2), f = (-4, 4)$

At point $(0, 0), f = (0, 0)$

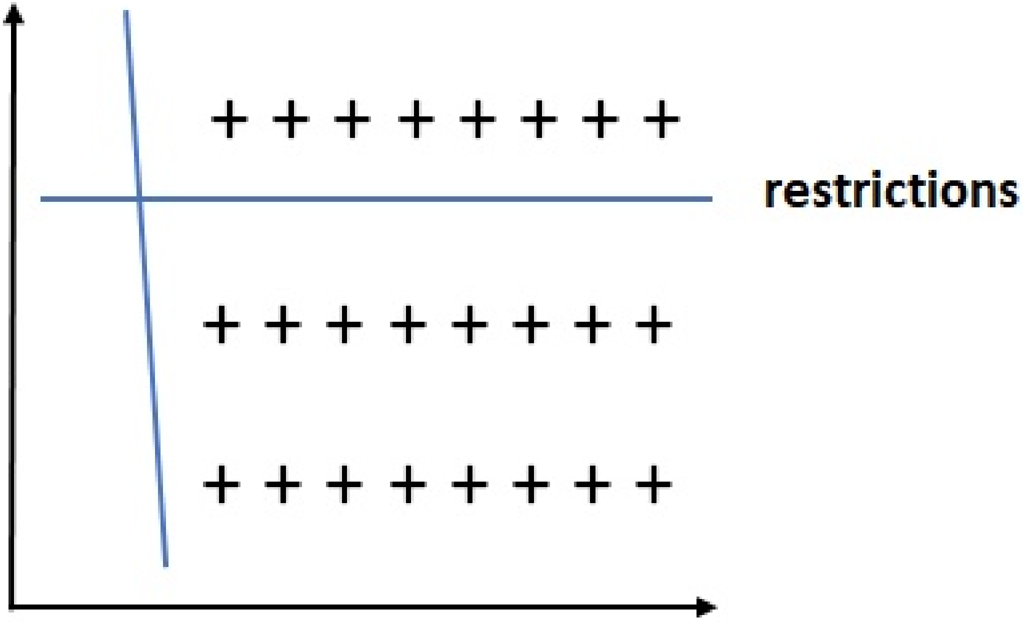
The points $(4, -12), (-5, 3), (-4, 4), (0, 0)$ are in objective space and $(4, -12), (-5, 3)$ are only extreme efficient points.

Now, we require to find all the efficient points and this can be done by using the following algorithm.

2.4 Reduction Algorithm

Consider f_1 and f_2 two linear objective function. According to this algorithm f_1 is of lexicographic order on F, and we take f_2 as additional restriction which is flexible in nature. We can eliminate all those efficient points which are already found, hence reducing the feasible space. Every time we found an efficient point this additional restriction is modified and this step is repeated until we get all the efficient points.

Firstly, we calculate $b' = max\{f_2 \mid f \in F\}$ and $a' = min\{f_2 \mid f \in F\}$. A new objective function $\psi(f_1, f_2) = f_2 + (b' - a' + 1)f_1$ can be defined. The function $\psi(f_1, f_2)$ is of lexicographic order on F and it is used in the algorithm. The basic idea of algorithm is shown in following diagram.



For efficient points we start our research from the left to the right and from bottom to top. The restriction is added and it is allowed to move from the top to the down and algorithm will stop when the restriction is reached on the lowest level of f_2 we can not find efficient point further.

Now we will show some results. The first result proved that f is of lexicographic order on F. The second result proved that the lexicographic ordering is a completion of the ordering given by concept of efficiency.

2.4.1 Some Propositions

1) Consider $f, f' \in f$and $\psi(f) = f_2 + (b' - a' + 1) f_1$ with $b' = max\{f_2 \mid f_2 = \sum_{i=1} b_i y_i \ and \ y \in Y\}$. Then f is lexicographically smaller than f' iff $\psi(f) < \psi(f')$.

Proof: Let $f, f' \in F$ and $\psi(f) = f_2 + (b' - a' + 1) f_1$ with $b' = max\{f_2 \mid f_2 = \sum_{i=1} b_i y_i \quad and \quad y \in Y\}$ and $a' = min\{f_2 \mid f_2 = \sum_{i=1} b_i y_i \quad y \in Y\}$.

First we consider that f is lexicographically smaller than f'. Thus we can either have $f_1 = f'_1$ and $(f'_2 - f_2) > 0$ or $f_1 < f'_1$ and $|f_2 - f'_2| < (b' - a' + 1)$.

Case1: when $f_1 = f'_1$ and $(f'_2 - f_2) > 0$

$$\begin{aligned}\psi(f') - \psi(f) &= f'_2 + (b' - a' + 1)f'_1 - f_2 - (b' - a' + 1)f_1 \\ &= f'_2 - f_2 + (b' - a' + 1)(f'_1 - f_1) \\ &= f'_2 - f_2 > 0 \text{ and therefore } \psi(f') > \psi(f) \text{ is true.}\end{aligned}$$

Case2: when $f_1 < f'_1$ and $|f_2 - f'_2| < (b' - a' + 1)$

$$\begin{aligned}\psi(f') - \psi(f) &= f'_2 + (b' - a' + 1)f'_1 - f_2 - (b' - a' + 1)f_1 \\ &= (f'_2 - f_2) + (b' - a' + 1)(f'_1 - f_1) \\ &> -(b' - a' + 1) + (b' - a' + 1) = 0\end{aligned}$$

and hence again $\psi(f') > \psi(f)$ holds. Now we consider that $\psi(f) < \psi(f')$ holds. Then we have $\psi(f') - \psi(f) = f''_2 - f_2 + (b' - a' + 1)(f'_1 - f_1) > 0$.The estimations of a',b' are with the end goal that $(b' - a' + 1)|f'_2 - f_1| > |f'_2 - f_2|$ is true for $f \in F$. In this manner we have $f'_1 - f_1 \geq 0$. There are currently two cases. In the event that $f'_1 - f_1 > 0$ then f is lexicographically smaller. But if $f'_1 - f_1 = 0$ then $f'_1 = f_1$ and $f'_2 - f_2 > 0$. In this way again f is lexicographically smaller than f'.

Corollary: Let $p, q \in F$ be extreme efficient with $p_1 < q_1$ and $\psi(f) = f_2 + (b' - a' + 1)f_1$. At that point $\psi(p) = min\{\psi(f)|f \in F\}$ holds.

2) Let $f, f' \in F$. If f dominates f', then $\psi(f) < \psi(f')$ holds and f is Lexicographically smaller than f'.

Proof: Due to $f < f'$ and $(b' - a' + 1) > 0$ we have $(f_2 - f'_2) + (b' - a' + l)(f_l - f'_1) < 0$. It takes after $f_2 + (b' - a' + 1)f_1 < f'_2 + (b' - a' + 1)f'_1$ and in this manner $\psi(f) < \psi(f')$. By (1) we at long last understand that f is lexicographically smaller than f'.

Remark: If we take two efficient objective function vectors with the segments p_2, q_2 rather than a', b' all together to find $f(z)$ the results (1) and (2) are as it were local explanations. Local implies that results (1) and (2) are valid for all f between the used efficient objective function vectors p and q i.e. for all $f \in S$ with $S = \{f \in F \mid p_1 \leq f_1 \leq q_1 \;\; and \;\; q_2 \leq f_2 \leq p_2\}$. If $S = \{f \in F \mid p_1 \leq f_1 \leq q_1 \;\&\; q_2 \leq f_2 \leq p_2\}$ we can say S is formed by p and q.

3) Let $S \subseteq F$ be generated by $p, q \in F$, $a' \in (q_2, p_2)$ and $S' = \{f | f \in S \;\; and \;\; s_2 < a'\}$. If $\psi(f') = min\{\psi(f)|f \in S'\}$ then s' is efficient in S' and if S is generated by the two efficient points $p, q \in E$ then even $f' \in E$ holds.

Proof: Assume f' is not an efficient point, at that point there is a $f \in S$ with the condition that f' is dominated over f. By (2) we have $\psi(f) < \psi(f')$ in logical inconsistency to $\psi(f') = min\{\psi(f) \mid f \in S\}$. Presently assume S is produced by $p, q \in E$ and there is a $f \in F \backslash S$ dominated over f'. At that point this f too dominates p or q, which denies the presumption that p and q are efficient.

We have constructed linear objective function, by which we can determine the efficient point in S. We have to find another efficient point for this we first eliminate the previously found efficient point i.e. $p = (p_1, p_2)$. To find next efficient point we add a restriction $f_2 < p_2$. With addition of this restriction we get a subset S' by minimizing ψ over S', let p' be new efficient point. For using this idea to find all the efficient points we have to show that p' is the only efficient point found after adding the restriction.

4) Let $p, q \in E$ with $p \neq q$, $S' = \{f \in F \mid p_1 \leq f_1 \leq q_1, q_2 \leq f_2 < p_2\}$ and $\psi(p') = min\{\psi(z) \mid f \in S'\}$. Then p' is the only efficient point eliminated by the restriction $f_2 < p'_2$.

Proof: Adding $f_2 < p_2$ to the limitations of the set S' the subsets S^2 and S^3 are cut off. In the accompanying we consider the subsets S^2 and S^3. The set S^2 is easy to handle. All components of S^2 are dominated by p' and in this manner cutting off the set S^2 no efficient point will be lost. Presently we show that S^3 has no efficient points. Assume there is a $f \in S^3$. At that point, $\psi(f) - \psi(p) = (f_2 - p_2) + (b' - a' + 1)(f_1 - p_1) < (f_2 - p_2) + (b' - a' + 1)(-1) < 0$ and $\psi(f) < \psi(p)$ in contradiction to p is negligible. Thus S^3 is empty and contains no efficient points.

Now we can give the following algorithm to determine all efficient points.

2.4.2 Algorithm

Step 1: /* Calculate extreme efficient points $k^1 = (k_1^1, k_2^1)$ and $k_2 = (k_1^2, k_2^2)$ */

$k_1^2 = min\{f_1(x) \mid y \in Y\}$

$k_2^2 = min\{f_2(x) \mid y \in Y \ and \ f_1(x) = k_1^2\}$

$k_2^1 = min\{f_2(x) \mid y \in Y\}$

$k_1^1 = min\{f_1(x) \mid y \in Y \ \ and \ \ f_2(y) = k_2^1\}$

$k^1 = (k_1^1, k_2^1)$, $k_2 = (k_1^2, k_2^2)$

if $k^1 = k^2$ **then** $i = 2$, $E = \{k^2\}$ **STOP**

else $i = 2$, $E = \{k^1, k^2\}$

Step 2:/* Look for efficient points using $\psi(f) = f_2 + (k_2^2 - k_2^1 + 1)f_1$ and reducing the feasible set successively*/

for $\psi(k^i) < \psi(k^1)$ **do**

$S = \{f \in F \mid k_1^2 \leq f_1 \leq k_1^1 \;\; and \;\; k_2^1 \leq f_2 < k_2^i\}$

f' =optimize $(\psi(f), S)$

if $\psi(f') < \psi(k^1)$ then $i = i + 1, k^1 = f', E = E \cup \{k^i\}$

else $i = 1$

/* Output all efficient points */

for $(j = 1...., i)$ do {print k^j}

2.4.3 Example

Now we will apply reduction algorithm on our bi-criteria integer linear problem.

Min $F = (f_1, f_2)$ where $f_1 = y_1 - 2y_2$ and $f_2 = -3y_1 + 2y_2$

$$\begin{aligned} \text{s.t.} \quad -y_1 + y_2 &\leq 2 \\ y_1 + y_2 &\leq 4 \\ y_1, y_2 &\geq 0 \end{aligned}$$

Now will use above algorithm to find extreme efficient and all other efficient points.

Step 1: Extreme efficient Points:

Let we have two extreme efficient points $k^1 = (k_1^1, k_2^1)$ and $k^2 = (k_1^2, k_2^2)$.

k^1 = Minimize f_1 and take f_2 as restriction or other constraint.

$k_2^1 = min\{f_2(x) \mid y \in Y\} = \{-12\}$

$k_1^1 = min\{f_1(x) \mid y \in Y \;\; and \;\; f_2(x) = k_2^1\}$ = min $\{f_1 \;\; and \;\; -3y_1 + 2y_2 = -12\}$

$$\begin{aligned} &\text{Min} \;\; y_1 - 2y_2 \\ &\text{s.t.} \;\; -y_1 + y_2 \leq 2 \\ &\quad y_1 + y_2 \leq 4 \\ &\quad -3y_1 + 2y_2 = -12 \\ &\quad y_1, y_2 \geq 0 \end{aligned}$$

solving this we get $k_1^1 = 4$

i.e. $k^1 = (4, -12)$ at $(4, 0) \in Y$

$k^2 =$ Minimize f_2 and take f_1 as restriction or other constraint.

$k_1^2 = \min\ \{f_1(x) \mid y \in Y\} = -5$

$k_2^2 = \min\ \{f_2(x) \mid y \in Y \ \ and \ \ f_1(x) = k_1^2\} = min\{f_2(x) \mid y \in Y \ and \ y_1 - 2y_2 = -5\}$

$$\begin{aligned} \text{Min} \quad & -3y_1 + 2y_2 \\ \text{s.t.} \ & -y_1 + y_2 \leq 2 \\ & y_1 + y_2 \leq 4 \\ & -3y_1 + 2y_2 = -12 \\ & y_1, y_2 \geq 0 \end{aligned}$$

solving this we get $k_2^2 = 3$

i.e. $k^2 = (-5, 3)$ at $(1, 3) \in Y$

Hence, we get $k^1 = (4, -12)$ and $k^2 = (-5, 3)$ as extreme efficient points. Update $i = 2$

$E = \{(4, -12), (-5, 3)\}$.

Step 2: Now we will find all the other efficient points.

$$\begin{aligned} \psi(f) &= f_2 + (k_2^2 - k_2^2 + 1)f_1 \\ &= (3y_1 + 2y_2) + (3 - (-12) + 1)(y_1 - 2y_2) \\ &= (3y_1 + 2y_2) + 16y_1 - 32y_2 \\ &= 13y_1 - 30y_2 \end{aligned}$$

Now, we have to check if $\psi(k^2) < \psi(k^1)$ is true or not.

$\psi(1, 3) = -77, \psi(4, 0) = 12$

Therefore, $\psi(k^2) < \psi(k^1)$ is true. Now we construct a set S and optimize $\{\psi(f), S\}$ i.e.

$$\begin{aligned} f' = \min\ & 13y_1 - 30y_2 \\ & -y_1 + y_2 \leq 2 \\ & y_1 + y_2 \leq 4 \\ & -5 \leq y_1 - 2y_2 \leq 4 \\ & -12 \leq -3y_1 + 2y_2 \leq 2 \\ & y_1, y_2 \geq 0 \end{aligned}$$

solving this we get $f' = -47$, at (1,2) $\in Y$

therefore, $\psi(f') = -47$, $\psi(k^1) = 12$, $\psi(f') < \psi(k^1)$. As $\psi(f') < \psi(k^1)$ is true we can proceed our algorithm further. Update $i = i + 1 = 3$

We can find k^3 as below:

$f' = -47$ at $(1, 2)$ therefore we will evaluate f_1 and f_2 at (1,2). The values of f_1 and f_2 at (1,2) will correspondingly give k^3. Hence, $f_1 = -3$, $f_2 = 1$ at (1,2). As $k^3 = f'$ at (1,2), therefore $k^3 = (-3, 1)$ is new efficient point. Update $E = E \cup \{k^3\}$.

This gives $E = \{(4, -12), (-5, 3), (-3, 1)\}$. Applying the above procedure again and again we get $E = \{(4, -12), (-5, 3), (-3, 1), (0, -4), (1, -7), (-2, -2)(3, -9)\}$.

In our bi-criteria linear problem the two extreme points are $\{4, -12\}$ and $\{-5, 3\}$. Therefore we have, $a' = -5$, $b' = 3$ and the new objective function $\psi(f_1, f_2) = f_2 + 9f_1$. With this function we get the following order $\psi(-2, -2) < \psi(0, -4) < \psi(1, -7) < \psi(3, -9) < \psi(4, 12) < \psi(-3, 1) < \psi(-5, 3)$.

Result: Above mentioned algorithm gives all efficient points.

Proof: Let e^1 and e^2 are two efficient points and if $e^1 = e^2$ then it is an ideal point and also the only efficient point of this optimization problem. Now consider the case when $e^1 \neq e^2$. The procedure optimizes in the following loop only and determines efficient points, by the result of proposition 3) and hence all efficient points are found in E.

Now, in next step we will show that the algorithm finds all the efficient points. Consider that an efficient point e is not found in E by this algorithm. For simplicity we suppose $S^{(i)} = \{f \in F \mid e_1^2 \leq f_1 \leq e_1^1 \ \ and \ \ e_2^1 \leq f_2 \leq e_2^1\}$. With the subsets $S^{(i)}$ having the property $S^{(n)} \subset ...S^{(i+1)} \subset S^{(i)} \subset ... \subset S^{(2)}$. Now we can obtain a subset $S^{(i)}$ such that $e \in S^{(i)}$ and $e \notin S^{(i+1)}$. If the algorithm can find the efficient point $e^{(i)}$ then the subset $S^{(i)}$ is obtained and the algorithm can find the next efficient point $e^{(i+1)}$. Since $e \in S^{(i)}$ and $e \notin S^{(i+1)}$ we can have $e_2^1 \leq e_2 \leq e_2^{(i+1)}$ and because of efficiency it follows $e_1^1 \leq e_1 \leq e_1^{(i+1)}$. Therefore, we have $\psi(e) \leq \psi(e^{(i+1)})$ which is contradiction to the minimality condition of $\psi(e^{(i+1)})$. Due to this contradiction there cannot exist an efficient point e and all efficient points are obtained by above algorithm. Finally, the number of efficient points are obtained and the algorithm terminates after a finite number of iterations.

Chapter 3

Optimizing a Linear Fractional function over the integer efficient set

In this chapter we are reviewing, the idea of Wassila Drici; Fatma Zohra Ouail; Mustapha Moulai[27] of optimizing a linear fractional function over the integer efficient set. Basically in this we have a multi-objective integer linear problem defined on a feasible region and a linear fractional problem whose efficient solutions exist in feasible region of multi-objective problem. This is done by using Branch and Cut technique and the continuous linear fractional programming. In this we add efficient cuts and efficiency test is also applied by which large number of non-efficient points are left out.
Multi-objective optimization problems, are those in which we optimize, several objectives functions which can be linear or non-linear. We cannot find one optimal solution for multi-objective problems, so we search for a set of solutions called efficient solutions. Previously, these type of problems became an important area of multi-objective optimization. Optimizing a non linear function over the efficient set of a Multi-objective Integer Linear Programming problem (MOILP) is a not easy to solve. Because of discrete aspect of decision variables, and also because the shape of feasible region having efficient set is unknown. In general, the set of efficient solutions may have very large cardinality, this made difficult for the decision maker (DM) to find the best solution. In spite of that the decision maker can create his predictions using one function apart from multi-objective integer linear problem functions. This chapter includes an algorithm to obtain

an optimal solution of a maximization linear fractional problem over a non-convex set of efficient solutions of multi-objective integer linear problem.
In this we solve integer linear fractional problem, using the Branch and Bound method which is strengthened by the addition of efficient cuts and tests. In the next section we will give some definitions and notations regarding the multi-objective integer linear problem and integer linear fractional problem. In section (3.2), the algorithm is discussed. The section (3.3) consists of theoretical results for the algorithm and the section (3.4) contains example.

3.1 Definitions and Notations:

Consider a multi-objective integer linear problem:

MOILP $\quad max \quad f_i(x) = \sum a^i x^i; \quad i \in (1, 2, ..., s)$

$$\text{subject to } x \in E = X \cap \mathbb{Z}^n$$

where the set of integers is denoted by $\mathbb{Z}$ and $s \geq 2$; $A \in \mathbb{R}^{m\times n}$; $b \in \mathbb{R}^n$ and $X = \{x \in \mathbb{R}^n \mid Ax \leq b, x \geq 0\}$ and $X \neq \phi$ and X is convex- polyhedron set also $E \neq \phi$.
Efficient Solution: Consider $x^* \in D$ be a feasible solution of multi-objective integer linear problem the x^* is called an efficient solution iff there exist no other feasible solution such that $f_i(x) \geq f_i(x^*)$ where $x \in E$ and $i \in (1, 2, ..., s)$ and $f_t(x) \geq f_t(x^*)$ for at least one $t \in (1, 2, ..., s)$. If these two conditions are not satisfied then we can say x^* is not an efficient point and then $f_i(x^*)$ is dominated by $f_i(x)$.
Now, consider a linear fractional problem which is not necessarily a combination of multi-objective integer linear problem's objective functions and integer linear fractional problem is maximized by an efficient solution of multi-objective integer linear problem. This can be determined by solving the following problem:

(ILFP) $\quad \max \dfrac{cx + d}{px + q}$

subject to

$$x \in X_E \subset E$$

where c, d are real n vectors, $\alpha, \beta \in \mathbb{R}$ be any real constants and X_E is efficient set of multi-objective integer linear problem. The denominator of the objective function of integer linear fractional problem is always strictly positive for all $x \in X$. Let x_{opt} is the best efficient solution of integer linear fractional problem (ILFP) obtained upto step k and f_{opt} is its respective value.

Now, we apply Branch and Bound method, by this we will obtain an optimal solution for integer linear fractional problem (ILFP) by solving various continuous and more constrained linear fractional problems $(LFP)_k$, $k \geq 0$. These $(LFP)_k$'s can be defined as:

$(LFP)_k$
$$\max \frac{cx+d}{px+q}$$
subject to
$$x \in X_k$$

where $X_0 = X$ and X_{k+1} can be determined by adding efficient cuts as discussed below.

Let x^{*k} be an optimal solution of $(LFP)_k$ at node k of the structured tree.

Case 1: If x^{*k} is integer we will test the efficiency of x^{*k} by solving mixed integer linear problem stated by Ecker and Kouada (1975) as:

$EK(x^{(*k)})$
$$\max Z = \sum_{1}^{s} \psi_i$$
subject to
$$Cx = I\psi + Cx^{(*k)}$$
$$x \in D; \;\; \psi_i \geq 0; \text{ for all } i \in (1,2,...,s)$$

where $C = (a^i)_{i\in(1,2,\ldots,s)}$ is $(s\times n)$ matrix, I is $(r\times r)$ identity matrix and $\psi = (\psi_i)_{i\in(1,2,\ldots,s)}$.

The efficiency condition for a point to be efficient is that $Z = 0$. If this condition is not satisfied than x^{*k} is not an efficient point and then we will use $EK(x^{*k})$ to obtain efficient points.

Case 2: The x^{*k} is not integer, then branching process will be used and then two new linear fractional problems will be created and solved as problem $(LFP)_k$.

Let x^{*k} be the first integer solution obtained after solving problem $(LFP)_k$ by using the branching process. Let B_k denotes the set of all basic variables and N_k denotes the set of all non-basic variables of x^{*k}.

Let δ_j be the jth component of the growth vector δ of function defined, at every step as:
$$\delta_j = f_2(c - z_j^{(1)}) - f_1(p - z_j^{(2)})$$
where $z_j^{(1)} = c_B B^{-1} b_j$; $z_j^{(2)} = p_B B^{-1} b_j$; b_j is the jth component of matrix A; $f_1 = cx^{*(k)} + d$; $f_2 = px^{*(k)} + q$; and $f(x^{*(k)}) = \dfrac{f_1}{f_2}$.

Necessary and Sufficient condition for optimality: The necessary and sufficient condition for optimality for $(LFP)_k$ can be stated as:
A basic feasible solution x^* is an optimal solution of the problem $(LFP)_k$ iff the component $\delta_j \leq 0$ for all $j \in N_k$.
Proof: The growth rate of each function $f_i, i \in (1, 2, ..., s)$ of multi-objective integer linear problem can be obtained using their reduced gradient vectors $a^{-1} = a^i - f^i, i \in \{1, 2, ..., s\}$. Now using this information we make an efficient cut and remove those integer solutions which are not efficient. Therefore, we define the set of increasing directions of the criteria as follows:
$S_k = \{j \in N_k \ / \ \exists\{1, 2, ..., s\}; a_j^{-i} > 0\} \cup \{j \in N_k / a_j^{-i} = 0, \forall i \in \{1, 2, ..., s\}\}$.
Now there are two types of efficient cuts are present.
Cut 1:

$$\sum_{j \in S_k} x_j \geq 1$$

Cut 2: This cut is added in accordance with the following condition:

$$f(x) \geq f_{opt}$$

This method is based on the criteria of Branch and Cut method. Therefore, in each step we add efficient cuts to avoid those solutions which are not efficient for multi-objective integer linear problem.
If we obtain an integer solution at node k we generate the following set

$$X_{k+1}^1 = \{x \in X_k / \sum_{j \in S_k} x_j \geq 1\},$$

and the following set

$X_{k+1}^2 = \{x \in X_k / f(x) \geq f_{opt}\}$. which can be written as:

$$X_{k+1} = X_{k+1}^1 \cup X_{k+1}^2.$$

3.2 Description of the method:

We suppose that $(LFP)_k$ is considered as a node of the structured tree. A node k of the structured tree is fathomed if the following conditions are satisfied:

1) If the solution x^{*k} of the corresponding problem integer linear fractional problem is an efficient solution ($x^{*(k)} \in X_E$).

2) If $x^{*(k)}$ is a non efficient integer solution and $S_k = \phi$, which means that no improvement of the objective linear functions can be done along remaining domain.

3) If f_{opt} obtained is greater than or equal to value f at the node.

4) If the corresponding program $(LFP)_k$ is unfeasible.

Now, consider the case when $x^{*(k)}$ is not integer optimal solution of integer linear fractional problem.

Let $x_j^{*(k)}$ be one of the component of $x^{*(k)}$ and let $x_j^{*(k)}$ is fractional number then we can write $x_j \geq \lfloor x_j^{*(k)} \rfloor$ and $x_j \geq \lceil x_j^{*(k)} \rceil$ where and are floor and celling functions respectively. Using Branch and Bound method we found the integer solution and now we will check efficiency of this integer solution using $EK(x^{*(k)})$ problem. If it is efficient then we will update the current value.

If the obtained solution is not efficient we will add efficient cuts and we will now solve a new problem. This algorithm ends when all the nodes are fathomed. Hence, x_{opt} will be the optimal solution of the integer linear fractional problem with respect to f_{opt}.

3.3 Algorithm:

Step 1: Initialization Set the optimal value of f as $f_{opt} = -\infty$, x_{opt} as unknown solution, $k = 0$ and $X_0 = X$.

Step 2: General Step Until there's presence of non-fathomed node in tree, that node is chosen which is not yet fathomed which is having greatest number k and corresponding $(LFP)_k$ is solved with the help of dual simplex method or any other method.

Now, after solving $(LFP)_k$ two cases arise:

Case 1: if the given problem is unfeasible, then we have fathomed node k.

Case 2: if the given problem is feasible, then we suppose its optimal solution to be x^{*k}.

Now again two cases arise:

Case 2a: If $f_{opt} \geq f_{x^{*(k)}}$, then the node k is fathomed. We will stop here and repeat step2.

Case 2b: If $f_{opt} \ngeq f_{x^{*(k)}}$, we will check whether $x^{*(k)}$ is integer or not. Again two cases arise:

If $x^{*(k)}$ is integer:

We will now test for efficiency of $x^{*(k)}$.

Efficiency test: For efficiency of $x^{*(k)}$ solve $EK(x^{*(k)})$.

$EK(x^{(*k)})$

$$\max Z = \sum_{1}^{s} \psi_i$$

subject to

$$Cx = I\psi + Cx^{(*k)}$$

$$x \in D; \quad \psi_i \geq 0; \forall\ i \in (1, 2, ..., s)$$

where $C = (a^i)_{i\in\{1,2,...,s\}}$ is $(s\times n)$ matrix, I is $(r\times r)$ identity matrix and $\psi = (\psi_i)_{i\in\{1,2,...,s\}}$.

After solving $EK(x^{(*k)})$ two cases arise:

1: If $Z = 0$, we say solution $x^{*(k)}$ is efficient then corresponding node k is fathomed. We will update f_{opt}; $iff\ f_{opt} \geq f(x)$ and we will stop here and repeat from step 2.

2: If $Z \neq 0$, then we will find he optimal solution by solving $EK(x^{(*k)})$ let it be $\hat{x}^k$ update f_{opt}, $iff\ f_{opt} \geq f(x)$ and now we will add efficient cuts to $(LFP)_k$.

Efficient cuts: Determine the set N_k and construct S_k;

Now, again two cases arise if:

1) $S = \phi$, then k node is fathomed, repeat step 2.

2) $S \neq \phi$, add cut $\sum_{j\in S_k} x_j \geq 1$ to $(LFP)_k$

and if $f(x) \geq f_{opt}$ then add

$$\sum_{j\in S_k} x_j \geq 1$$

and $f(x) \geq f_{opt}$, repeat step 2.

If $x^{*(k)}$ is not an integer:

We will apply Branch and Bound method. Choose any non-integer value from the components of $x^{*(k)}$. Now add two constraints $x_j \geq \lfloor x_j^{*(k)} \rfloor$ to get $(LFP)_{k_1}$ and $x_j \geq \lceil x_j^{*(k)} \rceil$ to get $(LFP)_{k_2}$ such that $(k_1 > k+1; k_2 > k+1)$ and $k_1 \neq k_2$, hence dividing the feasible set X_k into two parts X_{k_1} and X_{k_2}, doing so we can add cut

$$\sum_{j\in S_k} x_j \geq 1$$

and $f(x) \geq f_{opt}$ and repeat step 2.

3.4 Theoretical Results

The algorithm provided above generates an optimal efficient solutions of integer linear fractional problem in finite number of steps.

1) Let $S_k \neq \phi$ at current integer solution $x^{*(k)}$. If $x \neq x^{*(k)}$ is an integer efficient solution in domain $X_k \backslash \{x^{*(l)}\}$, then $x \in X_{k+1}$.

Proof: Let x be an integer solution of the set $X_k \backslash \{x^{*(l)}\}$ such that $x \notin X_{k+1}$, then $x \notin X^1_{k+1}$ or $x \notin X^2_{k+1}$.

We will prove this by contradiction, if $x \notin X^1_{k+1}$, then $x \notin \{x \in X_k / \sum_{j \in S_k} x_j \geq 1\}$ implies $x \in \{x \in X_k / \sum_{j \in N_k \backslash S_k} x_j \geq 1\}$.

Hence, the following holds:

$$\sum_{j \in S_k} x_j < 1$$

and

$$\sum_{j \in N_k \backslash S_k} x_j \geq 1$$

It follows that $x_j = 0$ for all index $j \in S_k$, and $x_j \geq 1$ for at least one index $j \in N_k \backslash S_k$.

By using the simplex table in $x^{*(k)}$, the following equality holds $\forall\ i \in \{1, 2, ..., s\}$:

$$\begin{aligned}
f_i(x) = a^i x = a^i x^{*(k)} + \sum_{j \in S_k} a_j^{-i} x_j \\
= a^i x^{*(k)} + \sum_{j \in S_k} a_j^{-i} x_j + \sum_{j \in N_k \backslash S_k} a_j^{-i} x_j \\
= a^i x^{*(k)} + \sum_{j \in N_k \backslash S_k} a_j^{-i} x_j
\end{aligned}$$

This implies $f_i(x) \leq f_i(x^{*(k)})$ $\forall i \in \{1, 2, ..., s\}$, with $f_i(x) \leq f_i(x^{*(k)})$ for at least one criterion since $a_j^{-i} \leq 0, \forall j \in N_k \backslash S_k$.

Hence, $f_i(x)$ is dominated by $f_i(x^{*(k)})$ and x is not efficient. This is contradiction. Hence, $x \in X^1_{k+1}$.

Similarly, if $x \notin X^2_{k+1}$, $f(x) < f_{opt}$. Thus, x is not optimal, a contradiction.

Result: The constraint

$$\sum_{j \in S_k} x_j \geq 1$$

defines an efficient cut.

Proof: By the last theorem, we can say that

$$\sum_{j \in S_k} x_j \geq 1$$

is an efficient valid constraint and all the integer efficient solutions in the current domain X_k satisfy this constraint. Also, the current integer solution $x^{*(k)}$ does not satisfy this constraint since $x_j = 0 \; \forall j \in S_k$. Hence, we can say that the constraint

$$\sum_{j \in S_k} x_j \geq 1$$

is an efficient cut.

2) The algorithm converges to an optimal solution of integer linear fractional problem, this solution if exist have finite number of iterations.

Proof: As, E, the set of the integer feasible solutions of multi-objective integer linear problem, is a finite bounded set contained in X, the number of elements in X_E is also finite. Every time we get an optimal integer solution $x^{*(k)}$, an efficient cut $x^{*(k)} \in X_E$ is added or there is no further improvement of f_{opt}. Therefore, in accordance to above theorem and corollary, at least the solution $x^{*(k)}$ is eliminated when any sub-problem like linear fractional problem $(LFP)_l$ is studied where $l > k$, but no integer efficient solution is omitted.

3.5 Example

Now, we will apply above algorithm to solve the following problem:

Consider the multi-objective integer linear problem:

$$(P) \qquad \max\ f_1 = y_1 + y_2 - y_3 - 2y_4$$
$$\max\ f_2 = y_1 - 3y_2 - y_3 + y_4$$

$$\max\ f_3 = 0y_1 + 2y_2 + 2y_3 + 2y_4$$

$$\max\ f_4 = 0y_1 - y_2 + 2y_3 + 2y_4$$

s.t.

$$3y_1 - 2y_2 + y_3 + x_4 \leq 3$$

$$y_2 + 3y_3 + 4y_4 \leq 4$$

$$y_1 + y_2 + y_3 + y_4 \leq 5$$

$$2y_2 + y_3 \leq 6$$

$$y_1, y_2, y_3, y_4 \in N$$

$$y_1, y_2, y_3, y_4 \geq 0$$

Now, the linear fractional problem is given below as:

$(ILFP)_E$

$$\max\ z(x) = \frac{-y_1 - y_2 + 2y_3 + 3y_4 - 15}{y_2 + y_3 + y_4 + 1}$$

s.t.

$$x \in X_E$$

Initialization: Let $f_{opt} = -\infty$, $k = 0$. The $(LFP)_0$ is now solved i.e.

Step 1:

$(LFP)_0$

$$\max\ z(x) = \frac{-y_1 - y_2 + 2y_3 + 3y_4 - 15}{y_2 + y_3 + y_4 + 1}$$

s.t.

$$3y_1 - 2y_2 + y_3 + x_4 \leq 3$$

$$y_2 + 3y_3 + 4y_4 \leq 4$$

$$y_1 + y_2 + y_3 + y_4 \leq 5$$

$$2y_2 + y_3 \leq 6$$

$$y_1, y_2, y_3, y_4 \in N$$

$$y_1, y_2, y_3, y_4 \geq 0$$

we get the following table 1

		d_j	0	1	1	1	0	0	0	0	
		c_j	-1	-1	2	3	0	0	0	0	
d_B	c_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	x_B
0	0	y_7	1	0	0	1/5	0	-1/5	1	-2/5	9/5
2	1	y_3	0	0	1	8/5	0	2/5	0	-1/5	2/5
1	-1	y_2	0	1	0	-4/5	0	-1/5	0	3/5	14/5
0	0	y_5	3	0	0	-11/5	1	-4/5	0	7/5	41/5
	$z_1 = -17$	$z_j^1 - c_j$	1	0	0	1	0	1	0	-1	
	$z_2 = 21/5$	$z_j^2 - d_j$	0	0	0	-1/5	0	1/5	0	2/5	
		$\triangle_j$	-21/5	0	0	-4/5	0	-38/5	0	-13/5	
		$\bar{a_1}$	1	0	0	2/5	0	3/5	0	-4/5	
		$\bar{a_2}$	1	0	0	1/5	0	-1/5	0	8/5	
		$\bar{a_3}$	0	0	0	2/5	0	-2/5	0	-4/5	
		$\bar{a_4}$	0	0	0	-2	0	-1	0	1	

Table 1

we get $y^{*0} = (0, \frac{14}{5}, \frac{2}{5}, 0)$ as first optimal solution and $f(x^{*0}) = -\dfrac{85}{21}$.
The first optimal solution obtained above is not integer, therefore we apply branching process. After applying branching process we get two different nodes constraints such as:

$$C_1 : y_2 \leq \lfloor \frac{14}{5} \rfloor \qquad (1)$$

$$C_2 : y_2 \geq \lceil \frac{14}{5} \rceil \qquad (2)$$

Adding constraint (1) to table 1 we get the following table 2:

		d_j	0	1	1	1	0	0	0	0	0	
		c_j	-1	-1	2	3	0	0	0	0	0	
d_B	c_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	x_B
0	0	y_7	1	0	0	-1/3	0	-1/3	1	0	-2/3	7/3
2	1	y_3	0	0	1	4/3	0	1/3	0	0	-1/3	2/3
1	-1	y_2	0	1	0	0	0	0	0	0	1	2
0	0	y_5	3	0	0	-1/3	1	-1/3	0	0	7/3	19/3
0	0	y_8	0	0	0	-4/3	0	-1/3	0	1	-5/3	4/3
	$z_1 = -47/3$	$z_j^1 - c_j$	1	0	0	-1/3	0	2/3	0	0	-5/3	
	$z_2 = 11/3$	$z_j^2 - d_j$	0	0	0	1/3	0	1/3	0	0	5/3	
		$\triangle_j$	-11/3	0	0	-4	0	-23/3	0	0	-13/3	
		$\bar{a_1}$	1	0	0	-2/3	0	1/3	0	0	-4/3	
		$\bar{a_2}$	1	0	0	7/3	0	1/3	0	0	8/3	
		$\bar{a_3}$	0	0	0	-2/3	0	-2/3	0	0	-4/3	
		$\bar{a_4}$	0	0	0	-2/3	0	-2/3	0	0	5/3	

Table 2

This table gives $y^{*1} = (0, 2, \frac{2}{3}, 0)$ as an optimal solution and $f(y^{*1}) = -\dfrac{47}{11}$.
Adding constraint (2) to table 1 we get the following table 3:

		d_j	0	1	1	1	0	0	0	0	0	
		c_j	-1	-1	2	3	0	0	0	0	0	
d_B	c_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	x_B
0	0	y_7	1	0	0	0	0	-1/4	1	-1/4	1/4	7/4
2	1	y_3	0	0	1	0	0	0	0	1	2	0
1	-1	y_2	0	1	0	0	0	0	0	0	-1	3
0	0	y_5	3	0	0	0	1	-1/4	0	-1/4	-11/4	35/4
1	3	y_4	0	0	0	1	0	1/4	0	-3/4	-5/4	1/4
	$z_1 = -69/4$	$z_j^1 - c_j$	1	0	0	0	0	3/4	0	-1/4	5/4	
	$z_2 = 17/4$	$z_j^2 - d_j$	0	0	0	0	0	1/4	0	1/4	-1/4	
		$\triangle_j$	-17/4	0	0	0	0	-15/2	0	-13/4	-1	
		$\bar{a_1}$	1	0	0	0	0	1/2	0	-1/2	1/2	
		$\bar{a_2}$	1	0	0	0	0	-1/4	0	7/4	1/4	
		$\bar{a_3}$	0	0	0	0	0	-1/2	0	-1/2	1/2	
		$\bar{a_4}$	0	0	0	0	0	-1/2	0	-1/2	-5/2	

Table 3

This table gives $y^{*2} = (0, 3, 0, \frac{1}{4})$ as an optimal solution and $f(y^{*2}) = -\dfrac{69}{17}$.

From above tables we get $f(y^{*2}) > f(y^{*1})$, therefore, we will explore C_2 node first but the solution obtained by adding C_2 is not an integer therefore again we apply branching procedure and we get the following constraints:

$$C_3 : y_4 \leq \lfloor \frac{1}{4} \rfloor \qquad (3)$$

$$C_4 : y_4 \geq \lceil \frac{1}{4} \rceil \qquad (4)$$

Now we will add C_3 and C_4 to table 3. Adding C_3 to table 3 we get table 4 as below:

		d_j	0	1	1	1	0	0	0	0	0	0	
		c_j	-1	-1	2	3	0	0	0	0	0	0	
d_B	c_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}	x_B
0	0	y_7	1	0	0	0	0	0	1	-1	-1	-1	2
2	1	y_3	0	0	1	0	0	0	0	1	2	0	0
1	-1	y_2	0	1	0	0	0	0	0	0	-1	0	3
0	0	y_5	3	0	0	0	1	0	0	-1	-4	-1	9
1	3	y_4	0	0	0	1	0	0	0	0	0	1	0
0	0	y_6	0	0	0	0	0	1	0	-3	-5	-4	1
	$z_1 = -18$	$z_j^1 - c_j$	1	0	0	0	0	0	0	2	5	3	
	$z_2 = 4$	$z_j^2 - d_j$	0	0	0	0	0	0	0	1	1	1	
		$\triangle_j$	-4	0	0	0	0	0	0	-26	-38	-1	
		$\bar{a_1}$	1	0	0	0	0	0	0	1	3	2	
		$\bar{a_2}$	1	0	0	0	0	0	0	1	-1	-1	
		$\bar{a_3}$	0	0	0	0	0	0	0	-2	-2	-2	
		$\bar{a_4}$	0	0	0	0	0	0	0	-2	-5	-2	

Table 4

This table gives $y^{*3} = (0, 3, 0, 0)$ as an optimal solution and $f(y^{*3}) = -\dfrac{18}{2}$.
Adding constraint C_4 to table 3 we get unfeasible solution, therefore, the node is fathomed. As we obtained optimal integer solution after adding constraint C_3 to table 3 i.e. $y^{*3} = (0, 3, 0, 0)$ we will now test the efficiency of this solution by applying Ecker and Kouada problem.

Efficiency test for $y^{*3} = (0, 3, 0, 0)$: To check the efficiency of $y^{*3} = (0, 3, 0, 0)$ we have to solve the following problem:

EK($y^{*(l)}$) $\qquad \max \theta = w_1 + w_2 + w_3 + w_4$

s.t.

$$3y_1 - 2y_2 + y_3 + x_4 \leq 3$$
$$y_2 + 3y_3 + 4y_4 \leq 4$$
$$y_1 + y_2 + y_3 + y_4 \leq 5$$
$$2y_2 + y_3 \leq 6$$

$$y_1 + y_2 - y_3 - 2y_4 - w_1 = 3$$
$$y_1 - 3y_2 - y_3 + y_4 - w_2 = -9$$
$$0y_1 + 2y_2 + 2y_3 + 2y_4 - w_3 = 6$$
$$0y_1 - y_2 + 2y_3 + 2y_4 - w_4 = -3$$
$$y_1, y_2, y_3, y_4 \in N$$
$$y_1, y_2, y_3, y_4, w_1, w_2, w_3, w_4 \geq 0$$

After solving this problem using TORA we get the following result:

subproblem	objective value, θ	y_1	y_2	y_3	y_4	w_1	w_2	w_3	w_4
5	2	1	3	0	0	1	1	0	0
7	4	2	3	0	0	2	2	0	0

The above table shows objective function value is not equal to zero. Hence, the optimal solution $y^{*3} = (0, 3, 0, 0)$ is not efficient. After applying efficiency test we get an efficient integer solution i.e. $\hat{y^3}$= (2,3,0,0) with $f(\hat{y^3}) = -5$. Update y_{opt}= (2,3,0,0) and $f_{opt} = -5$. To obtain the optimal solution we will add cuts to table 4. From table 4 we obtain $S_1 = \{1, 8, 9, 10\}$. Apply the efficient cut $y_1 + y_8 + y_9 + y_{10} \geq 1$ and $f(y) \geq -5$. After adding both the cuts in table 4 and applying dual simplex method we obtain table 5:

d_B	c_B		y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}	y_{11}	y_{12}	x_B
		d_j	0	1	1	1	0	0	0	0	0	0	0	0	
		c_j	-1	-1	2	3	0	0	0	0	0	0	0	0	
d_B	c_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}	y_{11}	y_{12}	x_B
0	0	y_7	1	0	0	0	0	0	1	-2	-2	-2	1	0	1
2	1	y_3	0	0	1	0	0	0	0	1	2	0	0	0	0
1	-1	y_2	0	1	0	0	0	0	0	0	-1	0	0	0	3
0	0	y_5	0	0	0	0	1	0	0	-4	-7	-4	3	0	6
1	3	y_4	0	0	0	1	0	0	0	0	0	1	0	0	0
0	0	y_6	0	0	0	0	0	1	0	-3	-5	-4	0	0	1
0	-1	y_1	1	0	0	0	0	0	0	1	1	1	-1	0	1
0	0	y_{12}	0	0	0	0	0	0	0	6	9	7	1	1	1
	$z_1 = -19$	$z_j^1 - c_j$	0	0	0	0	0	0	0	1	4	2	1	0	
	$z_2 = 4$	$z_j^2 - d_j$	0	0	0	0	0	0	0	1	1	1	0	0	
		$\triangle_j$	0	0	0	0	0	0	0	-23	-35	-27	-4	0	
		$\bar{a}_1$	0	0	0	0	0	0	0	0	2	1	1	0	
		$\bar{a}_2$	0	0	0	0	0	0	0	0	-2	-2	1	0	
		$\bar{a}_3$	0	0	0	0	0	0	0	-2	-2	-2	0	0	
		$\bar{a}_4$	0	0	0	0	0	0	0	-2	-5	-2	0	0	

<u>Table 5</u>

This table gives $y^{*4} = (1, 3, 0, 0)$ as an optimal solution and $f(y^{*4}) = -\dfrac{19}{4}$. Now, we will test the efficiency of $y^{*4} = (1, 3, 0, 0)$.

Efficiency test for $y^{*3} = (0, 3, 0, 0)$: To check the efficiency of $y^{*3} = (0, 3, 0, 0)$ we have to solve the following problem:

EK($y^{*(l)}$)

$$\max \ \theta = w_1 + w_2 + w_3 + w_4$$

s.t.

$$3y_1 - 2y_2 + y_3 + x_4 \leq 3$$

$$y_2 + 3y_3 + 4y_4 \leq 4$$

$$y_1 + y_2 + y_3 + y_4 \leq 5$$

$$2y_2 + y_3 \leq 6$$

$$y_1 + y_2 - y_3 - 2y_4 - w_1 = 3$$

$$y_1 - 3y_2 - y_3 + y_4 - w_2 = -9$$
$$0y_1 + 2y_2 + 2y_3 + 2y_4 - w_3 = 6$$
$$0y_1 - y_2 + 2y_3 + 2y_4 - w_4 = -3$$
$$y_1, y_2, y_3, y_4 \in N$$
$$y_1, y_2, y_3, y_4, w_1, w_2, w_3, w_4 \geq 0$$

After solving this problem using TORA we get the following result:

subproblem	objective value, θ	y_1	y_2	y_3	y_4	w_1	w_2	w_3	w_4
4	0	1	3	0	0	0	0	0	0
5	2	2	3	0	0	1		0	0

The above table shows objective function value is not equal to zero. Hence, the optimal solution $y^{*4} = (1, 3, 0, 0)$ is not efficient. After applying efficiency test we get an efficient integer solution i.e. $\hat{y^4}$= (2,3,0,0) with $f(\hat{y^4}) = -5$. Update y_{opt}= (2,3,0,0) and $f_{opt} = -5$. To obtain the optimal solution we will add cuts to table 5. From table 5 we obtain $S_2 = \{9, 10, 11\}$. Apply the efficient cut $y_9 + y_{10} + y_{11} \geq 1$. After adding the cut in table 5 and applying dual simplex method we obtain table 6:

		d_j	0	1	1	1	0	0	0	0	0	0	0	0	0	0	
		c_j	-1	-1	2	3	0	0	0	0	0	0	0	0	0	0	
d_B	c_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}	y_{11}	y_{12}	y_{13}	y_{14}	x_B
0	0	y_7	1	0	0	0	0	0	1	-2	-3	-3	0	0	1	0	0
2	1	y_3	0	0	1	0	0	0	0	1	2	0	0	0	0	0	0
1	-1	y_2	0	1	0	0	0	0	0	0	-1	0	0	0	0	0	3
0	0	y_5	0	0	0	0	1	0	0	-4	-10	-7	0	0	3	0	3
1	3	y_4	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0
0	0	y_6	0	0	0	0	0	1	0	-3	-5	-4	0	0	0	0	1
0	-1	y_1	1	0	0	0	0	0	0	1	2	2	0	0	-1	0	2
0	0	y_{12}	0	0	0	0	0	0	0	6	8	6	0	1	1	0	0
0	0	y_{11}	0	0	0	0	0	0	0	0	1	1	1	0	-1	0	1
0	0	y_{14}	0	0	0	0	0	0	0	6	8	6	0	0	1	1	0
	$z_1 = -20$	$z_j^1 - c_j$	0	0	0	0	0	0	0	1	-3	-1	0	0	1	0	
	$z_2 = 4$	$z_j^2 - d_j$	0	0	0	0	0	0	0	1	-1	-1	0	0	0	0	
		$\triangle_j$	0	0	0	0	0	0	0	-24	-32	-24	0	0	-4	0	
		$\bar{a}_1$	0	0	0	0	0	0	0	0	1	0	0	0	1	0	
		$\bar{a}_2$	0	0	0	0	0	0	0	0	-3	-3	0	0	1	0	
		$\bar{a}_3$	0	0	0	0	0	0	0	-2	-2	-2	0	0	0	0	
		$\bar{a}_4$	0	0	0	0	0	0	0	-2	-5	-2	0	0	0	0	

Table 6

This table gives $y^{*4} = (2,3,0,0)$ as an integer efficient solution and $f(y^{*4}) = -\dfrac{20}{4}$. Hence, the current node is fathomed.

Now, we will consider the constraint C_1. The solution $y^{*1} = (0, 2, \frac{2}{3}, 0)$ is not integer, after applying branch and bound method we get two new constraints such as:

$$C_5 : y_3 \leq \lfloor \frac{2}{3} \rfloor \qquad (5)$$

$$C_6 : y_3 \geq \lceil \frac{2}{3} \rceil \qquad (6)$$

Adding constraint (5) to table 2 we get the following table 7:

		d_j	0	1	1	1	0	0	0	0	0	0	0	
		c_j	-1	-1	2	3	0	0	0	0	0	0	0	
d_B	c_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}	y_{11}	x_B
0	0	y_7	1	0	0	0	0	-1/4	1	0	-3/4	-1/4	0	5/2
2	1	y_3	0	0	1	0	0	0	0	0	0	1	0	0
1	-1	y_2	0	1	0	0	0	0	0	0	1	0	0	2
0	0	y_5	3	0	0	0	1	-1/4	0	0	9/4	-1/4	0	13/2
1	3	y_8	0	0	0	0	0	0	0	1	-2	-1	0	2
1	3	y_4	0	0	0	1	0	1/4	0	0	-1/4	-3/4	0	1/2
0	0	y_{11}	1	0	0	0	0	2	0	0	2	1	1	2
	$z_1 = -31/2$	$z_j^1 - c_j$	1	0	0	0	0	3/4	0	0	-7/4	-1/4	0	
	$z_2 = 7/2$	$z_j^2 - d_j$	0	0	0	0	0	1/4	0	0	3/4	1/4	0	
		$\triangle_j$	-7/2	0	0	0	0	-13/2	0	0	-11/2	-3	0	
		$\bar{a}_1$	1	0	0	0	0	1/2	0	0	-3/2	-1/2	0	
		$\bar{a}_2$	1	0	0	0	0	-1/4	0	0	13/4	7/4	0	
		$\bar{a}_3$	0	0	0	0	0	-1/2	0	0	-3/2	-1/2	0	
		$\bar{a}_4$	0	0	0	0	0	-1/2	0	0	-3/2	-1/2	0	

Table 7

This table gives $y^{*6} = (0, 2, 0, \frac{1}{2})$ as an optimal solution and $f(y^{*6}) = -\dfrac{31}{7}$. Adding constraint (6) to table 2 we get the following table 8:

d_B	c_B													x_B
		d_j	0	1	1	1	0	0	0	0	0	0	0	
		c_j	-1	-1	2	3	0	0	0	0	0	0	0	
d_B	c_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}	y_{11}	x_B
0	0	y_7	1	0	0	-1	0	-3	1	0	0	-2	0	3
2	1	y_3	0	0	1	0	0	0	0	0	0	-1	0	1
1	-1	y_2	0	1	0	1	0	4	0	0	0	3	0	1
0	0	y_5	3	0	0	2	1	9	0	0	0	7	0	4
1	3	y_8	0	0	0	-2	0	-8	0	1	0	-5	0	3
1	3	y_9	0	0	0	-1	0	-4	0	0	1	-3	0	1
0	0	y_{11}	1	0	0	4	0	8	0	0	0	3	1	1
	$z_1 = -14$	$z_j^1 - c_j$	-1	0	0	-1	0	7	0	0	0	-5	0	
	$z_2 = 3$	$z_j^2 - d_j$	0	0	0	1	0	3	0	0	0	2	0	
		$\triangle_j$	-3	0	0	-11	0	-21	0	0	0	-13	0	
		$\bar{a_1}$	1	0	0	-1	0	-6	0	0	0	-4	0	
		$\bar{a_2}$	1	0	0	3	0	13	0	0	0	8	0	
		$\bar{a_3}$	0	0	0	-2	0	-6	0	0	0	-4	0	
		$\bar{a_4}$	0	0	0	1	0	6	0	0	0	5	0	

Table 8

This table gives $y^{*7} = (0, 1, 1, 0)$ as an optimal solution and $f(y^{*7}) = -\dfrac{14}{3}$. From above tables we get $f(y^{*6}) > f(y^{*7})$, therefore, we will explore C_5 node first but the solution obtained by adding C_5 is not an integer therefore again we apply branching procedure and we get the following constraints:

$$C_7 : y_4 \leq \lfloor \frac{1}{2} \rfloor \qquad (7)$$

$$C_8 : y_4 \geq \lceil \frac{1}{2} \rceil \qquad (8)$$

When these constraints are added to table 7 the problem becomes unfeasible, hence in both the cases the node is fathomed.

Now, we will explore C_6, adding C_6 to table 2 we get $y^{*7} = (0, 1, 1, 0)$ as an optimal solution which is integer. We will test its efficiency now.

Efficiency test for $y^{*7} = (0, 1, 1, 0)$: To check the efficiency of $y^{*7} = (0, 1, 1, 0)$ we have to solve the following problem:

$$\text{EK}(y^{*(l)}) \qquad \max \quad \theta = w_1 + w_2 + w_3 + w_4$$

s.t.

$$3y_1 - 2y_2 + y_3 + x_4 \leq 3$$

$$y_2 + 3y_3 + 4y_4 \leq 4$$

$$y_1 + y_2 + y_3 + y_4 \leq 5$$

$$2y_2 + y_3 \leq 6$$

$$y_1 + y_2 - y_3 - 2y_4 - w_1 = 0$$

$$y_1 - 3y_2 - y_3 + y_4 - w_2 = -4$$

$$0y_1 + 2y_2 + 2y_3 + 2y_4 - w_3 = 4$$

$$0y_1 - y_2 + 2y_3 + 2y_4 - w_4 = 1$$

$$y_1, y_2, y_3, y_4 \in N$$

$$y_1, y_2, y_3, y_4, w_1, w_2, w_3, w_4 \geq 0$$

After solving this problem using TORA we get the following result:

subproblem	objective value, θ	y_1	y_2	y_3	y_4	w_1	w_2	w_3	w_4
2	2	1	1	1	0	1	1	0	0

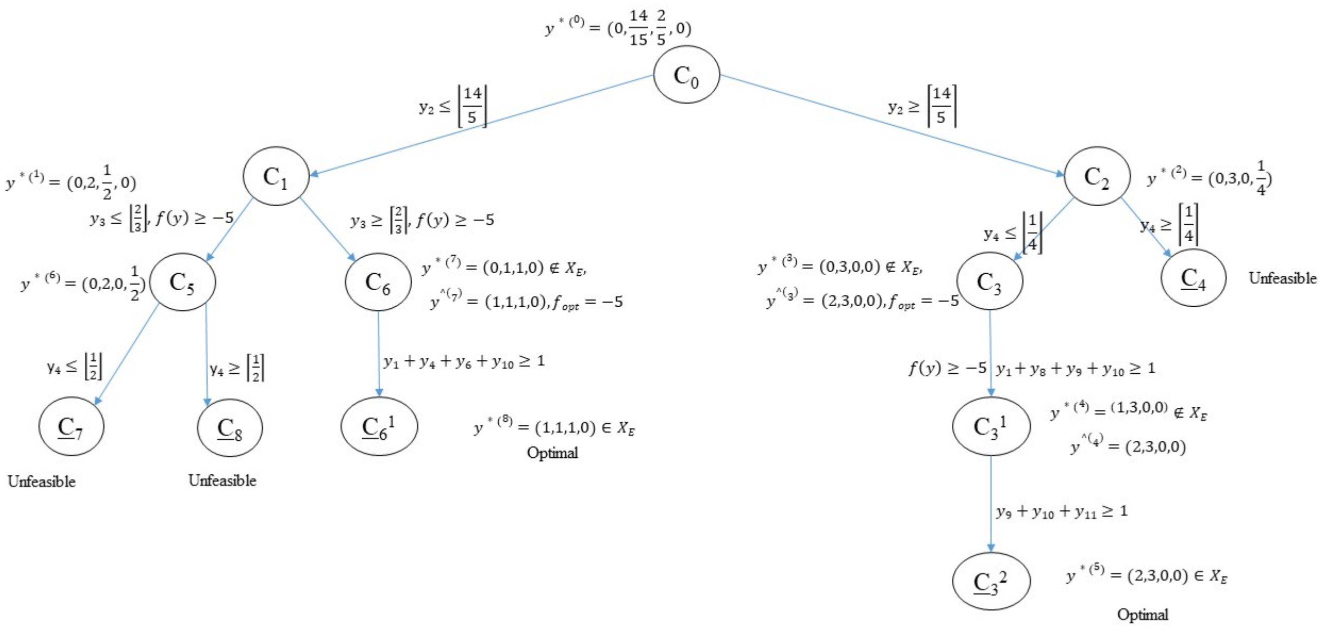
The above table shows objective function value is not equal to zero. Hence, the optimal solution $y^{*7} = (0,1,1,0)$ is not efficient. After applying efficiency test we get an efficient integer solution i.e. $\hat{y}^7$= (1,1,1,0) with $f(\hat{y}^7) = -5$. Update y_{opt}= (1,1,1,0) and $f_{opt} = -5$. To obtain the optimal solution we will add cuts to table 8. From table 8 we obtain $S_3 = \{1,4,6,10\}$. Apply the efficient cut $y_1 + y_4 + y_6 + y_{10} \geq 1$. After adding the cut in table 8 and applying dual simplex method we obtain table 9:

		d_j	0	1	1	1	0	0	0	0	0	0	0	0	0	
		c_j	-1	-1	2	3	0	0	0	0	0	0	0	0	0	
d_B	c_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}	y_{11}	y_{12}	y_{13}	x_B
0	0	y_7	0	0	0	-4	0	-2	1	0	0	-3	0	1	0	2
2	1	y_3	0	0	1	0	0	0	0	0	0	-1	0	0	0	1
1	-1	y_2	0	1	0	4	0	1	0	0	0	3	0	0	0	1
0	0	y_5	0	0	0	6	1	-1	0	0	0	4	0	3	0	1
1	3	y_8	0	0	0	-8	0	-8	0	1	0	-5	0	0	0	3
1	3	y_9	0	0	0	-4	0	-4	0	0	1	-3	0	0	0	1
0	0	y_{11}	0	0	0	7	0	8	0	0	0	4	1	1	0	0
0	-1	y_1	1	0	0	1	0	8	0	0	0	1	0	-1	0	1
0	0	y_{13}	0	0	0	7	0	8	0	0	0	4	0	1	1	0
	$z_1 = -15$	$z_j^1 - c_j$	0	0	0	-8	0	-2	0	0	0	-6	0	1	0	
	$z_2 = 3$	$z_j^2 - d_j$	0	0	0	3	0	1	0	0	0	2	0	0	0	
		$\triangle_j$	0	0	0	-21	0	-9	0	0	0	-12	0	-3	0	
		$\bar{a}_1$	0	0	0	-7	0	-2	0	0	0	-5	0	1	0	
		$\bar{a}_2$	0	0	0	12	0	2	0	0	0	7	0	1	0	
		$\bar{a}_3$	0	0	0	-6	0	-2	0	0	0	-4	0	0	0	
		$\bar{a}_4$	0	0	0	6	0	1	0	0	0	5	0	0	0	

Table 9

This table gives $y^{*8} = (1, 1, 1, 0)$ as an optimal solution and $f(y^{*8}) = -5$. The solution obtained by this table is integer efficient. Hence, the node is fathomed.

Now, the algorithm will stop since all the formed nodes are fathomed and we obtained two integer efficient solution for $(ILFP)_E$ are $x_{opt_1} = (2, 3, 0, 0)$ and $x_{opt_2} = (1, 1, 0, 0)$ with optimal value $f_{opt} = -5$. The application of Branch and Bound method is shown by following diagram:



3.6 Conclusion

In this chapter, we obtained integer efficient solutions of $(ILFP)_E$ over the efficient set of (MOLIP) using an algorithm. The algorithm generates efficient points without enumerating the set of all the efficient points. In fact, we use branch and bound method which is further strengthened by adding efficient cuts and efficiency tests, by this we fathomed number of nodes. Hence, many non-efficient integer feasible solutions can be avoided.

Bibliography

[1] Aneja, YP & Nair, KPK 1979, "Bicriteria transportation problem." Management Science, vol. 25, no. 1, pp. 73-78.

[2] Ban, VT 1983, "A finite algorithm for minimizing a concave function under linear constraints and its applications." In Proceedings of IFIP Working Conference on Recent Advances on System Modelling and Optimization.

[3] Benson, HP 1997, "Generating the efficient outcome set in multiple-objective linear programs: the bicriteria case." Acta Mathematica Vietnamica, vol. 22, pp. 29-51.

[4] Benson, HP 1998a, "An outer approximation algorithm for generating all efficient extreme points in the outcome set of a multiple objective linear programming problem." Journal of Global Optimization, vol. 13, no. 1, pp. 1-24.

[5] Benson, HP 1998b, "Hybrid approach for solving multiple-objective linear programs in outcome space." Journal of Optimization Theory and Applications, vol. 98, no. 1, pp. 17-35.

[6] Benson, HP 1998c, "Further analysis of an outcome set-based algorithm for multiple-objective linear programming." Journal of Optimization Theory and Applications, vol. 97, no. 1, pp. 1-10.

[7] Benson, HP & Sun, E 2000, "Outcome space partition of the weight set in multiobjective linear programming." Journal of Optimization Theory and Applications, vol. 105, no. 1, pp. 17-36.

[8] Benson, HP & Sun, E 2002, "A weight set decomposition algorithm for finding all efficient extreme points in the outcome set of a multiple objective linear program." European Journal of Operational Research, vol. 139, no. 1, pp. 26-41.

[9] Cohon, JL 1978, "Multiobjective programming and planning." Academic, New York.

[10] Dhaenens, C, Lemesre, J & Talbi, EG 2010, "K-PPM: A new exact method to solve multi-objective combinatorial optimization problems." European Journal of Operational Research, vol. 200, no. 1, pp. 45-53.

[11] Ecker, J, & Kouada, I, 1975 Finding efficient points for linear multiple objective programs. Mathematical Programming, no. 8, pp. 375377.

[12] Ida, M 2005, "Efficient solution generation for multiple objective linear programming based on extreme ray generation method." European Journal of Operational Research, vol. 160, no. 1, pp. 242-251.

[13] Jozefowiez, N, Laporte, G & Semet, F 2012,"A generic branch-and-cut algorithm for multiobjective optimization problems: Application to the multilabel traveling salesman problem." INFORMS Journal on Computing, vol. 24, no. 4, pp. 554-564.

[14] Klein, D & Hannan, E 1982 "An algorithm for the multiple objective integer linear programming problem." European Journal of Operational Research, vol. 9, no. 4, pp. 378-385.

[15] Krichen, S, Masri, H & Guitouni, A 2012, "Adjacency based method for generating maximal efficient faces in multiobjective linear programming." Applied Mathematical Modelling, vol. 36, no. 12, pp. 6301-6311.

[16] Kuhn, HW & Tucker, AW 1951, "Proceedings of 2nd Berkeley Symposium.", pp. 481-492.

[17] Lemesre, J, Dhaenens, C & Talbi, EG 2007, "Parallel partitioning method (PPM): A new exact method to solve bi-objective problems." Computers & operations research, vol. 34, no. 8, pp. 2450-2462.

[18] Lokman, Banu, and Murat Kksalan. "Finding all nondominated points of multi-objective integer programs." Journal of Global Optimization, vol. 57, no. 2 (2013): pp. 347-365.

[19] Mavrotas, G & Diakoulaki, D 1998, "A branch and bound algorithm for mixed zero-one multiple objective linear programming." European Journal of Operational Research, vol. 107, no. 3, pp. 530-541.

[20] Neumayer, P, Schweigert, D, 1994 Three algorithms for bicriteria integer linear programs. OR Spektrum, no. 16, pp. 267-276.

[21] Ozpeynirci, & Koksalan, M 2010, "An exact algorithm for finding extreme supported nondominated points of multiobjective mixed integer programs." Management Science, vol. 56, no. 12, pp. 2302-2315.

[22] Przybylski, A, Gandibleux, X & Ehrgott, M 2010b, "A two phase method for multi-objective integer programming and its application to the assignment problem with three objectives." Discrete Optimization, vol. 7, no. 3, pp. 149-165.

[23] Stidsen, T, Andersen, KA & Dammann, B 2014, "A Branch and Bound Algorithm for a Class of Biobjective Mixed Integer Programs." Management Science.

[24] Sylva, J & Crema, A 2004, "A method for finding the set of non-dominated vectors for multiple objective integer linear programs."European Journal of Operational Research, vol. 158, no. 1, pp. 46-55.

[25] Sylva, J & Crema, A 2007, "A method for finding well-dispersed subsets of non-dominated vectors for multiple objective mixed integer linear programs." European journal of operational research, vol. 180, no. 3, pp. 1011-1027.

[26] Tuy, H & Horst, R 1988, "Convergence and restart in branch-and-bound algorithms for global optimization. Application to concave minimization and dc optimization problems." Mathematical Programming, vol. 41, no. 1-3, pp. 161-183.

[27] Wassila Drici, Fatma Zohra Ouail, Mustapha Moulai1, 2017, Optimizing a linear fractional function over the integer efficient set. Ann Oper Res, DOI 10.1007/s10479-017-2691-0.

[28] Yamamoto, Y, 2002, "Optimization over the efficient set: Overview." Journal of Global Optimization, vol. 22, pp. 285-317.

[29] Zeleny, M, 1974 Linear multiobjective programming. Springer. New York Berlin Heidelberg.

[30] Zerdani, O, Moulai, M, "Optimization over an integer efficient set of a multiple objective linear fractional problem." Applied Mathematical Sciences, vol. 5, pp. 2451-2466.