Journal of Energy Technology

JET Volume 16 (2023) p.p. 23-40
Issue 1, 2023
Type of article: 1.01
http://www.fe.um.si/si/jet.htm

GENERALISED FUZZY LINEAR PROGRAMMING

GENERALIZIRANO MEHKO LINEARNO PROGRAMIRANJE

Janez Usenik[ℜ], Maja Žulj[1]

Keywords: linear programming, fuzzy linear programming, generalised linear programming, generalised fuzzy linear programming

Abstract

Linear programming is one of the widely used methods for optimising business systems, which includes organisational, financial, logistic and control subsystems of energy systems in general. It is possible to express numerous real-world problems in a form of linear program and then solve by simplex method *[1]*. In the development of linear programming, we are facing a number of upgrades and generalisations, as well as replenishment. Particularly interesting in recent years is an option that decision variables and coefficients are fuzzy numbers. In this case we are dealing with fuzzy linear programming. If we also include in a fuzzy linear program a generalisation with respect to Wolfe's modified simplex method *[1]*, we obtain a generalised fuzzy linear program (GFLP). Usenik and Žulj introduced methods for solving those programs and proved the existence of the optimal solution in *[2]*. In the article, the simplex algorithm which enables the determining of an optimal solution for GFLP is described. There is a numerical example at the end of the article that illustrates the algorithm.

Povzetek

Linearno programiranje je najbolj uporabljena metoda optimizacije poslovnih sistemov, med katere štejemo tudi organizacijske, finančne, logistične in nasploh upravljalne podsisteme energetskega sistema. Veliko praktičnih problemov je mogoče izraziti v obliki linearnega programa, ki ga nato rešimo s simpleksno metodo *[1]*. Razvoj linearnega programiranja je doživel vrsto nadgradenj, posplošitev in dopolnitev. V zadnjih letih je še posebej zanimiva možnost, da so odločitvene spremenljivke in koeficienti mehka števila – v tem primeru gre za mehko linearno programiranje. Ko pa v ta program uvedemo še pojem generalizacije v Wolfejev pomenu *[1]*, govorimo o generaliziranem mehkem linearnem programiranju (GMLP). Usenik in Žulj *[2]* sta razvila postopke reševanja takšnih programov in dokazala eksistenco optimalne rešitve. V članku opišemo algoritem simpleksnega postopka za GMLP, ki omogoča izračun optimalne rešitve, in na koncu dodamo numerični primer, ki ilustrira izvedeni algoritem.

ℜ Corresponding author: Prof. Janez Usenik, PhD, Tel.: +38640 647 686, E-mail address: janez.usenik@guest.um.si

1 University of Maribor, Faculty of Energy Technology, Hočevarjev trg 1, 8270 Krško, Slovenia, E-mail address: maja.zulj@um.si

1 INTRODUCTION

Linear programming is one of the most frequently used techniques in operations research, introduced in 1939 by Russian mathematician Leonid Vitalijevič Kantorovič, who also proposed a method for solving it. Between 1946 and 1947, American mathematician Georg Bernard Dantzig defined a general formulation of linear programming. In 1947, he introduced the so-called simplex method, a method that enables the successful solving of any linear programming problem [1]. Linear programming can be used in economic science and in the management of business or organisational systems, as well as in actions like production planning, the optimisation of the technological process, an optimal logistics service, optimal outsourcing, etc. Linear programming proved to be of considerable applicative importance at the time computers became more capable. Nowadays, a variety of competent computer software programs exist that even enable problems of enormous dimensions to be easily solved.

The theory of linear programming is developing in different ways. Let us point out two alternatives in this field. In the first alternative we use a dynamic approach, where we study the optimal behaviour of variables, which are functions of time in this approach. This problem is called continuous variable dynamic linear programming *[3], [4].* In *[5]* we can find a generalisation of c/b/A-continuous variable dynamic linear programming in the sense of Wolfe generalisation *[1].*

This generalisation is based on the condition that elements of some columns of matrix $A(t)$, at the time of formulating the problem, are unknown, but linked convexly in columns. In this case, we talk about *generalised continuous variable dynamic linear programming.*

The second alternative, which has attracted a lot of interest, especially in the last 20 years, is linear programming in conditions represented by fuzzy logic. In this matter, we are dealing with *fuzzy linear programming.* It is a tool for modelling imprecise data and it is based on fuzzy sets *[6].* In 1978, Zimmermann proposed the formulation of fuzzy linear programming problems in [7]. Since then, researchers have developed a relatively large number of different methods to solve such problems. It also turns out that there are no obstacles for generalisation in fuzzy linear programming. In this case, we talk about *generalised fuzzy linear programming.* Usenik and Žulj introduced methods for solving those programs and proved the existence of the optimal solution in *[2].*

2 FUZZY LINEAR PROGRAMMING

2.1 Fuzzy numbers

A fuzzy number is a generalisation of a regular real number in the sense that it does not refer to one single value but rather to a connected set of possible values. It is a special case of fuzzy sets, which were introduced by Lotfi A. Zadeh in *[6].* A fuzzy set $\tilde{A}$ is a set of ordered pairs: $\tilde{A}=\{(x,\mu_{\tilde{A}}(x))\,|\,x\in R\}$ where $\mu_{\tilde{A}}(x)$ is the membership function of x, which maps R to a subset of the non-negative real numbers whose supremum is finite. If $\sup_x \mu_{\tilde{A}}(x)=1$, the fuzzy set $\tilde{A}$ is called normal and if $\mu_{\tilde{A}}(tx+(1-t)y)\geq\min\{\mu_{\tilde{A}}(x),\mu_{\tilde{A}}(y)\}$, $x,y\in R$, $t\in[0,1]$, the fuzzy set $\tilde{A}$ is convex. Fuzzy number $\tilde{A}$ is a convex normalised fuzzy set $\tilde{A}$ on the real line, such that there exists at least one $x_0\in R$ with

$\mu_{\tilde{A}}(x_0)=1$ and where $\mu_{\tilde{A}}(x)$ is piecewise continuous, *[7]*. In the numerical example at the end of the article are triangular fuzzy numbers, represented with tree points: $\tilde{A}=(a,b,c)$, with membership function:

$$\mu_{\tilde{A}}=\begin{cases} \frac{1}{b-a}x-\frac{a}{b-a} & \text{for } a\le x\le b \\ -\frac{1}{c-b}x+\frac{c}{c-b} & \text{for } b\le x\le c \\ 0 & \text{otherwise} \end{cases}$$

Let $F(R)$ be the set of triangular fuzzy numbers, and $\tilde{A},\tilde{B}\in F(R)$ are given as $\tilde{A}=(a_1,a_2,a_3)$, $\tilde{B}=(b_1,b_2,b_3)$, $\lambda\in R$. Then operations of triangular fuzzy numbers are defined as
$\tilde{A}+\tilde{B}=(a_1+b_1,a_2+b_2,a_3+b_3)$; $\tilde{A}-\tilde{B}=(a_1-b_3,a_2-b_2,a_3-b_1)$;
$\lambda\tilde{A}=(\lambda a_1,\lambda a_2,\lambda a_3)\quad for\quad \lambda\ge 0$ and $\lambda\tilde{A}=(\lambda a_1,\lambda a_2,\lambda a_3)\quad for\quad \lambda<0$.

The nature of simplex algorithms in linear programming problems involves comparison, or so-called ranking, of fuzzy numbers. There is no generally accepted criteria for comparison of fuzzy numbers. One convenient method for comparison is the method of ranking functions. Ranking function $\mathbf{R}:F(R)\to R$ is a mapping from the set of fuzzy quantities into the set of real numbers. Fuzzy numbers can then be compared according to the corresponding real numbers. Several number-ranking functions can be found in the literature. For ranking triangular fuzzy numbers in the numerical example at the end of this article, the Chens method *[8]* is used.

2.2 Fuzzy linear programming

A general model of a linear programming problem can be written as:

$$\begin{aligned} &\text{maximize} && cx \\ &\text{subject to} && Ax\le b \\ & && x\ge 0 \end{aligned} \tag{2.1}$$

If some of the coefficients (c,A,b) in a linear program (2.1) cannot be precisely defined, we can treat them as fuzzy numbers and call this linear program a fuzzy linear program. Based on the place where fuzzy numbers appear, we can divide fuzzy linear programming (FLP) problems into four main categories:

- FLP with fuzzy resources ($\tilde{b}$);
- FLP with fuzzy coefficients in an objective function ($\tilde{c}$);
- FLP with fuzzy technological coefficients ($\tilde{a}_{ij}$);
- FLP with fuzzy variables ($\tilde{x}$).

By combining these four categories, we obtain many different types of fuzzy linear programming problems, each of them solvable using a different method. Around 50 methods have been analysed and a basis for a taxonomy of different methods has been published in *[9].*

The fuzzy linear program for $c\in R^n, \tilde{x}\in[F(R)]^n, \tilde{b}\in[F(R)]^m, A\in R^{m\times n}$ is in canonical form (completed with slack and artificial variables), defined by expressions:

$$\begin{aligned} & opt \ \ \tilde{z} = (c, \tilde{x}) \\ \text{subject to} \ \ & A\tilde{x} = \tilde{b} \\ & \tilde{x} \geq 0 \end{aligned} \tag{2.2}$$

where fuzzy quantities are compared according to a selected ranking method. Fuzzy vector $\tilde{x} \in (F(R))^n$ is a feasible solution for the problem (2.2) if and only if $\tilde{x}$ satisfies the constraints of the problem. A fuzzy feasible solution $\tilde{x}_{opt}$ is a fuzzy optimal solution for problem (2.2), if for all fuzzy feasible solutions $\tilde{x}$, we have $(c, \tilde{x}_{opt}) \geq (c, \tilde{x})$ when optimum means maximum, or $(c, \tilde{x}_{opt}) \leq (c, \tilde{x})$ when optimum means minimum. There are several methods to solve fuzzy linear programs, many of them analysed in *[2].*

3 GENERALISED FUZZY LINEAR PROGRAMMING

In the article we are dealing with the generalisation of fuzzy linear programming in accordance with the Wolfe approach *[1], [3], [5]*. Therefore, we tackle the problem concerning a linear program where technological coefficients and coefficients in an objective function are not precisely known. But we know that these coefficients are integrated in some known convex composition, i.e., the limited material, financial, logistic, energy, ecological or human resource that is required to produce some product, service, or similar.

Consider the fully fuzzy linear program in standard and canonical form:

$$\begin{aligned} & opt \, z = (\tilde{c}, \tilde{x}) \\ & \tilde{A}\tilde{x} = \tilde{b} \\ & \tilde{x} \geq \tilde{0} \end{aligned} \tag{3.01}$$

where:

$\tilde{c} = (\tilde{c}_1, \tilde{c}_2, \ldots, \tilde{c}_n)$, $\tilde{c}_i \geq \tilde{0}, i = 1, 2, \ldots, n$; a vector with either all, some or no components being fuzzy numbers;

$\tilde{b} = (\tilde{b}_1, \tilde{b}_2, \ldots, \tilde{b}_m)$, $\tilde{b}_j > \tilde{0}, j = 1, 2, \ldots, m$; a vector with either all, some or no components being fuzzy numbers;

$\tilde{x} = (\tilde{x}_1, \tilde{x}_2, \ldots \tilde{x}_m)$, $\tilde{x}_i \geq \tilde{0}, i = 1, 2, \ldots, n$; a vector with either all, some or no components being fuzzy numbers;

$\tilde{A} = [\tilde{a}_{ij}]_{m \times n}$ matrix $m \times n, \tilde{a}_{ij} \in R, i = 1, 2, \ldots, n; j = 1, 2, \ldots, m$; all the elements of the matrix can be fuzzy numbers, or just some or none of the elements are fuzzy numbers.

Now let us change the fuzzy linear program (3.01) in such a way that in matrix $\tilde{A}$ part of the elements in some columns are unknown, but there is a demand that they are restricted with some given convex constraint. We split the matrix:

$$\tilde{A}=\begin{bmatrix}\tilde{a}_{11} & \tilde{a}_{12} & \cdots & \tilde{a}_{1n}\\ \tilde{a}_{21} & \tilde{a}_{22} & \cdots & \tilde{a}_{2n}\\ \vdots & \vdots & \cdots & \vdots\\ \tilde{a}_{m1} & \tilde{a}_{m2} & \cdots & \tilde{a}_{mn}\end{bmatrix}$$

in accordance with the columns into two parts:

$$\tilde{A}=\begin{bmatrix}\tilde{a}_{11} & \tilde{a}_{12} & \cdots & \tilde{a}_{1\mu} & \tilde{a}'_{1,\mu+1} & \tilde{a}'_{1,\mu+2} & \cdots & \tilde{a}'_{1n}\\ \tilde{a}_{21} & \tilde{a}_{22} & \cdots & \tilde{a}_{2\mu} & \tilde{a}'_{2,\mu+1} & \tilde{a}'_{2,\mu+2} & \cdots & \tilde{a}'_{2n}\\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots\\ \tilde{a}_{m1} & \tilde{a}_{m2} & \cdots & \tilde{a}_{m\mu} & \tilde{a}'_{m,\mu+1} & \tilde{a}'_{m,\mu'2} & \cdots & \tilde{a}'_{mn}\end{bmatrix}=\left[\tilde{A}_0 \;\vdots\; \tilde{A}'\right]_{m\times n}$$

where:

$$\tilde{A}_0=\begin{bmatrix}\tilde{a}_{11} & \tilde{a}_{12} & \cdots & \tilde{a}_{1\mu}\\ \tilde{a}_{21} & \tilde{a}_{22} & \cdots & \tilde{a}_{2\mu}\\ \vdots & \vdots & \cdots & \vdots\\ \tilde{a}_{m1} & \tilde{a}_{m2} & \cdots & \tilde{a}_{m\mu}\end{bmatrix}=\left[\tilde{P}_1 \;\; \tilde{P}_2 \;\; \cdots \;\; \tilde{P}_\mu\right]$$

is a $m \times \mu$ matrix with the elements that are fuzzy numbers. Now denote with $\tilde{P}_j,\;\; j=1,2,\ldots,\mu$ the columns of matrix $\tilde{A}_0$.

In the matrix:

$$\tilde{A}'=\begin{bmatrix}\tilde{a}'_{1,\mu+1} & \tilde{a}'_{1,\mu+2} & \cdots & \tilde{a}'_{1n}\\ \tilde{a}'_{2,\mu+1} & \tilde{a}'_{2,\mu+2} & \cdots & \tilde{a}'_{2n}\\ \vdots & \vdots & \cdots & \vdots\\ \tilde{a}'_{m,\mu+1} & \tilde{a}'_{m,\mu'2} & \cdots & \tilde{a}'_{mn}\end{bmatrix}=\left[\tilde{P}_{\mu+1} \;\; \tilde{P}_{\mu+2} \;\; \cdots \;\; \tilde{P}_n\right]$$

the elements (in some columns) are currently unknown, but as well as coefficients in an objective function, they are bounded with some determined convex constraint.

Denote $\tilde{c}_j=\tilde{a}_{m+1}$, $j=1,2,\ldots,n$, and if we insert this into the row $(m+1)$ of matrix $\tilde{A}$ we obtain extended matrix $\hat{\tilde{A}}$:

$$\hat{\tilde{A}}=\begin{bmatrix}\tilde{a}_{11} & \tilde{a}_{12} & \cdots & \tilde{a}_{1\mu} & \tilde{a}'_{1,\mu+1} & \tilde{a}'_{1,\mu+2} & \cdots & \tilde{a}'_{1n}\\ \tilde{a}_{21} & \tilde{a}_{22} & \cdots & \tilde{a}_{2\mu} & \tilde{a}'_{2,\mu+1} & \tilde{a}'_{2,\mu+2} & \cdots & \tilde{a}'_{2n}\\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots\\ \tilde{a}_{m1} & \tilde{a}_{m2} & \cdots & \tilde{a}_{m\mu} & \tilde{a}'_{m,\mu+1} & \tilde{a}'_{m,\mu'2} & \cdots & \tilde{a}'_{mn}\\ \tilde{a}_{m+1,1} & \tilde{a}_{m+1,2} & \cdots & \tilde{a}_{m+1,\mu} & \tilde{a}'_{m+1,\mu+1} & \tilde{a}'_{m+1,\mu'2} & \cdots & \tilde{a}'_{m+1,n}\end{bmatrix}=\left[\tilde{A}_0 \;\vdots\; \tilde{A}'\right]_{(m+1)\times n}$$

The columns:

$$\hat{\tilde{P}}_j = \begin{bmatrix} \tilde{a}'_{1j} \\ \tilde{a}'_{2j} \\ \vdots \\ \tilde{a}'_{mj} \\ \tilde{a}'_{m+1,j} \end{bmatrix}, j = \mu+1, \mu+2, \ldots, n; \;\; \tilde{a}'_{m+1,j} = \tilde{c}_j$$

are restricted with some determined convex composition $\hat{\tilde{P}}_j \in \tilde{C}_j$, $j = \mu+1, \mu+2, \ldots, n$, where $\tilde{C}_j$ is a fuzzy convex polyhedron, determined by given constraints. Thus, we obtain the generalised fuzzy linear program *[2]*:

$$\begin{aligned} & opt(\tilde{c}, \tilde{x}) \\ & \tilde{A}\tilde{x} = \tilde{b} \\ & \tilde{A} = \left[\tilde{A}_0 \;\vdots\; \tilde{A}' \right] \\ & \hat{\tilde{P}} \in \tilde{C}_j, \; j = \mu+1, \mu+2, \ldots, n \\ & \tilde{x} \geq \tilde{0} \end{aligned} \tag{3.02}$$

In the generalised fuzzy linear program (GFLP), we have to define the optimum of the objective function and simultaneously also compute the optimal structure of varying convexly linked columns $\hat{\tilde{P}}_j$, $j = \mu+1, \mu+2, \ldots, n$.

There are two phases in the procedure to solving the generalised linear programming problems *[1], [2], [5]*. The first phase means solving the program (3.02) without varying convexly linked columns, therefore in the first phase we set $\tilde{A}' \equiv O$ and thus obtain a standard fuzzy linear program, which can be solved using the simplex algorithm. In the second phase in each iteration, we add to the obtained optimal solution from the first phase additional conditions of varying convexly linked columns, and thus we obtain a generalisation of the original program using Wolfe's modified simplex method.

3.1 Main program

In program (3.02), we seek the optimal solution that can be either minimum or maximum. From the theory of linear programming, it is known that the procedure is the same in both cases.

In (3.02) we extend matrix $\tilde{A} = \left[\tilde{A}_0 \;\vdots\; \tilde{A}' \right]$ with row $(m+1)$ and denote this matrix with $\tilde{A} = \left[\hat{\tilde{A}}_0 \;\vdots\; \hat{\tilde{A}}' \right]$. Also, we introduce vectors P_0 and $\hat{\tilde{b}}$:

$$P_0 = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}_{(m+1)\times 1} \qquad \hat{\tilde{b}} = \begin{bmatrix} \tilde{b} \\ 0 \end{bmatrix}_{(m+1)\times 1}$$

Now we denote $\tilde{z} = (\tilde{c}, \tilde{x}) = \tilde{z}_0$ and rewrite program (3.02) in this form:

$$\begin{aligned} &\max \tilde{x}_0 \\ &P_0 \tilde{x}_0 + \hat{\tilde{A}} \tilde{x} = \hat{\tilde{b}} \\ &\hat{\tilde{A}} = \left[\hat{\tilde{A}}_0 \ \vdots \ \hat{\tilde{A}}' \right]_{(m+1)\times n} \\ &\hat{\tilde{P}} \in \tilde{C}_j,\ j = \mu+1, \mu+2, \ldots, n \\ &\tilde{x} \geq \tilde{0} \end{aligned} \tag{3.03}$$

Definition: *Linear program of form (3.03) is called a* ***main program.***

Since we are using Dantzig simplex algorithm to solve this program, in each step, the row $(m+1)$ in inverse of the basis $P^{(k)}$ is a row of simplex multipliers:

$$\begin{aligned} \tilde{\pi}^{(k)} &= \left[-\tilde{\pi}_1 \ \ -\tilde{\pi}_2 \ \ \cdots \ \ -\tilde{\pi}_m \right] \\ \hat{\tilde{\pi}}^{(k)} &= \left[-\tilde{\pi}_1 \ \ -\tilde{\pi}_2 \ \ \cdots \ \ -\tilde{\pi}_m \ ; \ 1 \right] \end{aligned}$$

These simplex multipliers can be in general fuzzy numbers. Moreover, consider also:

$$\begin{aligned} &\hat{\tilde{\pi}}^{(k)} \cdot \hat{\tilde{P}}_j \equiv 0 && j - \text{ basic index} \\ &\hat{\tilde{\pi}}^{(k)} \cdot \hat{\tilde{P}}_j \overset{>}{\underset{<}{=}} 0 && j - \text{nonbasic index} \\ &\hat{\tilde{\pi}}^{(k)} \cdot P_0 = 1 \end{aligned} \tag{3.04}$$

3.2 Initial feasible solution

Solving of the generalised fuzzy linear program is based on Dantzig simplex algorithm and Wolfe's modified simplex method. Thus, we solve the problem in two phases. In the first phase we look for an optimal feasible basic solution to problem (3.03) without varying vectors $\hat{\tilde{P}}_j,\ j = \mu+1, \mu+2, \ldots, n$, i.e., a standard fuzzy linear program, which can be solved using the

Dantzig simplex algorithm. Solution $\tilde{x}^{(k_0)}$ is a vector with components that can be either fuzzy numbers or, in some cases, also regular real numbers. This depends on the type of fuzzy linear program that we are dealing with. The obtained solution $\tilde{x}^{(k_0)}$ is just a feasible basic solution for the main program (3.03) and in general not necessarily optimal. It is a starting point for computations in the second phase where we are seeking the solution of the main program (3.03). In the second phase we want to improve the solution obtained in the first phase. Into the basis we gradually bring vectors $\hat{\tilde{P}}_j$ considering convex constraints, i.e., $\hat{\tilde{P}}_j \in \tilde{C}_j$ for all $j = \mu+1, \mu+2, \ldots, n$.

3.3 First feasible solution

Let $\tilde{x}^{(k_0)}$ be the optimal solution of the first phase, which is for the second phase just the initial feasible basic solution, denoted by ${}^{(2)}\tilde{x}^{(0)}$. From the final simplex table of the first phase, we discern the row of simplex multipliers:

$$\hat{\tilde{\pi}}^{(0)} = \left[-\tilde{\pi}_1^{(0)} \ -\tilde{\pi}_2^{(0)} \ \cdots \ -\tilde{\pi}_m^{(0)} \ ; \ 1 \right]$$

After multiplying conditions of the main program (3.03) with these multipliers, we obtain

$$\hat{\tilde{\pi}}^{(0)} P_0 \tilde{x}_0 + \hat{\tilde{\pi}}^{(0)} \hat{\tilde{A}} x = \hat{\tilde{\pi}}^{(0)} \hat{\tilde{b}}$$

According to (3.04) we have:

$$\tilde{x}_0 + \sum_{j-nonbasic} \hat{\tilde{\pi}}^{(0)} \hat{\tilde{P}}_j x_j = \hat{\tilde{\pi}}^{(0)} \hat{\tilde{b}}$$

In the case where the optimum is a maximum and the value of function $\tilde{x}_0$ is optimal also for problem (3.03), there is $\hat{\tilde{\pi}}^{(0)} \hat{\tilde{P}}_j > 0$ for all non-basic indexes *j* and the problem is solved. If this is not the case, we have:

$$\begin{aligned} &\hat{\tilde{\pi}}^{(k)} \hat{\tilde{P}}_j < 0, \ j \in I_1 \\ &I_1 - \ a \ set \ of \ nonbasic \ indexes \\ &I_1 \subseteq \{1, 2, \ldots, n\} \end{aligned}$$

Since $\tilde{x}_0 = \hat{\tilde{\pi}}^{(0)} \hat{\tilde{b}} - \sum_{j-nonbasic} \hat{\tilde{\pi}}^{(0)} \hat{\tilde{P}}_j x_j$ the value of $\tilde{x}_0$ would be bigger if we find:

$$\min_j \hat{\tilde{\pi}}^{(0)} \hat{\tilde{P}}_j, \quad \hat{\tilde{P}}_j \in \tilde{C}_j \tag{3.05}$$

for all $j = \mu+1, \mu+2, \ldots, n$.

Problem (3.05) is called a ***system of first subprograms*** of the main program (3.03). For each unknown column we obtain one first subprogram, thus there will be as many first subprograms as there are unknown columns.

Allow the system of first subprograms to yield optimal solutions: $\hat{\tilde{\pi}}^{(0)}\hat{\tilde{P}}^{(1)}_{\mu+1}, \hat{\tilde{\pi}}^{(0)}\hat{\tilde{P}}^{(1)}_{\mu+2}, \ldots, \hat{\tilde{\pi}}^{(0)}\hat{\tilde{P}}^{(1)}_{n}$. In the case of real values, we choose the smallest (meaning we choose the value with the biggest absolute value). Assume that this is a solution with index λ_1 :

$$\min_j \hat{\tilde{\pi}}^{(0)}\hat{\tilde{P}}^{(1)}_{\mu+1} = \hat{\tilde{\pi}}^{(0)}\hat{\tilde{P}}^{(1)}_{\lambda_1},\ j = \mu+1, \mu+2, \ldots, n$$

In the case where these solutions are fuzzy numbers, we have to compare them using some ranking method. Dealing with triangular fuzzy numbers, it is appropriate to use the Chens method *[8]*.

As the most suitable to enter as the basis of the main program, of all the solutions: $\hat{\tilde{\pi}}^{(0)}\hat{\tilde{P}}^{(1)}_{\mu+1}, \hat{\tilde{\pi}}^{(0)}\hat{\tilde{P}}^{(1)}_{\mu+2}, \ldots, \hat{\tilde{\pi}}^{(0)}\hat{\tilde{P}}^{(1)}_{n}$ we choose the one that has in the process of ranking the biggest absolute value. Let this be the solution with index λ_1, i.e., $\min_j \hat{\tilde{\pi}}^{(0)}\hat{\tilde{P}}^{(1)}_{\mu+1} = \hat{\tilde{\pi}}^{(0)}\hat{\tilde{P}}^{(1)}_{\lambda_1},\ j = \mu+1, \mu+2, \ldots, n \cdot$

In accordance with the Wolfe procedure, we return to the main program and write conditions with vectors $P_0, \hat{\tilde{P}}_1, \ldots, \hat{\tilde{P}}_n$ except $\hat{\tilde{P}}^{(1)}_{\lambda_1}$, which we substitute with the convex composition:

$$\hat{\tilde{P}}^{(1)}_{\lambda_1} = \left[\hat{\tilde{P}}^{(1)}_{\lambda_1} \cdot {}_{(1)}\tilde{x}_{\lambda_1} + \hat{\tilde{Q}}^{(\ddot{}2)}_{\lambda_1} \cdot {}_{(2)}\tilde{x}_{\lambda_1} \right] \cdot \frac{1}{\tilde{x}_{\lambda_1}} \tag{3.06}$$

where we have:

$$\tilde{x}_{\lambda_1} = {}_{(1)}\tilde{x}_{\lambda_1} + {}_{(2)}\tilde{x}_{\lambda_1} \tag{3.07}$$

Formulation of convex composition (3.06) with condition (3.07) ensures that $\hat{\tilde{P}}^{(1)}_{\lambda_1} \in \tilde{C}_{\lambda_1}$. This means that we fixed $\hat{\tilde{P}}^{(1)}_{\lambda_1}$, thus we did not weaken the convex set $\tilde{C}_{\lambda_1}$, instead of $\hat{\tilde{P}}^{(1)}_{\lambda_1}$ we take $\hat{\tilde{Q}}^{(\ddot{}2)}_{\lambda_1}$.

From the main program, the ***modified main program of the first degree*** is obtained:

$$\max \tilde{x}_0$$

$$P_0\tilde{x}_0 + \sum_{j=1}^{n} \hat{\tilde{P}}_j \tilde{x}_j + \hat{\tilde{P}}^{(1)}_{\lambda_1} \cdot {}_{(1)}\tilde{x}_{\lambda_1} + \hat{\tilde{Q}}^{(2)}_{\lambda_1} \cdot {}_{(2)}\tilde{x}_{\lambda_1} = \hat{\tilde{b}}$$

$$\hat{\tilde{P}} \in \tilde{C}_j,\ j = \mu+1, \mu+2, \ldots, n;\ j \neq \lambda_1 \tag{3.08}$$

$$\hat{\tilde{P}}_{\lambda_1}^{(1)}, \hat{\tilde{Q}}_{\lambda_1}^{(\cdot\cdot 2)} \in \tilde{C}_{\lambda_1}$$
$$\tilde{x}_j \geq \tilde{0},\ j \neq 0,\ {}_{(1)}\tilde{x}_{\lambda_1} \geq \tilde{0},\ {}_{(2)}\tilde{x}_{\lambda_1} \geq \tilde{0}$$

Expression (3.06) can be written also in the form:

$$\begin{aligned} &\hat{\tilde{P}}_{\lambda_1}^{(1)} \cdot \tilde{x}_{\lambda_1} = \hat{\tilde{P}}_{\lambda_1}^{(1)} \cdot {}_{(1)}\tilde{x}_{\lambda_1} + \hat{\tilde{Q}}_{\lambda_1}^{(\cdot\cdot 2)} \cdot {}_{(2)}\tilde{x}_{\lambda_1} \\ &\tilde{x}_{\lambda_1} = {}_{(1)}\tilde{x}_{\lambda_1} + {}_{(2)}\tilde{x}_{\lambda_1} \end{aligned} \tag{3.09}$$

In (3.09), a $\hat{\tilde{P}}_{\lambda_1}^{(1)}$ is a new vector that enters the basis, thus the conditions in program (3.08) in accordance with the main program (3.03) are not demolished. Therefore, programs (3.08) and (3.03) are equivalent. Vector $\hat{\tilde{P}}_{s_1^{(1)}}^{(1)}$ that leaves the basis is determined using the usual procedure of the simplex algorithm, i.e. criterion of minimal quotient of positive elements of entering column $\hat{\tilde{P}}_{\lambda_1}^{(1)}$ with equilateral coefficients of the right-hand sides. In the case when elements of the entering column $\hat{\tilde{P}}_{\lambda_1}^{(1)}$ and/or right-hand side coefficients $\hat{\tilde{b}}_i, i = 1,2,\ldots,m$ are fuzzy numbers, it is necessary for both columns to first compute its ranking functions and then compute the corresponding quotients.
After the first iteration is done, all the $\tilde{c}_j$ are non-negative in the basic problem (3.03). By this action we obtain the first basic feasible solution of the main program, written as ${}^{(2)}\tilde{x}^{(1)} = \left[L_1^{(1)} \right]^{-1} \cdot \hat{\tilde{b}}^{(0)}$.

3.4 Further feasible solutions

From the basis $\hat{L}_1^{(1)}$ and its corresponding inverse $\left[\hat{L}_1^{(1)} \right]^{-1}$ we obtain new simplex multipliers in the last row, i.e., $\hat{\tilde{\pi}}^{(1)} = \left[-\tilde{\pi}_1^{(1)}\ -\tilde{\pi}_2^{(1)}\ \cdots\ -\tilde{\pi}_m^{(1)}\ ;\ 1 \right]$. Here we have:

$$\hat{\tilde{\pi}}^{(1)} \cdot \hat{\tilde{P}}_j \geq \tilde{0},\ \ j = 1,2,\ldots,\mu$$
$$\hat{\tilde{\pi}}^{(1)} \cdot \hat{\tilde{P}}_{s_1^{(1)}}^{(1)} = 0$$

where $\hat{\tilde{P}}_{s_1^{(1)}}^{(1)}$ is a new basic vector.

Afterwards, we repeat the same iteration as previously and get the problem that we call ***system of second subprograms***:

$$\begin{array}{ll} \min\limits_{i} \hat{\tilde{\pi}}^{(1)} \cdot \hat{\tilde{P}}_j & \min\limits_{i} \hat{\tilde{\pi}}^{(1)} \cdot \hat{\tilde{Q}}_{s_1}^{(2)} \\ \hat{\tilde{P}}_j \in \tilde{C}_j & \hat{\tilde{Q}}_{s_1}^{(2)} \in \tilde{\mathrm{C}}_{s_1} \\ j = \mu+1, \mu+2, \ldots, n;\ j \neq s_1^{(1)} & \\ i = 1, 2, \ldots m & \end{array}$$

The process is similar to seeking the first solution. Namely, we find all solutions of the system of the second subprograms, then choose the minimum average from them, and then initiate the corresponding column into the basis of the main program. This yields a modified main program of the second degree, a subsequently modified program of the third degree, and so forth.

In general, in the *k*-th step we obtain a ***modified main program of degree (k-1)****:*

$$\begin{aligned} & \max \tilde{x}_0 \\ & P_0 \tilde{x}_0 + \sum_{j=1}^{\mu} \hat{\tilde{P}}_j \tilde{x}_j + \sum_{s_j} \left(\sum_{\varepsilon=1}^{k-1} \hat{\tilde{P}}_{s_j}^{(\varepsilon)} \cdot {}_{(\varepsilon)}\tilde{x}_{s_j} + \sum_{\varepsilon=1}^{k-1} \hat{\tilde{Q}}_{s_j}^{(\varepsilon+1)} \cdot {}_{(\varepsilon+1)}\tilde{x}_{s_j} \right) = \hat{\tilde{b}} \\ & s_j \in I_2 \subseteq \{\mu+1, \mu+2, \ldots, n\} \\ & \hat{\tilde{P}} \in \tilde{C}_j,\ j \in \{\mu+1, \mu+2, \ldots, n\} - I_2 \\ & \hat{\tilde{P}}_{s_j}^{(\varepsilon)}, \hat{\tilde{Q}}_{s_j}^{(\varepsilon+1)} \in \tilde{C}_{s_j},\ s_j \in I_2 \\ & \tilde{x}_j \geq \tilde{0},\ j \neq 0,\ {}_{(\varepsilon)}\tilde{x}_{s_j} \geq \tilde{0},\ {}_{(\varepsilon+1)}\tilde{x}_{s_j} \geq \tilde{0} \end{aligned} \tag{3,10}$$

Assume that while solving the program (3.10) we find a basic feasible solution ${}^{(2)}\tilde{x}^{(k-1)}$. This solution corresponds to basis $\hat{L}^{(k-1)}$ and its corresponding inverse $\left[\hat{L}^{(k-1)}\right]^{-1}$. The last row in this inverse is the row of simplex multipliers: $\hat{\tilde{\pi}}^{(k-1)} = \left[-\tilde{\pi}_1^{(k-1)}\ -\tilde{\pi}_2^{(k-1)}\ \cdots\ -\tilde{\pi}_m^{(k-1)}\ ;\ 1\right]$

According to definition of the simplex algorithm we have:

$$\begin{aligned} & \hat{\tilde{\pi}}^{(k-1)} \cdot \hat{\tilde{P}}_j \geq 0 \qquad j = 1, 2, \ldots, \mu \\ & \hat{\tilde{\pi}}^{(k-1)} \cdot \hat{\tilde{P}}_j^{(k-1)} = 0 \quad j \in \{1, 2, \ldots, n\} \quad , j - \text{basic index} \\ & \hat{\tilde{\pi}}^{(k-1)} \cdot P_0 = 1 \\ & \hat{\tilde{\pi}}^{(k-1)} \cdot \hat{\tilde{P}}_j^{(k-1)} < 0 \quad j \in \{\mu+1, \mu+2, \ldots, n\}\ , j - \text{nonbasic index} \end{aligned}$$

From the $(k-1)$-th solution we attain the k-th solution, if we manage to solve the ***system of k-th subprograms***:

$$\begin{array}{ll}
\min\limits_i \hat{\tilde{\pi}}^{(k-1)} \cdot \hat{\tilde{P}}_j^{(k-1)} & \min\limits_i \hat{\tilde{\pi}}^{(k-1)} \cdot \hat{\tilde{Q}}_{s_j}^{(k)} \\
\hat{\tilde{P}}_j^{(k-1)} \in \tilde{C}_j & \hat{\tilde{Q}}_{s_j}^{(k)} \in \tilde{C}_{s_j} \\
j \in \{1,2,\ldots,n\} - I_2 & s_j \in I_2 \\
I_2 \subseteq \{\mu+1, \mu+2, \ldots, n\} &
\end{array}$$

From here we obtain the ***modified main program of degree k***:

$$\begin{array}{l}
\max \tilde{x}_0 \\
P_0\tilde{x}_0 + \sum\limits_{j=1}^{\mu} \hat{\tilde{P}}_j \tilde{x}_j + \sum\limits_{s_j}\left(\sum\limits_{\varepsilon=1}^{k-1} \hat{\tilde{P}}_{s_j}^{(\varepsilon)} \cdot {}_{(\varepsilon)}\tilde{x}_{s_j} + \sum\limits_{\varepsilon=1}^{k-1} \hat{\tilde{Q}}_{s_j}^{(\varepsilon+1)} \cdot {}_{(\varepsilon+1)}\tilde{x}_{s_j} \right) + \hat{\tilde{P}}_{s_k}^{(k)} \cdot {}_{(k)}\tilde{x}_{s_k} + \hat{\tilde{Q}}_{s_k}^{(k+1)} \cdot {}_{(k+1)}\tilde{x}_{s_k} = \hat{\tilde{b}} \\
s_k, s_j \in I_2 \subseteq \{\mu+1, \mu+2, \ldots, n\} \\
\hat{\tilde{P}}_j \in \tilde{C}_j,\ j \in \{\mu+1, \mu+2, \ldots, n\} - I_2 \\
\hat{\tilde{P}}_{s_j}^{(\varepsilon)}, \hat{\tilde{Q}}_{s_j}^{(\varepsilon+1)} \in \tilde{C}_{s_j},\ s_j \in I_2 \\
\tilde{x}_j \geq \tilde{0},\ j \neq 0,\ {}_{(\varepsilon)}\tilde{x}_{s_j} \geq \tilde{0},\ {}_{(k)}\tilde{x}_{s_k} \geq \tilde{0},\ {}_{(k)}\tilde{x}_{s_k} \geq \tilde{0},\ {}_{(k+1)}\tilde{x}_{s_k} \geq \tilde{0}
\end{array} \tag{3.11}$$

3.5 The existence of the optimal solution

We proceed with the described steps as long as we can improve the value of objective function $\tilde{x}_0$. We obtain the optimal feasible solution on the k_0 -th step only when in the modified main program of degree k_0, and the vector of simplex multipliers $\hat{\tilde{\pi}}_i^{(k_0)}, i = 1,2,\ldots, p_{k_0}$ fulfil:

$$\begin{array}{lll}
\hat{\tilde{\pi}}^{(k_0)} \cdot \hat{\tilde{P}}_j \geq 0 & , j = 1,2,\ldots,\mu & \\
\hat{\tilde{\pi}}^{(k_0)} \cdot \hat{\tilde{P}}_j^{(k_0)} = 0 & , j \in \{\mu+1, \mu+2, \ldots, n\} & , j - \text{basic index} \\
\hat{\tilde{\pi}}^{(k_0)} \cdot P_0 = 1 & & \\
\hat{\tilde{\pi}}^{(k_0)} \cdot \hat{\tilde{P}}_j^{(k_0)} \geq 0 & , j \in \{\mu+1, \mu+2, \ldots, n\} & , j - \text{nonbasic index}
\end{array}$$

In this circumstance, in the system of (k_0+1) - th subprograms:

$$\min_i \hat{\tilde{\pi}}^{(k_0)} \cdot \hat{\tilde{P}}_j^{(k_0)}$$
$$\hat{\tilde{P}}_j^{(k_0)} \in \tilde{C}_j$$
$$j \in \{\mu+1, \mu+2, \dots, n\}$$

we cannot find any product $\hat{\tilde{\pi}}^{(k_0)} \cdot \hat{\tilde{P}}_j^{(k_0)}$ that is negative, and therefore it is impossible to bring such a vector into the basis of the main program that would improve the value of objective function $\tilde{x}_0$. Thus, we have found the optimal solution of program (3.02), since program (3.02) and modified main program (3.11) are equivalent and therefore have the same optimal solution.

3.6 Degenerate solution

In the given algorithm, degeneracy is also possible. The approach for solving problems in such cases is briefly described in *[2].*

4 NUMERICAL EXAMPLE

We want to solve the problem:

$$\begin{aligned}
\min z = 2x_1 + 6x_2 + 8x_3 + 5x_4 + c_5x_5 + c_6x_6 \\
4x_1 + x_2 + 2x_3 + 2x_4 + a'_{15}x_5 + a'_{16}x_6 \ge (78,80,82) \\
2x_1 + 5x_2 + 4x_4 + a'_{25}x_5 + a'_{26}x_6 \ge 40 \\
2x_2 + 4x_3 + x_4 + a'_{35}x_5 + a'_{36}x_6 \ge (118,120,122) \\
x_i \ge 0,\ i = 1,2,3,4,5,6
\end{aligned}$$

with the given convex restriction of varying columns $\hat{P} \in C_5$, $\hat{P}_6 \in C_6$.

$$C_5 : \begin{cases} a'_{15} + a'_{25} + a'_{35} - a'_{45} = 3 \\ 2a'_{15} + 5a'_{25} + a'_{35} + 3a'_{45} = (9,10,11) \\ a'_{i5} \ge 0,\ i = 1,2,3,4 \end{cases}$$

$$C_6 : \begin{cases} 2a'_{16} + a'_{26} + a'_{36} + 2a'_{46} = (5,6,7) \\ 4a'_{16} + 2a'_{26} + 2a'_{36} - 4a'_{46} = 8 \\ a'_{i6} \ge 0,\ i = 1,2,3,4 \end{cases}$$

We solve this problem in two phases: in the first phase we solve the linear program without varying columns, and in the second phase we deal with systems of subprograms.

Main program

$$\max x_0$$

$$P_0 x_0 + \hat{P}_1 x_1 + \hat{P}_2 x_2 + \hat{P}_3 x_3 + \hat{P}_4 x_4 + \hat{P}_5 x_5 + \hat{P}_6 x_6 = \hat{\hat{b}}$$

$$P_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \hat{P}_1 = \begin{bmatrix} 4 \\ 2 \\ 0 \\ 2 \end{bmatrix}, \hat{P}_2 = \begin{bmatrix} 1 \\ 5 \\ 2 \\ 6 \end{bmatrix}, \hat{P}_3 = \begin{bmatrix} 2 \\ 0 \\ 4 \\ 8 \end{bmatrix}, \hat{P}_4 = \begin{bmatrix} 2 \\ 4 \\ 1 \\ 5 \end{bmatrix}, \hat{P}_5 = \begin{bmatrix} a'_{15} \\ a'_{25} \\ a'_{13} \\ a'_{45} \end{bmatrix}, \hat{P}_6 = \begin{bmatrix} a'_{16} \\ a'_{26} \\ a'_{16} \\ a'_{46} \end{bmatrix}$$

Initial main program without varying columns is of the form:

$$\begin{aligned} &\min z = 2x_1 + 6x_2 + 8x_3 + 5x_4 \\ &\quad 4x_1 + x_2 + 2x_3 + 2x_4 \geq (78, 80, 82) \\ &\quad 2x_1 + 5x_2 \qquad + 4x_4 \geq 40 \\ &\qquad 2x_2 + 4x_3 + x_4 \geq (118, 120, 122) \\ &\quad x_i \geq 0, i = 1,2,3,4,5,6 \end{aligned}$$

The fuzzy solution of the initial main program is *[2]:*

$$\tilde{x}_1 = (4.75, 5, 5.25) \quad \tilde{x}_2 = (6.1, 6, 5.9) \quad \tilde{x}_3 = (26.45, 27, 27.55) \quad \tilde{x}_4 = (0,0,0)$$

$$\tilde{z}_{\min} = (257.7, 262, 266.3) \quad \hat{\pi}^{(0)} = [-0.3,\ -0.4,\ -1.85 \ \vdots\ 1]$$

<u>We obtain the system of the first subprograms:</u>

a) For vector $\hat{\hat{P}}_5$:

$$\begin{aligned} \min \hat{\pi}^{(0)} \hat{\hat{P}}_5 &= -0,3a'_{15} - 0,4a'_{25} - 1,85a'_{35} + a'_{45} \\ a'_{15} + a'_{25} + a'_{35} - a'_{45} &= 3 \\ 2a'_{15} + 5a'_{25} + a'_{35} + 3a'_{45} &= (9,10,11) \end{aligned}$$

Solution:

$$\hat{\hat{P}}_5 = \begin{bmatrix} 0 \\ 0 \\ (4.5, 4.75, 5) \\ (1.5, 1.75, 2) \end{bmatrix}, \quad \min \hat{\pi}^{(0)} \hat{\hat{P}}_5 = (-6.825, -7.0375, -7.25)$$

b) For vector $\hat{\hat{P}}_6$

$$\begin{aligned} \min \hat{\pi}^{(0)} \hat{\hat{P}}_6 &= -0,3a'_{16} - 0,4a'_{26} - 1,85a'_{36} + a'_{46} \\ 2\,a'_{16} + a'_{26} + a'_{36} + 2\,a'_{46} &= (5,6,7) \\ 4a'_{16} + 2a'_{26} + 2a'_{36} - 4a'_{46} &= 8 \end{aligned}$$

Solution:

$$\hat{\tilde{P}}_5=\begin{bmatrix}0\\0\\(4.5,4.75,5)\\(1.5,1.75,2)\end{bmatrix}\quad,\quad \min\hat{\pi}^{(0)}\hat{\tilde{P}}_5=(-6.825,-7.0375,-7.25)$$

b) For vector $\hat{\tilde{P}}_6$

$$\min\hat{\pi}^{(0)}\hat{\tilde{P}}_6=-0,3a'_{16}-0,4a'_{26}-1,85a'_{36}+a'_{46}$$
$$2\,a'_{16}+\quad a'_{26}+\quad a'_{36}+2\,a'_{46}=(5,6,7)$$
$$4a'_{16}+\quad 2a'_{26}+\quad 2a'_{36}-4a'_{46}=8$$

Solution:

$$\hat{\tilde{P}}_6=\begin{bmatrix}0\\0\\(4.5,5,5.5)\\(0.25,0.5,0.75)\end{bmatrix}\quad,\quad \min\hat{\pi}^{(0)}\hat{\tilde{P}}_6=(-8.075,-8.75,-9.425)$$

We rank both solutions, and after considering $R\left(\hat{\pi}^{(0)}\hat{\tilde{P}}_6\right)>R\left(\hat{\pi}^{(0)}\hat{\tilde{P}}_5\right)$, we choose column $\hat{\tilde{P}}_6$ to move to the basis of the main program of the first degree.

Modified main program of the first degree:

$$\max x_0$$
$$P_0\tilde{x}_0+\hat{P}_1\tilde{x}_1+\hat{P}_2\tilde{x}_2+\hat{P}_3\tilde{x}_3+\hat{P}_4\tilde{x}_4+\hat{P}_5\tilde{x}_5+\left(\hat{\tilde{P}}_6^{(1)}\cdot{}_{(1)}\tilde{x}_6+\hat{\tilde{\tilde{Q}}}_6^{(2)}\cdot{}_{(2)}\tilde{x}_6\right)=\hat{\tilde{b}}$$

$$P_0=\begin{bmatrix}0\\0\\0\\1\end{bmatrix},\hat{P}_1=\begin{bmatrix}4\\2\\0\\2\end{bmatrix},\hat{P}_2=\begin{bmatrix}1\\5\\2\\6\end{bmatrix},\hat{P}_3=\begin{bmatrix}2\\0\\4\\8\end{bmatrix},\hat{P}_4=\begin{bmatrix}2\\4\\1\\5\end{bmatrix},\hat{P}_5=\begin{bmatrix}a'_{15}\\a'_{25}\\a'_{13}\\a'_{45}\end{bmatrix},$$

$$\hat{\tilde{P}}_6^{(1)}=\begin{bmatrix}0\\0\\(4.5,5,5.5)\\(0.25,0.5,0.75)\end{bmatrix},\hat{\tilde{Q}}_6^{(2)}=\begin{bmatrix}a'_{16}\\a'_{26}\\a'_{16}\\a'_{46}\end{bmatrix}$$

has the solution:

$$\tilde{z}_{\min}=(51.8,52,53.2)$$
$$\tilde{x}_1=(20,20,20.5)$$
$$x_2=x_3=x_4=0$$
$${}_{(1)}\tilde{x}_6=(23.6,24,24.4)$$
$$\hat{\tilde{\pi}}^{(1)}=\left[(0,-0.5,-0,5),(-1,0,0,),(-0.1,-0.1.-0,1);1\right]$$

We continue with the steps and attain an **optimal solution,** which is a result of the modified main program of degree 6.

$$z_{\min} = (31.4, 32.0, 32.6)$$
$$x_1 = x_2 = x_3 = x_4 = 0$$
$${}_{(3)}x_5 = {}_{(1)}x_6 = {}_{(2)}x_6 = 0$$
$${}_{(4)}\tilde{x}_6 = (8,8,8)$$
$${}_{(5)}\tilde{x}_6 = (31.2, 32.0, 32.8)$$
$${}_{(6)}\tilde{x}_6 = (23.6, 24.0, 24.4)$$

Decision variable $x_5 = {}_{(3)}x_5 = 0$, decision variable x_6 in the optimal solution is composed as:

$$\tilde{x}_6 = {}_{(4)}\tilde{x}_6 + {}_{(5)}\tilde{x}_6 + {}_{(6)}\tilde{x}_6 = (8,8,8) + (31.2,32,32.8) + (23.6,24.0,24.4) = (62.8,64.0,65.2)$$

The columns $\hat{\tilde{P}}_5$ and $\hat{\tilde{P}}_6$ are:

$$\hat{\tilde{P}}_5 = \hat{\tilde{P}}_5^{(3)} = \begin{bmatrix} (3.6,3.8,4) \\ 0 \\ 0 \\ (0.6,\ 0.8,1) \end{bmatrix} \quad \hat{\tilde{P}}_6 = \begin{bmatrix} (1.118,1.25,1.383) \\ (0.573,0.625,0.675) \\ (1.691,1.875,2.058) \\ (0.25,0.50,0.75) \end{bmatrix}$$

The original fuzzy linear program that also includes columns $\hat{\tilde{P}}_5$ and $\hat{\tilde{P}}_6$ is now:

$$\begin{aligned}
\min z &= 2x_1 + 6x_2 + 8x_3 + 5x_4 + (0.6,\ 0.8,1)\cdot x_5 + (0.25,0.50,0.75)\cdot x_6 \\
& 4x_1 + x_2 + 2x_3 + 2x_4 + (3.6,3.8,4)\cdot x_5 + (1.118,1.25,1.383)\cdot x_6 \geq (78,80,82) \\
& 2x_1 + 5x_2 \qquad + 4x_4 + \qquad (0.573,0.625,0.675)\cdot x_6 \geq 40 \\
& \qquad 2x_2 + 4x_3 + x_4 + \qquad (1.691,1.875,2.058)\cdot x_6 \geq (118,120,122) \\
& x_i \geq 0,\ i = 1,2,3,4,5,6
\end{aligned}$$

5 CONCLUSIONS

Generalisation of a linear program using Wolfe's modified simplex method is also possible and reasonable for fuzzy linear programming. Computational stages are based on a theoretical algorithm that demands multiple iterations of a two-phase procedure. In the first phase we find a solution for the modified main program, and in the second phase we solve the system of subprograms, where we determine the entering vector for basis. The use of simplex multipliers requires the well-known Dantzig simplex algorithm, which can be easily solved using several computer software programs. Dealing with a linear program where some or even all coefficients are fuzzy numbers still represents a challenge. In this case we must apply one of the methods found in *[2].*

The following research opens up a wide range of possibilities by introducing dynamics, i.e., taking into account the time component in fuzzy numbers/sets. Fuzzy sets describing a fuzzy linear program in this case become dependent on time: $\tilde{A}(t), \tilde{c}(t), \tilde{b}(t)$ and $\tilde{x}(t),\ t \in [0,T], T < +\infty$.

An extended application of fuzzy linear programming and its generalisation in actual real-life problems depend on powerful computer software capable of solving such problems. Unfortunately, such software does not yet exist.

References

[1] **G.B. Dantzig:** *Linear Programming and Extensions,* Princeton University Press, Princeton, New Jersey, 1998), eleventh printing.

[2] **J. Usenik and M. Žulj:** *Generalizirano mehko linearno programiranje,* Novo mesto: Fakulteta za organizacijske študije, 2021.

[3] **V. Rupnik:** *Zvezno dinamično linearno programiranje*, Ekonomska, poslovna in organizacijska knjižnica, 24, Založba Obzorja Maribor, Maribor, 1978.

[4] **J. Usenik**: Fuzzy dynamic linear programming in energy supply planning, *Journal of energy technology*, Oct. 2011, vol. 4, iss. 4, pp. 45–62.

[5] **J. Usenik:** *Generalizirano zvezno variabilno dinamično linearno programiranje,* Novo mesto: Fakulteta za industrijski inženiring, 2017.

[6] **L. A. Zadeh:** *Fuzzy sets*, Information and control, 8(3):338–353, 1965

[7] **H.-J. Zimmermann:** *Fuzzy Set Theory and its Applications*, Kluwer Academic Publishers (2001), Fourth edition.

[8] **S. Chen, C. Hwang, P. Hwang**: *Fuzzy Multiple Attribute Decision Making,* Springer-Verlag, (Berlin, Heidelberg, New York, 1992).

[9] **M. Repnik and D. Bokal***: A Basis for Taxonomy of Fuzzy Linear Programming Methods,* In 13th International Symposium on Operational Research in Slovenia, Bled, Slovenia, September 23-25, 2015. ZADNIK STIRN, Lidija (ur.), et al. SOR ’15 proceedings, pages 188–192. Ljubljana: Slovenian Society Informatika, Section for Operational Research, 2015