

Solving fully fuzzy linear system with arbitrary triangular fuzzy numbers (m, α, β)

Neetu Babbar · Amit Kumar · Abhinav Bansal

Published online: 19 October 2012
© Springer-Verlag Berlin Heidelberg 2012

Abstract In this paper, we discuss some new numerical methods to solve a fully fuzzy linear system (FFLS) with triangular fuzzy numbers of the form (m, α, β) . Almost every existing method that intends to solve a FFLS confines the coefficient matrix and the solutions to be non-negative fuzzy numbers. The main intent of the proposed methods is to remove these restrictions and widen the scope of fuzzy linear systems in scientific applications. The methods are illustrated with the help of numerical examples and are conceptually easy to understand and apply in real life situations.

Keywords Fully fuzzy linear systems (FFLS) · Fuzzy matrix · Triangular fuzzy numbers · Fuzzy optimization

1 Introduction

System of simultaneous linear equations plays a major role in various areas such as operational research, physics, statistics, engineering and social sciences. When the estimation of the system coefficients is imprecise and only some vague knowledge about the actual values of the parameters is available, it may be convenient to represent some or all of them with fuzzy numbers (Zadeh 1965). Fuzzy number arithmetic is widely applied and useful in computation of linear system whose parameters are

represented by fuzzy numbers, which are called fuzzy linear systems (FLS). A general model for solving a $n \times n$ fuzzy linear system whose coefficient matrix is crisp and the right-hand side column is an arbitrary fuzzy vector was first proposed by Friedman et al. (1998). A review of some methods for solving these systems can be found in (Abbasbandy et al. 2005, 2006; Abbasbandy and Jafarian 2006; Allahviranloo 2004a, b, 2005; Allahviranloo and Ghanbari 2010; Buckley and Qu 1991; Dehghan and Hashemi 2006; Friedman et al. 2000; Mosleh and Otadi 2010; Sun and Guo 2009; Yin and Wang 2009).

In addition, another important kind of fuzzy linear systems including fuzzy numbers in which all parameters are fuzzy numbers and is named as FFLS. For a $n \times n$ fully fuzzy linear systems, Dehghan and Hashemi (2006a) and Dehghan et al. (2007) proposed the Adomian decomposition method, and other iterative methods to find the positive fuzzy vector solution of fully fuzzy linear system. Dehghan et al. (2006) proposed some computational methods such as Cramer's rule, Gauss elimination method, LU decomposition method and linear programming approach for finding the approximated solution of FFLS. Muzzioli and Reynaerts (2006) introduced an algorithm to find vector solution by transforming the system $A_1x + b_1 = A_2x + b_2$ into the FFLS $Ax = b$ where $A = A_1 - A_2$ and $b = b_1 - b_2$.

Mosleh et al. (2007) proposed a method to find the solution of fully fuzzy linear system of the form $Ax + b = Cx + d$ with A, C square matrices of fuzzy coefficients and b, d fuzzy number vectors and the unknown vector x is vector consisting of n fuzzy numbers. Nasseri et al. (2008) used decomposition methods of the coefficient matrix for solving fully fuzzy linear system of equations. Allahviranloo et al. (2008) proposed a method for solving FFLS $Ax = b$, when coefficient matrix is positive. Gao (2009)

N. Babbar · A. Kumar
School of Mathematics and Computer Applications,
Thapar University, Patiala 147004, India

A. Bansal (✉)
Computer Science and Engineering Department,
Thapar University, Patiala 147004, India
e-mail: abhinav.bansal8@gmail.com

proposed a unified iterative scheme for solving non-square FFLS. Nasseri et al. (2009) proposed Greville's method to find the positive solution of FFLS. Mosleh et al. (2009) discussed symmetric times triangular (ST) decomposition to find the positive solution of FFLS. Nasseri and Zahmatkesh (2010) proposed a new method for computing the non-negative solution of FFLS. Liu (2010) developed a block homotopy perturbation method (HPM) for finding the approximation of the solution of FFLS. Nasseri and Sohrabi (2010) used a certain decomposition of the coefficient matrix of the FFLS to construct a new algorithm for solving FFLS. Kumar et al. (2010) proposed a new method to find the approximate solution of FFLS with trapezoidal fuzzy numbers as parameters. All the methods that have been reported in the literature presume the non-negativity constraint on the coefficients and the solutions that has greatly reduced the applicability and employability of FFLS in scientific applications with almost negligible use.

In this paper, we firstly discuss a new computational method for finding the non-negative solutions of a FFLS $\tilde{A} \otimes \tilde{x} = \tilde{b}$, where \tilde{A} is a fuzzy matrix with no restriction on its elements and \tilde{x} and \tilde{b} are fuzzy vectors with appropriate sizes. Later, we present a method to find all the solutions of a FFLS $\tilde{A} \otimes \tilde{x} = \tilde{b}$ with non-negative restriction on the coefficient matrix ($\tilde{A} \geq 0$). Lastly, a new method is proposed which jointly removes the sign constraints on the coefficients and the solution vector. All the methods are illustrated by solving numerical examples which cannot be solved by any of the existing methods. Further the methods are conceptually and computationally very simple. This simplicity is necessary and is required so as to revive the inclusion of fuzzy linear systems in scientific applications, which till date has limited applications due to the restricted nature of operations on fuzzy numbers and narrow mathematical usability.

The rest of this paper is organized as follows: In Sect. 2, shortcomings of the existing methods to solve FFLS are described. In Sect. 3 some basic definitions are reviewed. In Sect. 4 new methods are proposed for solving FFLS and also to illustrate the proposed methods, numerical examples are solved. In Sect. 5 the implications of the proposed methods are discussed and the paper is concluded in Sect. 6.

2 Shortcomings of existing methods

In this section, the shortcomings in the existing methods (Abbasbandy et al. 2005, 2006; Abbasbandy and Jafarian 2006; Allahviranloo 2004a, b, 2005; Allahviranloo and Ghanbari 2010; Allahviranloo et al. 2008; Buckley and Qu 1991; Dehghan and Hashemi 2006a, b; Dehghan et al.

2006; Dehghan and Hashemi 2007; Gao 2009; Engineering Mathematics 1 2010; Liu 2010; Mosleh and Otadi 2010; Mosleh et al. 2009; Nasseri et al. 2008, 2009; Nasseri and Sohrabi 2010; Nasseri and Zahmatkesh 2010; Sun and Guo 2009; Yin and Wang 2009) are pointed out:

1. All the existing methods presume the non-negativity of the coefficient matrix, i.e., $\tilde{A} \geq 0$. This restriction creates difficulty in using the existing methods to solve FFLS occurring in real life situations for which the coefficient matrix may not be entirely non-negative.
2. In all the existing methods, it is assumed that the system of equations is consistent and then the methods are developed, i.e., consistency of the FFLS cannot be checked using the existing methods. Also, in the existing methods there is no provision for determining if the solution is unique or infinite and the nature of infinite solution.
3. Further all the existing methods aim at finding the "non-negative" solutions of a FFLS. This apriori restriction creates difficulty in using the methods as the solutions may not be entirely non-negative.

This abundance of methods with several restrictions on the parameters will eventually serve no purpose if the concept of fuzzy linear systems is not suitably applied in scientific problems. There is thus an overwhelming need to develop computational methods which assume no restrictions on the parameters and widen the boundaries of applied fuzzy mathematics. To overcome the above described shortcomings, in Sect. 4 three new computational methods are proposed for solving a FFLS which tend to remove the sign restrictions imposed.

3 Preliminaries

In this section, some necessary backgrounds and notions of fuzzy set theory are reviewed (Dubois and Prade 1980; Kaufmann and Gupta 1985).

Definition 3.1 Let X denote a universal set. Then the fuzzy subset \tilde{A} of X is defined by its membership function $\mu_{\tilde{A}} : X \rightarrow [0, 1]$ which assigns a real number $\mu_{\tilde{A}}(x)$ in the interval $[0, 1]$ to each element $x \in X$, where the value of $\mu_{\tilde{A}}(x)$ at x shows the grade of membership of x in \tilde{A} .

Definition 3.2 A fuzzy set \tilde{A} , defined on the universal set of real number R , is said to be a fuzzy number if its membership function has the following characteristics:

- (i) \tilde{A} is convex, i.e., $\mu_{\tilde{A}}(\lambda x_1 + (1 - \lambda)x_2) \geq \min(\mu_{\tilde{A}}(x_1), \mu_{\tilde{A}}(x_2)) \forall x_1, x_2 \in R, \forall \lambda \in [0, 1]$.
- (ii) \tilde{A} is normal, i.e., $\exists x_0 \in R$ such that $\mu_{\tilde{A}}(x_0) = 1$.

(iii) $\mu_{\tilde{A}}$ is piecewise continuous.

Support of a fuzzy number is defined by the set $\text{Supp}(\tilde{A}) = \{x|x \in X, \mu_{\tilde{A}}(x) > 0\}$.

Definition 3.3 A fuzzy number $\tilde{A} = (m, \alpha, \beta)$ is said to be a triangular fuzzy number if its membership function is given by:

$$\mu_{\tilde{A}}(x) = \begin{cases} 1 - \frac{m-x}{\alpha}, & m - \alpha \leq x \leq m, \alpha > 0, \\ 1 - \frac{x-m}{\beta}, & m \leq x \leq m + \beta, \beta > 0 \\ 0 & \text{otherwise} \end{cases}$$

where, $m, \alpha, \beta \in R$.

Support of a triangular fuzzy number is the open interval $(m - \alpha, m + \beta)$.

Definition 3.4 A triangular fuzzy number $\tilde{A} = (m, \alpha, \beta)$ is said to be non-negative (negative) triangular fuzzy number, i.e., $\tilde{A} \geq 0$ ($\tilde{A} < 0$) if and only if $m - \alpha \geq 0$ ($m - \alpha > 0$).

A triangular fuzzy number $\tilde{A} = (m, \alpha, \beta)$ is said to be non-positive triangular fuzzy number, i.e., $\tilde{A} \leq 0$ ($\tilde{A} < 0$) if and only if $m + \beta \leq 0$ ($m + \beta < 0$). A zero triangular fuzzy number is denoted by $(0, 0, 0)$.

Definition 3.5 Two fuzzy numbers $\tilde{M} = (m, \alpha, \beta)$ and $\tilde{N} = (n, \gamma, \delta)$ are said to be equal if and only if $m = n, \alpha = \gamma$ and $\beta = \delta$.

Definition 3.6 A matrix $\tilde{A} = (\tilde{a}_{ij})$ is called a fuzzy matrix, if each element of \tilde{A} is a fuzzy number. \tilde{A} will be positive (negative) and denoted by $\tilde{A} > 0$ ($\tilde{A} < 0$) if each element of \tilde{A} be positive (negative). \tilde{A} will be non-positive (non-negative) and denoted by $\tilde{A} \leq 0$ ($\tilde{A} \geq 0$) if each element of \tilde{A} be non-positive (non-negative). We may represent $n \times m$ fuzzy matrix $\tilde{A} = (\tilde{a}_{ij})_{n \times m}$, where $\tilde{a}_{ij} = (a_{ij}, b_{ij}, c_{ij})$.

Definition 3.7 Let $\tilde{A} = (\tilde{a}_{ij})$ and $\tilde{B} = (\tilde{b}_{ij})$ be two $m \times n$ and $n \times p$ fuzzy matrices. We define $\tilde{A} \otimes \tilde{B} = \tilde{C} = (\tilde{c}_{ij})$ which is the $m \times p$ matrix where $\tilde{c}_{ij} = \sum_{k=1, \dots, n}^{\oplus} \tilde{a}_{ik} \otimes \tilde{b}_{kj}$.

Definition 3.8 Let $\tilde{A}_1 = (m, \alpha, \beta)$ and $\tilde{A}_2 = (n, \gamma, \delta)$ be two triangular fuzzy numbers, then

- (i) $(m, \alpha, \beta) \oplus (n, \gamma, \delta) = (m + n, \alpha + \gamma, \beta + \delta)$
- (ii) $-(m, \alpha, \beta) = (-m, \beta, \alpha)$
- (iii) $(m, \alpha, \beta) \ominus (n, \gamma, \delta) = (m - n, \alpha + \delta, \beta + \gamma)$.

3.1 Extended multiplication operation

In the literature (Dubois and Prade 1980; Kaufmann and Gupta 1985), the product of two fuzzy numbers $\tilde{A}_1 = (m, \alpha, \beta)$ and $\tilde{A}_2 = (n, \gamma, \delta)$ is defined as:

$$\tilde{A}_1 \otimes \tilde{A}_2 = (m, \alpha, \beta) \otimes (n, \gamma, \delta) = (mn, mn - \min((m - \alpha)(n - \gamma), (m + \beta)(n - \gamma)(n + \delta)), (m + \beta)(n + \delta), (m - \alpha)(n + \delta)), \max((m - \alpha)(n - \gamma), (m + \beta)(n - \gamma), (m + \beta)(n + \delta), (m - \alpha)(n + \delta)) - mn)$$

If either \tilde{A}_1 or \tilde{A}_2 is an unknown fuzzy number then it is cumbersome to use this expression to find the product $\tilde{A}_1 \otimes \tilde{A}_2$. To overcome this difficulty, the expression can be simplified on the basis of the sign of the operand:

$$\tilde{A}_1 \otimes \tilde{A}_2 = \begin{cases} (mn, f_1, f_2) & \text{if } (m, \alpha, \beta) \geq 0 \\ (mn, f_3, f_4) & \text{if } (m, \alpha, \beta) \leq 0 \\ (mn, f_5, f_6) & \text{otherwise} \end{cases}$$

$$\begin{aligned} f_1 &= mn - \min((m - \alpha)(n - \gamma), (m + \beta)(n - \gamma)), \\ f_2 &= \max((m - \alpha)(n + \delta), (m + \beta)(n + \delta)) - mn \\ f_3 &= mn - \min((m - \alpha)(n + \delta), (m + \beta)(n + \delta)) \\ f_4 &= \max((m - \alpha)(n - \gamma), (m + \beta)(n - \gamma)) - mn \\ f_5 &= mn - \min((m - \alpha)(n + \delta), (m + \beta)(n - \gamma)) \\ f_6 &= \max((m - \alpha)(n - \gamma), (m + \beta)(n + \delta)) - mn \end{aligned}$$

$$\begin{aligned} \min(x, y) &= \left(\frac{x+y}{2}\right) - \left|\frac{x-y}{2}\right|, \\ \max(x, y) &= \left(\frac{x+y}{2}\right) + \left|\frac{x-y}{2}\right| \end{aligned}$$

4 Proposed methods

In this section, three new computational methods are proposed to solve a FFLS $\tilde{A} \otimes \tilde{x} = \tilde{b}$ comprising triangular fuzzy numbers. The methods are conceptually simple and rely on analytical decomposition of a fuzzy linear system into its equivalent crisp linear form. Consider the $n \times n$ FFLS as:

$$\begin{aligned} (\tilde{a}_{11} \otimes \tilde{x}_1) \oplus (\tilde{a}_{12} \otimes \tilde{x}_2) \oplus \dots \oplus (\tilde{a}_{1n} \otimes \tilde{x}_n) &= \tilde{b}_1 \\ (\tilde{a}_{21} \otimes \tilde{x}_1) \oplus (\tilde{a}_{22} \otimes \tilde{x}_2) \oplus \dots \oplus (\tilde{a}_{2n} \otimes \tilde{x}_n) &= \tilde{b}_2 \\ &\vdots \\ (\tilde{a}_{n1} \otimes \tilde{x}_1) \oplus (\tilde{a}_{n2} \otimes \tilde{x}_2) \oplus \dots \oplus (\tilde{a}_{nn} \otimes \tilde{x}_n) &= \tilde{b}_n \end{aligned}$$

The matrix form of the above equation is $\tilde{A} \otimes \tilde{x} = \tilde{b}$ where the coefficient matrix $\tilde{A} = (\tilde{a}_{ij}), 1 \leq i, j \leq n$ is a $n \times n$ fuzzy matrix. Let $\tilde{a}_{ij} = (a_{ij}, m_{ij}, n_{ij}), \tilde{x} = (x_i, y_i, z_i) \geq 0, \tilde{b} = (b_i, g_i, h_i)$ be triangular fuzzy numbers. We may represent a $n \times n$ fuzzy matrix $\tilde{A} = (\tilde{a}_{ij})_{n \times n}$ where $(\tilde{a}_{ij}) = (a_{ij}, m_{ij}, n_{ij})$ with the notation $\tilde{A} = (A, M, N)$, where $A = (a_{ij}), M = (m_{ij})$ and $N = (n_{ij})$ are three $n \times n$ crisp matrices.

Before formally stating the method we introduce the following category of fuzzy numbers to simplify the concept of including arbitrary triangular fuzzy coefficients with no non-negative restriction.

4.1 “Near zero” triangular fuzzy numbers

A fuzzy number $\tilde{M} = (m, \alpha, \beta)$ is called near-zero fuzzy number, denoted by $\tilde{M} \equiv 0$ if its membership function $\mu_{\tilde{M}}(x)$ satisfies $\mu_{\tilde{M}}(0^+) = \mu_{\tilde{M}}(0^-) = \mu_{\tilde{M}}(0) \neq 0$. Alternatively it specifies that 0 falls in $\text{Supp}(\tilde{M})$, i.e., $m - \alpha < 0 < m + \beta$. An Example of a “near-zero” triangular fuzzy number is (2, 4, 5) which is neither non-negative, nor non-positive.

4.2 Decomposition method to find non-negative solutions of a FFLS with arbitrary coefficient matrix

In this section, we propose a simplified polynomial-time computational method to solve the $n \times n$ FFLS $\tilde{A} \otimes \tilde{x} = \tilde{b}$ under the apriori non-negative constraint on the solution vector. Here, we rewrite the FFLS as:

$$(A, M, N) \otimes (x, y, z) = (B, G, H)$$

Further, we decompose the coefficient matrix into its non-negative, non-positive and near zero fuzzy components, i.e.,

$$\tilde{A} = \tilde{A}_p \oplus \tilde{A}_n \oplus \tilde{A}_0$$

where

$$\tilde{A}_p = \begin{cases} (\tilde{a}_{ij}) & \text{if } \tilde{a}_{ij} \geq 0, \text{ i.e., } a_{ij} - m_{ij} \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

$$\tilde{A}_n = \begin{cases} (\tilde{a}_{ij}) & \text{if } \tilde{a}_{ij} \geq 0, \text{ i.e., } a_{ij} - n_{ij} \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

$$\tilde{A}_0 = \begin{cases} (\tilde{a}_{ij}) & \text{if } \tilde{a}_{ij} \equiv 0, \text{ i.e., } a_{ij} - m_{ij} < 0 < a_{ij} + n_{ij} \\ 0 & \text{otherwise} \end{cases}$$

Thus we can write $\tilde{A} \otimes \tilde{x} = \tilde{b}$ as:

$$(\tilde{A}_p \oplus \tilde{A}_n \oplus \tilde{A}_0) \otimes (x, y, z) = (B, G, H) \tag{1}$$

This system is equivalent to the system

$$(A_p, M_p, N_p) \otimes (x, y, z) \oplus (A_n, M_n, N_n) \otimes (x, y, z) \oplus (A_o, M_o, N_o) \otimes (x, y, z) = (B, G, H)$$

Note that the weakly distributive extended multiplication operation on two fuzzy numbers holds exactly here (Proposition 4.2). Using the arithmetic operations on triangular fuzzy numbers

$$\begin{aligned} \tilde{A}_p \otimes \tilde{x} &= (A_p, M_p, N_p) \otimes (x, y, z) \\ &= (A_p x, M_p x + (A_p - M_p)y, (A_p + N_p)z + N_p x) \end{aligned}$$

Similarly,

$$\begin{aligned} \tilde{A}_n \otimes \tilde{x} &= (A_n x, M_n x - (A_n - M_n)z, N_n x - (A_n + N_n)y) \\ \tilde{A}_0 \otimes \tilde{x} &= (A_o x, M_o x - A_o - M_o)z, N_o x + (A_o + N_o)z \end{aligned}$$

Note that crisp relation $A = A_p + A_n + A_o$ is universally true. Similar relations hold for matrices M and N also. Substituting the above obtained values in the FFLS (1), we obtain the following set of crisp equations:

$$\begin{cases} Ax = B \\ Mx + (A_p - M_p)y + (M_n - A_n + M_o - A_o)z = G \\ Nx - (A_n + N_n)y + (A_p + N_p + A_o + N_o)z = H \end{cases} \tag{2}$$

To solve the systems of equations obtained in (2) employ any method for solving crisp system of linear equations. A direct matrix inversion yields the following results:

$$\begin{cases} x = A^{-1}B \\ y = \alpha_1^{-1}(\gamma_1 - \beta_1(\alpha_2\alpha_1^{-1}\beta_1 - \beta_2)^{-1}(\alpha_2\alpha_1^{-1}\gamma_1 - \gamma_2)) \\ z = (\alpha_2\alpha_1^{-1}\beta_1 - \beta_2)^{-1}(\alpha_2\alpha_1^{-1}\gamma_1 - \gamma_2) \end{cases} \tag{3}$$

where $\alpha_1 = A_p - M_p, \alpha_2 = -A_n - N_n, \beta_1 = M_n - A_n + M_o - A_o, \beta_2 = A_p + N_p + A_o + N_o, \gamma_1 = G - MA^{-1}B$ and $\gamma_2 = H - NA^{-1}B$.

Note that the linear system (2) can be solved by a variety of available classical methods to solve a crisp linear system. The solution of the FFLS can be written as $\tilde{x} = (x, y, z)$ where x, y, z are three $n \times 1$ crisp vectors obtained directly using a matrix inversion rule (3) or by solving the linear system using any classical method.

4.2.1 Feasibility condition

The solution obtained above is considered (strong) feasible if and only if the obtained solution is non-negative and the non-negativity of the spreads is fulfilled, i.e., $x - y \geq 0, y \geq 0$ and $z \geq 0$. If these constraints are violated then the FFLS would be said to have a (weak) infeasible fuzzy solution.

Remark 4.1 The $n \times n$ FFLS $\tilde{A} \otimes \tilde{x} = \tilde{b}$ will have a (strong) feasible fuzzy non-negative solution $\tilde{x} = (x_i, y_i, z_i) \forall i = 1, 2, \dots, n$ if $(x_i, y_i, z_i) \geq 0$, i.e., $x_i - y_i \geq 0, y_i \geq 0, z_i \geq 0 \forall i = 1, 2, \dots, n$; otherwise the solution will be (weak) Infeasible.

Remark 4.2 The proposed method for solving a $n \times n$ FFLS $\tilde{A} \otimes \tilde{x} = \tilde{b}$ with arbitrary coefficient matrix has a polynomial time computational complexity $O(n^3)$, as it relies on the computation of inversion and multiplication of finite number of crisp matrices.

Remark 4.3 Under the constraint of non negativity of the coefficient matrix $\tilde{A}_n, \tilde{A}_0 = \tilde{0}_{n \times n}$ and the solution to FFLS reduces to the solution of the following equivalent crisp system. This crisp linear system has been solved by variety of methods by different researchers, for e.g., (Dehghan and Hashemi 2006a, b; Dehghan et al. 2006; Dehghan and

Hashemi 2007; Nasserri et al. 2008, 2009; Nasserri and Sohrabi 2010; Nasserri and Zahmatkesh 2010).

$$\begin{cases} Ax = B \\ Mx + (A_p - M_p)y = G \\ Nx + (A_p + N_p)z = H \end{cases}$$

Remark 4.4 The $n \times n$ FFLS may be consistent or inconsistent depending upon the consistency of the crisp linear system obtained in (2). Further the FFLS may generate trivial, unique or infinitely many feasible solutions depending upon the solutions obtained by solving the crisp linear system (2).

Proposition 4.1 The necessary condition for the $n \times n$ FFLS $\tilde{A} \otimes \tilde{x} = \tilde{b}$ to have a feasible fuzzy non-negative solution $\tilde{x} = (x_i, y_i, z_i) \forall i = 1, 2, \dots, n$ obtained directly using matrix inversion (3) if the matrices A, α_1, α_2 and $\alpha_2\alpha_1^{-1}\beta_1 - \beta_2$ are non singular.

Proof Straightforward.

Proposition 4.2 The transformation of FFLS (1) into crisp linear system (2) is conditioned upon the exactness of the weakly distributive law of fuzzy multiplication over addition:

$$(\tilde{A}_p \oplus \tilde{A}_n \oplus \tilde{A}_o) \otimes \tilde{x} = (\tilde{A}_p \otimes \tilde{x}) \oplus (\tilde{A}_n \otimes \tilde{x}) \oplus (\tilde{A}_o \otimes \tilde{x})$$

Proof Clearly,

$$\tilde{A} \otimes \tilde{x} = (\tilde{A}_p \oplus \tilde{A}_n \oplus \tilde{A}_o) \otimes \tilde{x}$$

From definition 3.7

$$(\tilde{A} \otimes \tilde{x})_i = \sum_{j=1, \dots, n}^{\oplus} \tilde{a}_{ij} \otimes \tilde{x}_j \quad \forall i = 1, 2, \dots, n$$

By segregating coefficients into three groups we easily get:

$$\sum_{j=1, \dots, n}^{\oplus} \tilde{a}_{ij} \otimes \tilde{x}_j = \sum_{\substack{j=1, \dots, n \\ \tilde{a}_{ij} \geq 0}}^{\oplus} \tilde{a}_{ij} \otimes \tilde{x}_j \oplus \sum_{\substack{j=1, \dots, n \\ \tilde{a}_{ij} \leq 0}}^{\oplus} \tilde{a}_{ij} \otimes \tilde{x}_j \oplus \sum_{\substack{j=1, \dots, n \\ \tilde{a}_{ij} = 0}}^{\oplus} \tilde{a}_{ij} \otimes \tilde{x}_j$$

Clearly:

$$\begin{aligned} (\tilde{A}_p \otimes \tilde{x})_i &= \sum_{\substack{j=1, \dots, n \\ \tilde{a}_{ij} \geq 0}}^{\oplus} \tilde{a}_{ij} \otimes \tilde{x}_j; (\tilde{A}_n \otimes \tilde{x})_i \\ &= \sum_{\substack{j=1, \dots, n \\ \tilde{a}_{ij} \leq 0}}^{\oplus} \tilde{a}_{ij} \otimes \tilde{x}_j; (\tilde{A}_n \otimes \tilde{x})_i = \sum_{\substack{j=1, \dots, n \\ \tilde{a}_{ij} = 0}}^{\oplus} \tilde{a}_{ij} \otimes \tilde{x}_j \end{aligned}$$

which completes the proof

$$\begin{aligned} \tilde{A} \otimes \tilde{x} &= (\tilde{A}_p \oplus \tilde{A}_n \oplus \tilde{A}_o) \otimes \tilde{x} \\ &= (\tilde{A}_p \otimes \tilde{x}) \oplus (\tilde{A}_n \otimes \tilde{x}) \oplus (\tilde{A}_o \otimes \tilde{x}) \end{aligned}$$

Proposition 4.3 The $n \times n$ FFLS $\tilde{A} \otimes \tilde{x} = \tilde{b}$ is called a homogeneous FFLS if \tilde{b} is a null column vector, i.e., with all elements as zero triangular fuzzy numbers. This FFLS will always have a trivial solution. The possibility of infinitely many solutions does not exist.

Proof The homogeneous FFLS $\tilde{A} \otimes \tilde{x} = \tilde{0}$ will always have only a trivial solution $\tilde{x} = \tilde{0} = (0, 0, 0)_{n \times 1}$. The case of infinite solutions is ruled out because for any fuzzy number $\tilde{A}, \tilde{A} \ominus \tilde{A} \neq 0$.

Proposition 4.4 The $n \times n$ FFLS $\tilde{A} \otimes \tilde{x} = \tilde{b}$ with a “near zero” coefficient matrix \tilde{A} , i.e., all the elements as “near zero” triangular fuzzy numbers will always hold the possibility of the existence of infinite feasible solutions.

Proof As $\tilde{A}_n, \tilde{A}_p = \tilde{0}_{n \times n}$, the FFLS in (2) reduces to the following crisp linear system. Clearly this FFLS will either yield infinite solutions or no solution.

$$\begin{cases} A_o x = B \\ M_o x + (M_o - A_o)z = G \\ N_o x + (N_o + A_o)z = H \end{cases}$$

The given system will yield no solution if the following crisp linear system is inconsistent

$$\begin{cases} (M_o - A_o)z = G - M_o A_o^{-1} B \\ (N_o + A_o)z = H - N_o A_o^{-1} B \end{cases}$$

However, if the above crisp linear system is consistent then the FFLS will yield infinite solutions of the form $\tilde{x} = \{(x, y, z) | x - y \geq 0, y \geq 0, z \geq 0\}$. Where x, z are obtained from solving the above crisp system. This is illustrated by example 4.3.

Example 4.1 A ball swings from point A to point B with a constant approximate velocity of $(3, 2, 3) \text{ ms}^{-1}$ for a time of $\tilde{t}_1 = (x_1, y_1, z_1) \text{ s}$ and then retracts with an approximate velocity of $(-2, 1, 1) \text{ ms}^{-1}$ for a time of $\tilde{t}_2 = (x_2, y_2, z_2) \text{ s}$ with a net displacement of $(5, 16, 17) \text{ mts}$. In another instance the ball is pulled backwards with a constant velocity of $(-4, 1, 2) \text{ ms}^{-1}$ for a time of \tilde{t}_1 and pushed forward with a velocity of $(4, 2, 1) \text{ ms}^{-1}$ covering $(-4, 12, 22) \text{ mts}$ from the original point. Considering all other parameters constant find \tilde{t}_1 and \tilde{t}_2 .

Solution The problem can be simply modeled as a FFLS:

$$\begin{aligned} (3, 2, 3) \otimes (x_1, y_1, z_1) \oplus (-2, 1, 1) \otimes (x_2, y_2, z_2) \\ = (5, 16, 17) \end{aligned}$$

$$\begin{aligned} (-4, 1, 2) \otimes (x_1, y_1, z_1) \oplus (4, 2, 1) \otimes (x_2, y_2, z_2) \\ = (-4, 12, 22) \end{aligned}$$

Note that we have apriori knowledge that \tilde{t}_1 and \tilde{t}_2 have to be non-negative fuzzy numbers. Also as all the fuzzy

coefficients are not entirely non-negative, hence this FFLS can not be solved by any of the existing methods. Using the proposed method we obtain the following matrices

$$\tilde{A}_p = \begin{bmatrix} (3, 2, 3) & 0 \\ 0 & (4, 2, 1) \end{bmatrix}, \quad \tilde{A}_o = \tilde{0},$$

$$\tilde{A}_n = \begin{bmatrix} 0 & (-2, 1, 1) \\ (-4, 1, 2) & 0 \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} (5, 16, 17) \\ (-4, 12, 22) \end{bmatrix}$$

Further we obtain the equivalent crisp linear system using (2) as follows:

$$\begin{aligned} 3x_1 - 2x_2 &= 5 \\ -4x_1 + 4x_2 &= -4 \\ 2x_1 + x_2 + y_1 + 3z_2 &= 16 \\ x_1 + 2x_2 + 2y_2 + 5z_1 &= 12 \\ 3x_1 + x_2 - y_2 + 6z_1 &= 17 \\ 2x_1 + x_2 + 2y_1 + 5z_2 &= 22 \end{aligned}$$

On solving this crisp linear system we obtain the following solution $x_1 = 3, x_2 = 2, y_1 = 2, y_2 = 0$ and $z_1 = 1, z_2 = 2$. Hence the solution of the FFLS is $\tilde{t}_1 = (3, 2, 1)$ and $\tilde{t}_2 = (2, 0, 2)$. Both the fuzzy numbers are non-negative. Hence the FFLS is consistent and has a unique (strong) feasible solution.

Example 4.2 Solve the following FFLS for non-negative solutions:

$$\begin{aligned} (4, 6, 1) \otimes (x_1, y_1, z_1) \oplus (4, 2, 4) \otimes (x_2, y_2, z_2) \\ = (24, 26, 31) \\ (3, 2, 1) \otimes (x_1, y_1, z_1) \oplus (2, 3, 1) \otimes (x_2, y_2, z_2) \\ = (14, 18, 13) \end{aligned}$$

Solution Clearly

$$\tilde{A}_p = \begin{bmatrix} 0 & (4, 2, 4) \\ (3, 2, 1) & 0 \end{bmatrix}, \quad \tilde{A}_o = \begin{bmatrix} (4, 6, 1) & 0 \\ 0 & (2, 3, 1) \end{bmatrix},$$

$$\tilde{A}_n = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} (24, 26, 31) \\ (14, 18, 13) \end{bmatrix}$$

Substituting $\tilde{A}_n = 0$ in Eq. (2) we obtain the following reduced set of a crisp linear system

$$\begin{cases} Ax = B \\ Mx + (A_p - M_p)y + (M_o - A_o)z = G \\ Nx + (A + N)z = H \end{cases}$$

which can be easily solved by matrix inversion as follows:

$$\begin{cases} x = A^{-1}B \\ y = (A_p - M_p)^{-1}(G - MA^{-1}B - (M_o - A_o)(A + N^{-1})(H - NA^{-1}B)) \\ z = (A + N^{-1})(H - NA^{-1}B) \end{cases}$$

On solving this system we obtain the following feasible solution: $x_1 = 2, x_2 = 4, y_1 = 1, y_2 = 2$ and $z_1 = 1, z_2 = 1$. Putting the obtained values in $\tilde{x}_1 = (x_1, y_1, z_1)$ and $\tilde{x}_2 =$

(x_1, y_1, z_1) we get, $\tilde{x}_1 = (2, 1, 1)$ and $\tilde{x}_2 = (4, 2, 1)$. Both the fuzzy numbers are non-negative and the solution is thus unique and feasible.

Example 4.3 Solve the FFLS for non-negative solutions:

$$\begin{aligned} (-1, 2, 3) \otimes (x_1, y_1, z_1) \oplus (2, 3, 2) \otimes (x_2, y_2, z_2) \\ = (3, 18, 17) \\ (-4, 1, 10) \otimes (x_1, y_1, z_1) \oplus (-1, 1, 2) \otimes (x_2, y_2, z_2) \\ = (-6, 20, 33) \end{aligned}$$

Solution Using the proposed method we obtain the following matrices

$$\tilde{A}_o = \begin{bmatrix} (-1, 2, 3) & (2, 3, 2) \\ (-4, 1, 10) & (-1, 1, 2) \end{bmatrix}, \quad \tilde{A}_p = \tilde{0}, \quad \tilde{A}_n = \tilde{0},$$

$$\tilde{B} = \begin{bmatrix} (3, 18, 17) \\ (-6, 20, 33) \end{bmatrix}$$

Further we obtain the equivalent crisp linear system using (2) as follows:

$$\begin{aligned} -x_1 + 2x_2 &= 3 \\ -4x_1 - x_2 &= -6 \\ 2x_1 + 3x_2 + 3z_1 + z_2 &= 18 \\ x_1 + x_2 + 5z_1 + 2z_2 &= 20 \\ 3x_1 + 2x_2 + 2z_1 + 4z_2 &= 17 \\ 10x_1 + 2x_2 + 6z_1 + z_2 &= 33 \end{aligned}$$

On solving this crisp linear system we obtain the following solution $x_1 = 1, x_2 = 2,$ and $z_1 = 3, z_2 = 1$. Note that the system is consistent because $(M_o - A_o)^{-1}(G - M_oA_o^{-1}B) = (N_o + A_o)^{-1}(H - N_oA_o^{-1}B)$. Further this FFLS has infinite feasible solutions which may be represented as follows:

$$\tilde{x} = \{\tilde{x}_1 = (1, \alpha, 3); \tilde{x}_2 = (2, \beta, 1) | \alpha \in [0, 1], \beta \in [0, 2]\}$$

4.3 Linear programming method to solve a FFLS with arbitrary solution vector and non-negative coefficients

In this section, we introduce a new technique to solve a FFLS with a non-negative coefficient matrix and no restrictions on the solution vector, i.e., the proposed method does not assume any apriori constraint of non negativity on the unknown quantity and generates the solution irrespective of its sign. The steps of the proposed method are as follows:

Step 1 Substituting $\tilde{A} = (\tilde{a}_{ij})_{n \times n}, \tilde{x} = (\tilde{x}_j)_{n \times 1}$ and $\tilde{B} = (\tilde{B}_i)_{n \times 1}$ the $n \times n$ FFLS may be written as:

$$\bigoplus_{j=1, \dots, n} \tilde{a}_{ij} \otimes \tilde{x}_j = \tilde{B}_i \quad \forall i = 1, 2, \dots, n$$

Step 2 If $\tilde{a}_{ij}, \tilde{x}_j$ and \tilde{B}_i are represented by triangular fuzzy numbers $(a_{ij}, b_{ij}, c_{ij}) \geq 0, (x_j, y_j, z_j)$ and (B_i, g_i, h_i) then the FFLS obtained in Step 1, may be written as:

$$\sum_{j=1, \dots, n}^{\oplus} (a_{ij}, b_{ij}, c_{ij}) \otimes (x_j, y_j, z_j) = (B_i, g_i, h_i) \quad \forall i = 1, 2, \dots, n$$

Step 3 By Assuming $(a_{ij}, b_{ij}, c_{ij}) \otimes (x_j, y_j, z_j) = (f_{ij}, p_{ij}, q_{ij})$, the FFLS obtained in step 2 may be written as:

$$\sum_{j=1, \dots, n}^{\oplus} (f_{ij}, p_{ij}, q_{ij}) = (B_i, g_i, h_i) \quad \forall i = 1, 2, \dots, n$$

using Sect. 3.1.

$$\begin{aligned} (f_{ij}, p_{ij}, q_{ij}) &= (a_{ij}, b_{ij}, c_{ij}) \otimes (x_j, y_j, z_j) \\ f_{ij} &= a_{ij}x_j \\ p_{ij} &= f_{ij} - \min((a_{ij} - b_{ij})(x_j - y_j), (a_{ij} + c_{ij})(x_j - y_j)) \\ p_{ij} &= f_{ij} - \frac{2a_{ij} - b_{ij} + c_{ij}}{2}(x_j - y_j) + \left| \frac{b_{ij} + c_{ij}}{2} \right| |x_j - y_j| \\ q_{ij} &= \max((a_{ij} - b_{ij})(x_j + z_j), (a_{ij} + c_{ij})(x_j + z_j)) - p_{ij} \\ q_{ij} &= -p_{ij} + \frac{2a_{ij} - b_{ij} + c_{ij}}{2}(x_j + z_j) + \left| \frac{b_{ij} + c_{ij}}{2} \right| |x_j + z_j| \\ \forall i &= 1, 2, \dots, n, \quad \forall j = 1, 2, \dots, n \end{aligned}$$

For notational convenience we substitute $x_j - y_j = L_j$ and $x_j + z_j = R_j \forall j = 1, 2, \dots, n$ where $x_j, y_j, z_j, w_j, L_j, R_j \in R$.

Step 4 Substitute f_{ij}, p_{ij}, q_{ij} in the FFLS obtained in step 3 to obtain the following semi-linear system of equations:

For $i = 1, 2, \dots, n$.

$$\sum_{j=1}^n f_{ij} = B_i \Rightarrow \sum_{j=1}^n a_{ij}x_j = B_i \tag{4}$$

Similarly,

$$\sum_{j=1}^n p_{ij} = g_i \Rightarrow \sum_{j=1}^n \frac{2a_{ij} - b_{ij} + c_{ij}}{2}L_j - \left(\frac{b_{ij} + c_{ij}}{2} \right) |L_j| = B_i - g_i \tag{5}$$

$$\begin{aligned} \sum_{j=1}^n q_{ij} = k_i &\Rightarrow \sum_{j=1}^n \frac{2a_{ij} - b_{ij} + c_{ij}}{2}R_j + \left(\frac{b_{ij} + c_{ij}}{2} \right) |R_j| \\ &= B_i + h_i \end{aligned} \tag{6}$$

Step 5 Substitute $L_j = L'_j - L''_j$ and $|L_j| = L'_j + L''_j$, where $L'_j = \begin{cases} L_j & \text{if } L_j > 0 \\ 0 & \text{otherwise} \end{cases}$ and $L''_j = \begin{cases} -L_j & \text{if } L_j < 0 \\ 0 & \text{otherwise} \end{cases}$. Similarly, $R_j = R'_j - R''_j$ and $|R_j| = R'_j + R''_j$. Therefore the equations, obtained in Step 4, may be written as for $i = 1, 2, \dots, n$

$$\sum_{j=1}^n a_{ij}x_j = B_i \tag{7}$$

$$\begin{aligned} \sum_{j=1}^n \frac{2a_{ij} - b_{ij} + c_{ij}}{2} (L'_j - L''_j) - \left(\frac{b_{ij} + c_{ij}}{2} \right) (L'_j + L''_j) \\ = B_i - g_i \end{aligned} \tag{8}$$

$$\begin{aligned} \sum_{j=1}^n \frac{2a_{ij} - b_{ij} + c_{ij}}{2} (R'_j - R''_j) + \left(\frac{b_{ij} + c_{ij}}{2} \right) (R'_j + R''_j) \\ = B_i + h_i \end{aligned} \tag{9}$$

$$L'_j, L''_j, R'_j, R''_j \geq 0 \quad \forall j = 1, 2, \dots, n$$

Step 6 To solve the original FFLS we construct the following linear programming problem (LPP)

$$\text{Minimize } \bar{Z} = \sum_{i=1}^n Z_i + \sum_{i=1}^n Z'_i + \sum_{i=1}^n Z''_i$$

subject to

$$\sum_{j=1}^n a_{ij}x_j + Z_i = B_i \quad \forall i = 1, 2, \dots, n$$

$$\begin{aligned} \sum_{j=1}^n ((a_{ij} - b_{ij})L'_j - (a_{ij} + c_{ij})L''_j) + Z'_i = B_i - g_i \\ \forall i = 1, 2, \dots, n \end{aligned}$$

$$\begin{aligned} \sum_{j=1}^n ((a_{ij} + c_{ij})R'_j - (a_{ij} - b_{ij})R''_j) + Z''_i = B_i + h_i \\ \forall i = 1, 2, \dots, n \end{aligned}$$

$$x_j - L'_j + L''_j \geq 0 \quad \forall j = 1, 2, \dots, n$$

$$R'_j - R''_j - x_j \geq 0 \quad \forall j = 1, 2, \dots, n$$

$$L'_j, L''_j, R'_j, R''_j, Z_j, Z'_j, Z''_j \geq 0 \quad \forall j = 1, 2, \dots, n$$

For obtaining the equivalent matrix representation of the above LPP, we define the following matrices

Define U as a unit column vector with all the entries equal to 1 and order $n \times 1$. We may obtain the equivalent expressions for other matrices as follows:

$$\begin{aligned} Z &= (Z_i)_{n \times 1}, Z' = (Z'_i)_{n \times 1}, Z'' = (Z''_i)_{n \times 1}, L'(L')_{n \times 1}, \\ L''(L'')_{n \times 1}, R'(R')_{n \times 1} \\ R'' &= (R''_i)_{n \times 1}, A = (a_{ij})_{n \times n}, M = (b_{ij})_{n \times n}, N = (c_{ij})_{n \times n}, \\ B &= (B_i)_{n \times 1}, G = (g_i)_{n \times 1}, \\ H &= (h_i)_{n \times 1} \end{aligned}$$

Thus the LPP can be rewritten as follows:

$$\begin{aligned} \text{Minimize } \bar{Z} &= U^T Z + U^T Z' + U^T Z'' \\ \text{subject to} \end{aligned}$$

$$\begin{aligned} Ax + Z &= B \\ (A - M)L' + (A + N)L'' + Z' &= B - G \\ (A + N)R' - (A - M)R'' + Z'' &= B + H \\ x - L' + L'' &\geq 0 \quad R' - R'' - x \geq 0 \\ L', L'', R', R'', Z, Z', Z'' &\geq 0 \end{aligned}$$

Step 7 Solve the LPP obtained in step 6 and substitute the values of x, L', L'', R', R'' obtained in step 6 in $\tilde{x}_j =$

$(x_j, x_j - L'_j + L''_j, R'_j - R''_j - x_j) \forall j = 1, 2, \dots, n$ to find the solution of the above FFLS.

Remark 4.5 The FFLS will be consistent, i.e., it will generate a solution if and only if the LPP is consistent. Further, the necessary condition for the above LPP to generate a feasible (strong) solution (if it is consistent) is that the value of the objective function must be zero $\bar{Z} = 0$ otherwise the solution will be termed as an (weak) infeasible solution. Thus, the sufficient conditions for the FFLS to have a feasible (strong) fuzzy solution are $L' \cdot L''^T = R' \cdot R''^T = \bar{Z} = 0$.

Remark 4.6 The nature of the solutions of the FFLS depends upon the nature of the solutions of the LPP which may be unique, trivial or infinitely many, i.e., the FFLS may yield no solution, unique solution or infinitely many solutions depending upon the nature of the LPP that is constructed.

Remark 4.7 Under the constraint of the non negativity of the solution vector $L'' = R'' = 0$, substituting these into the LPP obtained in step 6 we obtain a simple crisp linear system (similar to as discussed in Remark 4.3) which has been solved by variety of classical methods by several researchers.

Example 4.4 Solve the following FFLS

$$\begin{bmatrix} (5, 1, 1) & (6, 1, 2) \\ (3, 1, 4) & (5, 2, 1) \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \end{bmatrix} = \begin{bmatrix} (8, 22, 12) \\ (9, 31, 7) \end{bmatrix}$$

where, \tilde{x}_1, \tilde{x}_2 are arbitrary triangular fuzzy numbers

Solution The solution of the chosen FFLS may be obtained by using the steps of the above method and converting the FFLS into the LPP as follows:

Minimize $Z = (Z_1 + Z_2 + Z'_1 + Z'_2 + Z''_1 + Z''_2)$
subject to

$$\begin{aligned} 5x_1 + 6x_2 + Z_1 &= 8 \\ 3x_1 + 5x_2 + Z_2 &= 9 \\ 4L'_1 - 6L''_1 + 5L'_2 - 8L''_2 + Z'_1 &= -14 \\ 2L'_1 - 7L''_1 + 3L'_2 - 6L''_2 + Z'_2 &= -22 \\ 6R'_1 - 4R''_1 + 8R'_2 - 5R''_2 + Z''_1 &= 20 \\ 7R'_1 - 2R''_1 + 6R'_2 - 3R''_2 + Z''_2 &= 16 \\ x_1 - L'_1 + L''_1 &\geq 0 \\ x_2 - L'_2 + L''_2 &\geq 0 \\ R'_1 - R''_1 - x_1 &\geq 0 \\ R'_2 - R''_2 - x_2 &\geq 0 \\ Z_1, Z_2, Z'_1, Z'_2, Z''_1, Z''_2, L'_1, L'_2, L''_1, L''_2, R'_1, R'_2, R''_1, R''_2 &\geq 0 \end{aligned}$$

On solving the above LPP, the following solution is obtained:

$$\begin{aligned} x_1 = -2, x_2 = 3, L'_1 = 0, L''_1 = 4, L'_2 = 2, L''_2 = 0, R'_1 = 0, \\ R''_1 = 1, R'_2 = 3, R''_2 = 0, \\ Z_1 = Z_2 = Z'_1 = Z'_2 = Z''_1 = Z''_2 = 0 \end{aligned}$$

Clearly the FFLS meets the necessary and sufficient conditions for the possibility of a feasible fuzzy solution. Substituting the values obtained in the following equation yields the solution of the FFLS. $\tilde{x}_j = (x_j, x_j - L'_j + L''_j, R'_j - R''_j - x_j) \forall j = 1, 2$, we obtain the following solution:

$$\tilde{x}_1 = (-2, 2, 1) \quad \text{and} \quad \tilde{x}_2 = (3, 1, 0)$$

Note that $\tilde{x}_1 \leq 0$ and $\tilde{x}_2 \geq 0$. This clearly justifies that the existing methods can not be used to solve this FFLS because all the methods presume the non negativity of the solution vector.

4.4 Fuzzy Cramer’s rule for solving an arbitrary FFLS with no joint restrictions on the sign of coefficients and the sign of variables

In this section, we propose a polynomial time algorithm to solve a FFLS where the coefficient matrix and the solution vector are unconstrained on the dimension of sign; however, we exclude the possibility of “near zero” fuzzy numbers in the solution vector, i.e., the only constraint that is now imposed on an arbitrary FFLS is that the solution vector cannot include elements which have zero in their support. The reason for this restriction lies in the structure of arithmetic operations on fuzzy numbers that is explained in Proposition 4.6.

The FFLS is written as:

$$(A, M, N) \otimes (x, y, z) = (B, G, H)$$

Now we decompose the coefficient matrix into its non-negative, non-positive and near zero fuzzy components and the solution vector into its non-negative and non-positive components, i.e.,

$$\tilde{A} = \tilde{A}_p \oplus \tilde{A}_n \oplus \tilde{A}_o \quad \text{and} \quad \tilde{x} = \tilde{x}_p \oplus \tilde{x}_n \quad \text{where}$$

$$\tilde{x}_p = \begin{cases} (\tilde{x}_i) & \text{if } \tilde{x}_i \geq 0, \text{ i.e., } x_i - y_i \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

$$\tilde{x}_n = \begin{cases} (\tilde{x}_i) & \text{if } \tilde{x}_i \leq 0, \text{ i.e., } x_i + z_i \leq 0 \\ 0 & \text{otherwise} \end{cases}$$

Thus we can write $\tilde{A} \otimes \tilde{x} = \tilde{b}$ as:

$$\begin{aligned} \tilde{A}_p \otimes \tilde{x}_p \oplus \tilde{A}_n \otimes \tilde{x}_p \oplus \tilde{A}_o \otimes \tilde{x}_p \oplus \tilde{A}_p \otimes \tilde{x}_n \oplus \tilde{A}_n \otimes \tilde{x}_n \oplus \tilde{A}_o \otimes \tilde{x}_n \\ \tilde{x}_n = \tilde{b} \end{aligned} \tag{10}$$

which is equivalent to:

$$\begin{aligned} &(A_p, M_p, N_p) \otimes (x_p, y_p, z_p) \oplus (A_n, M_n, N_n) \otimes (x_p, y_p, z_p) \\ &\oplus (A_o, M_o, N_o) \otimes (x_p, y_p, z_p) \oplus (A_p, M_p, N_p) \otimes (x_n, y_n, z_n) \\ &\oplus (A_n, M_n, N_n) \otimes (x_n, y_n, z_n) \oplus (A_o, M_o, N_o) \\ &\otimes (x_n, y_n, z_n) = (B, G, H) \end{aligned}$$

Note that the weakly distributive extended multiplication operation on two fuzzy numbers holds exactly here (Proposition 4.2). Using the arithmetic operations on triangular fuzzy numbers

$$\begin{aligned} \tilde{A}_p \otimes \tilde{x}_p &= (A_p x_p, M_p x_p + (A_p - M_p)y_p, (A_p + N_p)z_p + N_p x_p) \\ \tilde{A}_n \otimes \tilde{x}_p &= (A_n x_p, M_n x_p - (A_n - M_n)z_p, N_n x_p - (A_n + N_n)y_p) \\ \tilde{A}_o \otimes \tilde{x}_p &= (A_o x_p, M_o x_p - (A_o - M_o)z_p, N_o x_p + (A_o + N_o)z_p) \\ \tilde{A}_p \otimes \tilde{x}_n &= (A_p x_n, -N_p x_n + (A_p + N_p)y_n, (A_p - M_p)z_n - M_p x_n) \\ \tilde{A}_n \otimes \tilde{x}_n &= (A_n x_n, -N_n x_n - (A_n + N_n)z_n, -M_n x_n - (A_n - M_n)y_n) \\ \tilde{A}_o \otimes \tilde{x}_n &= (A_o x_n, -N_o x_n + (A_o + N_o)y_n, -M_o x_n - (A_o - M_o)y_n) \end{aligned}$$

Note that crisp relation $A = A_p + A_n + A_o$ is universally true. Similar relations hold for matrices M and N also. Further $x = x_p + x_n, y = y_p + y_n$ and $z = z_p + z_n$.

For convenience we create two step functions: $u(a) = \begin{cases} 1 & \text{if } a \geq 0 \\ 0 & \text{otherwise} \end{cases}; v(a) = \begin{cases} 1 & \text{if } a < 0; \\ 0 & \text{otherwise} \end{cases}$

We define two diagonal matrices: $U = (u(x_i))_{n \times n}$, i.e., $U = (u_{ij}) = \begin{cases} u(x_i) & \forall i = j \\ 0 & \forall i \neq j \end{cases}$. Similarly, $V = (v(x_i))_{n \times n}$ such that $x_p = Ux, y_p = Uy, z_p = Uz, x_n = Vx, y_n = Vy$, and $z_n = Vz$.

This on substituting in (10) yields following set of crisp equations:

$$\begin{cases} Ax = B \\ Q_{11}y + Q_{12}z = G - A_1x \\ Q_{21}y + Q_{22}z = H - A_2x \end{cases} \quad (11)$$

where

$$\begin{aligned} A_1 &= MU - NV, A_2 = -MV + NU \\ Q_{11} &= (A_p - M_p)U + (A_p + N_p + A_o + N_o)V \\ Q_{12} &= (M_n - A_n + M_o - A_o)U - (A_n + N_n)V \\ Q_{21} &= -(A_n + N_n)U - (A_n - M_n + A_o - M_o)V \\ Q_{22} &= (A_p + N_p + A_o + N_o)U + (A_p - M_p)V \end{aligned}$$

To solve the systems of equations obtained in (2) we employ cramer’s rule to solve the block linear system.

$$x_i = \frac{\det(A^i)}{\det(A)}, \quad i = 1, 2, \dots, n$$

where A^i is the matrix obtained by replacing the i th column of A by B .

$$y_i = \frac{\det(\bar{Q}^i)}{\det(Q)}, \quad i = 1, 2, \dots, n,$$

where $\det(Q) = \det \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix} = \det(Q_{11} - Q_{12}Q_{22}^{-1}Q_{21}) \det(Q_{22})$. (Note that evaluating the determinant of the $2n \times 2n$ block matrix significantly reduces the number of computational operations by evaluating the determinant of two $n \times n$ matrices with two matrix multiplication and one inverse operation. However, the computational complexities in both cases remain same). and \bar{Q}^i is the matrix obtained by replacing the i th column of Q_{11} with $G - A_1x$ and Q_{21} with $H - A_2x$.

Similarly

$$z_i = \frac{\det(\bar{Q}^i)}{\det(Q)}, \quad i = 1, 2, \dots, n,$$

and \bar{Q}^i is the matrix obtained by replacing the i th column of Q_{12} with $G - A_1x$ and Q_{22} with $H - A_2x$.

Note that the $3n \times 3n$ linear system (11) can be solved by a variety of available classical methods to solve a crisp linear system. The solution of the FFLS can be written as $\tilde{x} = (x, y, z)$ where x, y, z are three $n \times 1$ crisp vectors obtained directly using the fuzzy cramer’s rule or by solving the linear system using any classical method.

Remark 4.8 The $n \times n$ FFLS $\tilde{A} \otimes \tilde{x} = \tilde{b}$ will have a (strong) feasible fuzzy solution $\tilde{x} = (x_i, y_i, z_i) \forall i = 1, 2, \dots, n$ if $y_i \geq 0, z_i \geq 0$ and $0 \notin \text{Supp}(\tilde{x}) \forall i = 1, 2, \dots, n$; otherwise the solution will be (weak) Infeasible.

Remark 4.9 Under the constraint of non negativity of the solution vector $V = 0_{n \times n}$ and the method of solving the FFLS reduces to the method described in Sect. 4.2.

Remark 4.10 Under the constraint of negligibility of spreads (or small spreads) with respect to mean of a fuzzy number, the product of spreads in the extended multiplication operation is neglected, for example, See (Dehghan and Hashemi 2006a, 2006b; Dehghan et al. 2006; Dehghan and Hashemi 2007; Nasseri et al. 2008, 2009; Nasseri and Sohrabi 2010; Nasseri and Zahmatkesh 2010). However this further aggravates the encumbrance of restrictions applied in solving an FFLS and thus is removed in the methods described in this paper. Under this constraint of negligibility of spreads Q_{ij} takes the following form: $Q_{11} = A_p U + (A_p + A_o)V, Q_{12} = (-A_n - A_o)U - A_n V, Q_{21} = -A_n U - (A_n + A_o)V$ and $Q_{22} = (A_p + A_o)U + A_p V$.

Proposition 4.5 The necessary condition for the $n \times n$ FFLS $\tilde{A} \otimes \tilde{x} = \tilde{b}$ to have a unique and feasible fuzzy solution $\tilde{x} = (x_i, y_i, z_i) \forall i = 1, 2, \dots, n$ obtained directly using cramer’s rule (4.4) is that the determinant of matrices A, A^i, Q, \bar{Q}^i and \bar{Q}^i should not be zero.

Proof Straightforward.

Proposition 4.6 *Allowing the possibility of “near-zero” fuzzy numbers in the solution vector, combinatorially explodes the solution space with computationally complexity growing from polynomial time $O(n^3)$ (P-class) to exponential time $O(2^n n^3)$ (i.e., an unconstrained FFLS is an NP problem).*

Proof The modified fuzzy cramer’s rule has a straightforward computational complexity of $O(n^3)$ with finite number of matrix multiplications, inverse and determinant operations. However, this polynomial-time complexity is a consequence of the constraint on the solutions that propels the exclusion “near-zero” fuzzy numbers in the solution vector. The reason lies in the structure of arithmetic operations on fuzzy numbers primarily in the multiplication between two “near-zero” fuzzy numbers, i.e., if the solution vector is unconstrained then the multiplication operation $\tilde{A}_o \otimes \tilde{x}_o$ results in indeterminacy giving rise to a non-polynomial time complexity of the algorithm for solving the FFLS: $\tilde{A}_o \otimes \tilde{x}_o = (A_o x_o, A_o x_o - \min((A_o - M_o)(x_o + z_o), (A_o + N_o)(x_o - y_o)), A_o x_o + \max((A_o - M_o)(x_o - y_o), (A_o + N_o)(x_o + z_o)))$. The splitting of this system into linear tuples without min–max constraints is infeasible considering the entropy associated in the operation. Note that the operation is totally uncertain as we have no apriori information of $(x_o + z_o)$ or $(x_o - y_o)$ which is paradoxically the objective of the problem. Consequently this can be solved only by decomposing the system into 2^n FFLS’s (of order $n \times n$) with each FFLS constrained with an additional assumption (Ex. $(A_o - M_o)(x_o + z_o) < (A_o + N_o)(x_o - y_o)$) leading to an overall exponential complexity of $O(2^n n^3)$. The verifier algorithm is evidently polynomial time resulting to the conclusion that solving an unconstrained FFLS is an NP-problem. Commenting on the NP-completeness and NP-hardness is left as a potential topic of future research.

Example 4.5 Solve the following FFLS

$$\begin{aligned} (2, 1, 1) \otimes (x_1, y_1, z_1) \oplus (-1, 1, 0) \otimes (x_2, y_2, z_2) &= (6, 4, 11) \\ (-5, 1, 4) \otimes (x_1, y_1, z_1) \oplus (9, 3, 6) \otimes (x_2, y_2, z_2) &= (-28, 50, 21) \end{aligned}$$

where, \tilde{x}_1, \tilde{x}_2 are arbitrary triangular fuzzy numbers

Solution As the coefficients and the solution vector both are arbitrary, hence the given FFLS cannot be solved by any of the existing methods. Using the proposed method following solution is obtained:

$$x_1 = \frac{\det \begin{pmatrix} 6 & -1 \\ -28 & 9 \end{pmatrix}}{\det \begin{pmatrix} 2 & -1 \\ -5 & 9 \end{pmatrix}} = 2; x_2 = \frac{\det \begin{pmatrix} 2 & 6 \\ -5 & -28 \end{pmatrix}}{\det \begin{pmatrix} 2 & -1 \\ -5 & 9 \end{pmatrix}} = -2$$

Hence following matrices are constructed:

$$U = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, V = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, A_1 = \begin{bmatrix} 1 & 0 \\ 1 & 6 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 1 & -1 \\ 4 & -3 \end{bmatrix}, Q_{11} = \begin{bmatrix} 1 & 0 \\ 0 & 15 \end{bmatrix}, Q_{12} = \begin{bmatrix} 0 & 1 \\ 6 & 0 \end{bmatrix}$$

$$Q_{21} = \begin{bmatrix} 0 & 2 \\ 1 & 0 \end{bmatrix}, Q_{22} = \begin{bmatrix} 3 & 0 \\ 0 & 6 \end{bmatrix}$$

$$\begin{aligned} \det(Q) &= \det \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix} \\ &= \det(Q_{11} - Q_{12}Q_{22}^{-1}Q_{21}) \det(Q_{22}) = 165 \end{aligned}$$

$$\det(\bar{Q}^1) = 165, \det(\bar{Q}^2) = 330, \det(\bar{Q}^1) = 165, \det(\bar{Q}^2) = 165$$

This gives: $y_1 = 1, y_2 = 2, z_1 = 1, z_2 = 1$ i.e.

$$\tilde{x}_1 = (2, 1, 1) \text{ and } \tilde{x}_2 = (-2, 2, 1)$$

Note that the FFLS has a unique and a feasible (strong) solution and $\tilde{x}_1 \geq 0; \tilde{x}_2 \leq 0$.

Example 4.6 Solve the following FFLS taken from (Dehghan et al. 2006)

$$\begin{aligned} (5, 1, 1) \otimes (x_1, y_1, z_1) \oplus (6, 1, 2) \otimes (x_2, y_2, z_2) &= (50, 10, 17) \\ (7, 1, 0) \otimes (x_1, y_1, z_1) \oplus (4, 0, 1) \otimes (x_2, y_2, z_2) &= (48, 5, 7) \end{aligned}$$

Solution The method described in (Dehghan et al. 2006) constrains the coefficient and solutions to be positive fuzzy numbers. However, as the methods described in the paper do not assume positivity constraint on the parameters, hence any of the above methods (4.2, 4.3, 4.4) can be applied to solve such a kind of problem. Further (Dehghan et al. 2006) unexplainably constrains the spreads of the fuzzy solution to be negligible as compared to the mean. Under this assumption the method 4.4 gives following

$$\text{matrices: } A = \begin{bmatrix} 5 & 6 \\ 7 & 4 \end{bmatrix}, U = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, V = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, A_1 =$$

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}, A_2 = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix}, Q_{11} = \begin{bmatrix} 5 & 6 \\ 7 & 4 \end{bmatrix}, Q_{22} = \begin{bmatrix} 5 & 6 \\ 7 & 4 \end{bmatrix}$$

and $Q_{12} = 0, Q_{21} = 0$. On solving the crisp system (Eq. 11) following solution is generated $x_1 = 4, x_2 = 5, y_1 = 1/11, y_2 = 1/11, z_1 = 0, z_2 = 1/2$ which is exactly same as that of (Dehghan et al. 2006), i.e., $\tilde{x}_1 = (4, 1/11, 0)$ and $\tilde{x}_2 = (5, 1/11, 1/2)$. This establishes the equivalence of new techniques with earlier proposed paradigms.

5 Discussion and results

The paper studies the domain of FFLS that is unconstrained on the dimension of sign. Three new computational approaches toward solving a FFLS in polynomial time (i.e.

P-class FFLS) are discussed which have profound implications. The category of “near-zero” fuzzy numbers which have not been taken care of in the earlier research is conceptualized and their impact in terms of computational complexity of the algorithms for solving a FFLS is described. The manuscript progresses in three stages. First, the signed decomposition method (4.2) removes the sign restriction on the coefficient matrix, as a result of which all kinds of FFLS which have non-negative, non-positive or near-zero fuzzy coefficients can be solved easily in polynomial time complexity. The linear optimization problem (4.3) removes all apriori sign restrictions on the solution vector which is a tough criterion. Finally the fuzzy cramer’s rule (4.4) removes joint restrictions on the coefficients as well as the solution vector; however, confining the solutions to exclude near zero fuzzy numbers, the reasons of which are explained earlier. The implications can be summarized as follows:

1. The polynomial time algorithms significantly advance the theory of FFLS by removing sign constraints on the parameters in gradual steps (firstly on the coefficient matrix, then on the solution vector and then jointly on all parameters).
2. As a result the implementation scope of FFLS is widely increased as the number of restrictions has been greatly reduced.
3. Using the proposed methods it is easy to check whether the FFLS is consistent or inconsistent and generates feasible or infeasible solutions. Further interestingly we see that a FFLS can have unique, trivial and infinitely many solutions- a concept that has not been studied so far.

Six new numerical examples have also been solved to demonstrate the usability of the proposed algorithms. Example 4.1 establishes real-life usage of a FFLS and also demonstrates the need of methods for solving an arbitrary coefficient FFLS and that apriori knowledge of sign of solution vector can greatly reduce the computational effort. Examples 4.3 demonstrates the case where solving a FFLS results in infinitely feasible solutions. Also the examples 4.1–4.5 cannot be solved by any of the existing methods (Abbasbandy et al. 2005, 2006; Abbasbandy and Jafarian 2006; Allahviranloo 2004a, b, 2005; Allahviranloo and Ghanbari 2010; Allahviranloo et al. 2008; Buckley and Qu 1991; Dehghan and Hashemi 2006; Dehghan and Hashemi 2006a, b; Dehghan et al. 2006, 2007; Gao 2009; Kumar et al. 2010; Liu 2010; Mosleh and Otadi 2010; Mosleh et al. 2009; Nasser et al. 2008, 2009; Nasser et al. 2009; Nasser and Sohrabi 2010; Nasser and Zahmatkesh 2010; Sun and Guo 2009; Yin and Wang 2009) due to varied number of imposed restrictions used in the existing methods. Example 4.6 illustrates clearly the backward compatibility of the newly devised algorithms.

6 Conclusion

In this paper, three new computational methods for solving a FFLS are presented. The proposed methods are easy to understand and apply in real life situations. The methods are illustrated with the help of numerical examples. Further the methods tend to remove several constraints that are usually considered and hence widen the scope of fuzzy linear systems. The formulation of such simple conceptual and numerical methods will erect several possible paths of employability of fuzzy linear systems in engineering and scientific problems. Future work to solve a FFLS with less computational effort involving no joint restrictions on the coefficients and the solution and to demonstrate the practical usability of fuzzy linear systems in science unavoidably persists.

Acknowledgments The authors would like to thank to the “Editor-in-Chief” and the anonymous referees for various suggestions which have led to an improvement in both the quality and clarity of the paper. I, Dr. Amit Kumar, want to acknowledge the innocent inner blessings of Mehar. I believe that Mehar is an angel for me and without Mehar’s blessings it was not possible to think of the idea proposed in this paper. Mehar is a lovely daughter of Parmpreet Kaur (Research Scholar under my supervision).

References

- Abbasbandy S, Jafarian A (2006) Steepest descent method for system of fuzzy linear equations. *Appl Math Comput* 175:823–833
- Abbasbandy S, Jafarian A, Ezzati R (2005) Conjugate gradient method for fuzzy symmetric positive-definite system of linear equations. *Appl Math Comput* 171:1184–1191
- Abbasbandy S, Ezzati R, Jafarian A (2006) LU decomposition method for solving fuzzy system of linear equations. *Appl Math Comput* 172:633–643
- Allahviranloo T (2004a) Numerical methods for fuzzy system of linear equations. *Appl Math Comput* 155:493–502
- Allahviranloo T (2004b) Successive overrelaxation iterative method for fuzzy system of linear equations. *Appl Math Comput* 162:189–196
- Allahviranloo T (2005) The adomian decomposition method for fuzzy system of linear equations. *Appl Math Comput* 163:553–563
- Allahviranloo T, Ghanbari M (2010) Solving fuzzy linear system by homotopy perturbation method. *Int J Comput Cogn* 8:24–30
- Allahviranloo T, Mikarilvand N, Kiani NA, Shabestari RH (2008) Signed decomposition of fully fuzzy linear systems. *Appl Appl Math* 3:77–88
- Buckley JJ, Qu Y (1991) Solving systems of linear fuzzy equations. *Fuzzy Sets Syst* 43:33–43
- Dehghan M, Hashemi B (2006a) Solution of the fully fuzzy linear system using the decomposition procedure. *Appl Math Comput* 182:1568–1580
- Dehghan M, Hashemi B (2006b) Iterative solution of fuzzy linear systems. *Appl Math Comput* 175:645–674
- Dehghan M, Hashemi B, Ghatee M (2006) Computational methods for solving fully fuzzy linear systems. *Appl Math Comput* 179:328–343
- Dehghan M, Hashemi B, Ghatee M (2007) Solution of the fully fuzzy linear system using iterative techniques. *Chaos Solitons Fractals* 34:316–336

- Dubois D, Prade H (1980) Fuzzy sets and systems: theory and applications. Academic Press, New York
- Friedman M, Ming M, Kandel A (1998) Fuzzy linear systems. *Fuzzy Sets Syst* 96:201–209
- Friedman M, Ming M, Kandel A (2000) Duality in fuzzy linear systems. *Fuzzy Sets Syst* 109:55–58
- Gao J (2009) A unified iterative scheme for solving fully fuzzy linear system. *Global Congr Intell Syst* 1:431–435
- Kaufmann A, Gupta MM (1985) Introduction to fuzzy arithmetic: theory and applications. Van Nostrand Reinhold, New York
- Kumar A, Neetu Bansal A (2010) A new method to solve fully fuzzy linear system with trapezoidal fuzzy numbers. *Can J Sci Eng Math* 1:45–56
- Liu HK (2010) On the solution of fully fuzzy linear systems. *World Acad Sci Eng Technol Math Comput Sci* 6:29–33
- Mosleh M, Otadi M (2010) Regular splitting method for approximating linear system of fuzzy equations. *Int J Contemp Math Sci* 5:263–274
- Mosleh M, Abbasbandy S, Otadi M (2007) Full fuzzy linear systems of the form $Ax + b = Cx + d$. In: First joint congress on fuzzy and intelligent systems, Ferdowsi University of Mashhad, Iran. <http://www.docstoc.com/docs/6566376/Full-fuzzy-linear-systems-of-the-form-Axb=Cxd>
- Mosleh M, Otadi M, Khanmirzaie A (2009) Decomposition method for solving fully fuzzy linear systems. *Iran J Optim* 1:188–198
- Muzzioli S, Reynaerts H (2006) Fuzzy linear systems of the form $A_1x + b_1 = A_2x + b_2$. *Fuzzy Sets Syst* 157:939–951
- Nasseri SH, Sohrabi M (2010) Gram–Schmidt approach for linear system of equations with fuzzy parameters. *J Math Comput Sci* 1:80–89
- Nasseri SH, Zahmatkesh F (2010) Huang method for solving fully fuzzy linear system of equations. *J Math Comput Sci* 1:1–5
- Nasseri SH, Sohrabi M, Ardil E (2008) Solving fully fuzzy linear systems by use of a certain decomposition of the coefficient matrix. *Int J Comput Math Sci* 2:140142
- Nasseri SH, Matinfar M, Kheiri Z (2009) Greville's method for the fully fuzzy linear system of equations. *Adv Fuzzy Sets Syst* 4:301–311
- Sun X, Guo S (2009) Solution to general fuzzy linear system and its necessary and sufficient condition. *Fuzzy Inf Eng* 3:317–327
- Yin JF, Wang K (2009) Splitting iterative methods for fuzzy system of linear equations. *Comput Math Model* 20:326–335
- Zadeh LA (1965) Fuzzy sets. *Inf Control* 8:338–353