Thai Journal of Mathematics
Volume 21 Number 1 (2023)
Pages 29–37

http://thaijmath.in.cmu.ac.th

ISSN 1686-0209

A New Method Based on Jacobi Iteration for Fuzzy Linear Systems

Zhen Huang[1], Zhuo Chen[1], Shijun Zhang[2], Shiheng Wang[2] and Ke Wang[1,*]

[1] *Department of Mathematics, Shanghai University, Shanghai 200444, P.R. China*
e-mail : kwang@math.ecnu.edu.cn (K. Wang)

[2] *Nanyang Vocational College of Agriculture, Nanyang 473000, P.R. China*

Abstract A new iterative method is suggested based on Jacobi iteration for solving a class of fuzzy linear systems of equations with crisp coefficient matrix and fuzzy right-hand side. The iterative scheme is established and the convergence theorems are presented. Numerical examples show that the method is effective and efficient compared with the classical Jacobi method.

MSC: 65F10; 08A72

Keywords: fuzzy linear system; Jacobi iterative method; new iterative scheme

Submission date: 12.04.2021 / Acceptance date: 14.10.2022

1. Introduction

Fuzzy linear systems (FLSs) occur in many fields, such as control problems, information, physics, statistics, engineering, economics, finance and even social sciences [1]. Thus, it is significant to study the numerical methods for solving FLSs.

In [1], a general model was proposed by Friedman et al. with embedding technique for solving a class of $n \times n$ FLSs

$$Ax = y, \tag{1.1}$$

where

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}$$

is a crisp matrix, $y = \left[y_1, y_2, \cdots, y_n\right]^{\mathrm{T}}$ is a fuzzy vector, and $x = \left[x_1, x_2, \cdots, x_n\right]^{\mathrm{T}}$ is unknown. The details of an FLS and the embedding model see in Section 2. With this model, many numerical methods [2–19] were developed to solve FLS (1.1). Abbasbandy, Ezzati and Jafarian discussed LU decomposition method [2], steepest descent method

*Corresponding author.

Published by The Mathematical Association of Thailand.

[3] and conjugate gradient method [4]. Allahviranloo considered the Jacobi and Gauss Sidel methods [6], SOR method [7] and Adomian decomposition method [8]. Dehghan and Hashemi [9] extended several well-known numerical algorithms of solving system of linear equations. Ezzati [10] designed a general model. Fariborzi Araghi and Fallahzadeh [11] used the inherited LU factorization. By introducing the width function of a fuzzy number, Li, Li and Kong [13] established a new algorithm model based on generalized elimination method. Miao, Wang, Wu, Yin and Zheng studied SOR [17], Uzawa-SOR [16], block (SOR) [14, 18] and new splitting [19] iterative methods. Nasseri, Matinfar and Sohrabi [15] proposed QR-decomposition method. Akram, Allahviranloo, Pedrycz and Ali [5] suggested a technique to solve (1.1) with LR-bipolar fuzzy numbers. Koam, Akram, Muhammad and Hussain [12] presented a new scheme for solving (1.1) with m-polar fuzzy numbers.

It is generally known that the classical Jacobi method is simple to implement and suitable to be used in parallel computing. Allahviranloo [6] investigated the Jacobi method for (1.1). In this paper, a new improved method based on Jacobi iteration is provided for solving (1.1), compared with the Jacobi method.

The rest of the paper is organized as follows. Section 2 gives some basic definitions and results of FLS. In Section 3, the new method is established with convergence theorems. Two numerical examples in Section 4 are discussed and the conclusion is in Section 5.

2. Preliminaries

Generally, following [1], a fuzzy number is a pair of $(\underline{u}(r), \overline{u}(r))$, $0 \leqslant r \leqslant 1$, satisfying,

- $\underline{u}(r)$ is a bounded left continuous nondecreasing function over $[0,1]$,
- $\overline{u}(r)$ is a bounded left continuous nonincreasing function over $[0,1]$,
- $\underline{u}(r) \leqslant \overline{u}(r)$, $0 \leqslant r \leqslant 1$.

The arithmetic operations of arbitrary fuzzy numbers $x = (\underline{x}(r), \overline{x}(r))$, $y = (\underline{y}(r), \overline{y}(r))$, $0 \leqslant r \leqslant 1$, and real number k, are as follows,

(1) $x = y$ if and only if $\underline{x}(r) = \underline{y}(r)$ and $\overline{x}(r) = \overline{y}(r)$,
(2) $x + y = (\underline{x}(r) + \underline{y}(r), \overline{x}(r) + \overline{y}(r))$, and
(3) $kx = \begin{cases} (k\underline{x}(r), k\overline{x}(r)), & k \geqslant 0, \\ (k\overline{x}(r), k\underline{x}(r)), & k < 0. \end{cases}$

Definition 2.1. [1] A fuzzy number vector $X = (x_1, x_2, \cdots, x_n)^{\mathrm{T}}$ given by

$$x_i = (\underline{x}_i(r), \overline{x}_i(r)), \quad 1 \leqslant i \leqslant n,\ 0 \leqslant r \leqslant 1,$$

is called a solution of the fuzzy linear system (1.1) if

$$\begin{cases} \underline{\sum_{j=1}^{n} a_{ij} x_j} = \sum_{j=1}^{n} \underline{a_{ij} x_j} = \underline{y}_i, \\ \overline{\sum_{j=1}^{n} a_{ij} x_j} = \sum_{j=1}^{n} \overline{a_{ij} x_j} = \overline{y}_i. \end{cases} \tag{2.1}$$

By (2.1), Friedman et al. [1] extend FLS (1.1) to a $2n \times 2n$ crisp linear system (embedding model)

$$SX = Y \tag{2.2}$$

where $S = (s_{kl})$, s_{kl} are determined as follows

$$\begin{array}{lll} a_{ij} \geqslant 0 \Rightarrow s_{ij} = a_{ij}, & s_{n+i,\, n+j} = a_{ij}, & \\ a_{ij} < 0 \Rightarrow s_{i,\, n+j} = a_{ij}, & s_{n+i,\, j} = a_{ij}, & \end{array} \quad 1 \leqslant i, j \leqslant n,$$

and any s_{kl} which is not determined by the above items is zero, $1 \leqslant k,\, l \leqslant 2n$, and

$$X = \begin{bmatrix} \underline{x}_1 \\ \vdots \\ \underline{x}_n \\ \overline{x}_1 \\ \vdots \\ \overline{x}_n \end{bmatrix}, \; Y = \begin{bmatrix} \underline{y}_1 \\ \vdots \\ \underline{y}_n \\ \overline{y}_1 \\ \vdots \\ \overline{y}_n \end{bmatrix}.$$

Further, the matrix S has the structure $\begin{bmatrix} S_1 & S_2 \\ S_2 & S_1 \end{bmatrix}$, $A = S_1 + S_2$, and (2.2) can be rewritten as

$$\begin{cases} S_1\underline{X} + S_2\overline{X} = \underline{Y}, \\ S_2\underline{X} + S_1\overline{X} = \overline{Y}, \end{cases} \tag{2.3}$$

where

$$\underline{X} = \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \\ \vdots \\ \underline{x}_n \end{bmatrix}, \; \overline{X} = \begin{bmatrix} \overline{x}_1 \\ \overline{x}_2 \\ \vdots \\ \overline{x}_n \end{bmatrix}, \; \underline{Y} = \begin{bmatrix} \underline{y}_1 \\ \underline{y}_2 \\ \vdots \\ \underline{y}_n \end{bmatrix}, \; \overline{Y} = \begin{bmatrix} \overline{y}_1 \\ \overline{y}_2 \\ \vdots \\ \overline{y}_n \end{bmatrix}.$$

The following theorem indicates when FLS (1.1) has a unique solution.

Theorem 2.2. [1] *The matrix S is nonsingular if and only if the matrices $A = S_1 + S_2$ and $S_1 - S_2$ are both nonsingular, that is, only when $A = S_1 + S_2$ and $S_1 - S_2$ are nonsingular, FLS (1.1) has a unique solution.*

In the next section, a new iterative scheme based on Jacobi iteration is presented for nonsingular FLS (1.1), in fact, (2.2).

3. The New Method Based on Jacobi Iteration

For nonsingular system (2.2), that is $A = S_1 + S_2$ and $S_1 - S_2$ are invertible, with the following splitting,

$$S = D - L - U, \tag{3.1}$$

where

$$D = \begin{bmatrix} D_1 & 0 \\ 0 & D_1 \end{bmatrix}, \; L = \begin{bmatrix} L_1 & 0 \\ -S_2 & L_1 \end{bmatrix}, \; U = \begin{bmatrix} U_1 & -S_2 \\ 0 & U_1 \end{bmatrix},$$

$D_1 = \mathrm{diag}\,(s_{ii})\,, s_{ii} \neq 0, i = 1, 2, \cdots, n$, $D_1 - L_1 - U_1 = S_1$ and L_1 and U_1 are strictly lower and upper triangular matrices, the Jacobi iterative scheme is as follows [6],

$$X_{k+1} = H_J X_k + D^{-1}Y, \quad k = 0, 1, \cdots, \tag{3.2}$$

where $X_k = \begin{bmatrix} \underline{X}_k \\ \overline{X}_k \end{bmatrix}$ and

$$\begin{aligned} H_J &= D^{-1}(L+U) \\ &= \begin{bmatrix} D_1^{-1}(L_1+U_1) & -D_1^{-1}S_2 \\ -D_1^{-1}S_2 & D_1^{-1}(L_1+U_1) \end{bmatrix}. \end{aligned}$$

By (2.3), the iterative scheme (3.2) is also

$$\begin{cases} \underline{X}_{k+1} = D_1^{-1}(L_1+U_1)\underline{X}_k - D_1^{-1}S_2\overline{X}_k + D_1^{-1}\underline{Y}, \\ \overline{X}_{k+1} = D_1^{-1}(L_1+U_1)\overline{X}_k - D_1^{-1}S_2\underline{X}_k + D_1^{-1}\overline{Y}, \end{cases} \quad k = 0, 1, \cdots. \tag{3.3}$$

For (3.2) or (3.3), the follow convergence theorems hold.

Theorem 3.1. *The Jacobi iterative scheme (3.2) or (3.3) converges if and only if the spectral radius of H_J is less than one, i.e., $\rho(H_J) < 1$.*

Theorem 3.2. *If S is symmetric and $s_{ii} > 0$, the Jacobi iterative scheme (3.2) or (3.3) converges if and only if S and $2D - S$ are positive definite.*

Theorem 3.3. *If S is strictly diagonally dominant matrix, the Jacobi iterative scheme (3.2) or (3.3) is convergent.*

Remark 3.4. Theorems 3.1-3.3 are well-known results for systems of linear equations, see [20], thus hold for (3.2) or (3.3).

In [21], Wang and Chen discussed a modified Jacobi method for linear system of equations. The improvement of the proposed method is to use the combination of the current point obtained by Jacobi method and the previous point to get the new point to get the optimal factor of linear combination by solving the least square optimization. By that idea, a new improved Jacobi iterative scheme for FLS (2.2) or (2.3), referred to NJ, can be established:

$$\begin{cases} X_{k+\frac{1}{2}} = H_J X_k + D^{-1}Y, \\ X_{k+1} = \omega_k X_{k+\frac{1}{2}} + (1-\omega_k) X_k, \end{cases} \quad k = 0, 1, \cdots, \tag{3.4}$$

where

$$\omega_k = -\frac{(SX_k - Y)^{\mathrm{T}}\left(SX_{k+\frac{1}{2}} - SX_k\right)}{\left(SX_{k+\frac{1}{2}} - SX_k\right)^{\mathrm{T}}\left(SX_{k+\frac{1}{2}} - SX_k\right)},$$

or

$$\begin{cases} \underline{X}_{k+\frac{1}{2}} = D_1^{-1}(L_1+U_1)\underline{X}_k - D_1^{-1}S_2\overline{X}_k + D_1^{-1}\underline{Y}, \\ \underline{X}_{k+1} = \omega_k \underline{X}_{k+\frac{1}{2}} + (1-\omega_k)\underline{X}_k, \\ \overline{X}_{k+\frac{1}{2}} = D_1^{-1}(L_1+U_1)\overline{X}_k - D_1^{-1}S_2\underline{X}_k + D_1^{-1}\overline{Y}, \\ \overline{X}_{k+1} = \omega_k \overline{X}_{k+\frac{1}{2}} + (1-\omega_k)\overline{X}_k, \end{cases} \quad k = 0, 1, \cdots, \tag{3.5}$$

where

$$\omega_k = -\frac{(SX_k - Y)^{\mathrm{T}}\left(SX_{k+\frac{1}{2}} - SX_k\right)}{\left(SX_{k+\frac{1}{2}} - SX_k\right)^{\mathrm{T}}\left(SX_{k+\frac{1}{2}} - SX_k\right)}.$$

Remark 3.5. The combination factor ω_k is the solution of the least square optimization

$$\min_{\omega_k} \left\| S\left(\omega_k X_{k+\frac{1}{2}} + (1-\omega_k)X_k\right) - Y \right\|^2 .$$

The NJ method is described as the following algorithm.

Algorithm 1 NJ algorithm

1: Given initial vector X and error precision ε, calculate $R = Y - SX$ and set $k = 0$;

2: While $\|R\|_2 > \varepsilon \|Y\|_2$ and $k < k_{\max}$, do

$$\begin{cases} X_p = X, \\ X = H_J X + D^{-1} Y, \\ \omega = -\dfrac{(SX - Y)^{\mathrm{T}} (SX - SX_p)}{(SX - SX_p)^{\mathrm{T}} (SX - SX_p)}, \\ X = \omega X + (1-\omega) X_p, \\ R = Y - SX, \\ k = k + 1. \end{cases}$$

Remark 3.6. According to [21], the NJ method is convergent if and only if $\rho(H_J) < 1$, thus, Theorems 3.1-3.3 also hold for the NJ method.

4. Numerical Examples

This section gives two examples to show the effectiveness of the new method. All implements using Matlab 7 run in a Windows 7 DELL laptop with Intel 2.80GHz CPU and 8.00GB RAM. In the numerical experiments, the initial guess is 0 and the stopping criterion is

$$\|R_k\|_2 < 10^{-6},$$

where R_k is the residual vector after k iterations, i.e., $R_k = Y - SX_k$.

In the tables, x_a and x_b mean that $SX = Y$ is solved as two numeric systems

$$S \begin{bmatrix} x_{a1} \\ x_{a2} \\ \vdots \\ x_{a,2n} \end{bmatrix} = \begin{bmatrix} y_{a1} \\ y_{a2} \\ \vdots \\ y_{a,2n} \end{bmatrix} \text{ and } S \begin{bmatrix} x_{b1} \\ x_{b2} \\ \vdots \\ x_{b,2n} \end{bmatrix} = \begin{bmatrix} y_{b1} \\ y_{b2} \\ \vdots \\ y_{b,2n} \end{bmatrix}$$

not one symbolic system

$$S \begin{bmatrix} x_{a1} + x_{b1} r \\ x_{a2} + x_{b2} r \\ \vdots \\ x_{a,2n} + x_{b,2n} r \end{bmatrix} = \begin{bmatrix} y_{a1} + y_{b1} r \\ y_{a2} + y_{b2} r \\ \vdots \\ y_{a,2n} + y_{b,2n} r \end{bmatrix}$$

in the actual calculations.

Example 4.1. Consider $n \times n$ fuzzy linear system $Ax = y$ with

$$A = \begin{bmatrix} 8 & -1 & -1 & -1 & & & & & \\ -1 & 8 & -1 & -1 & \ddots & & & & \\ -1 & -1 & 8 & -1 & \ddots & \ddots & & & \\ -1 & -1 & -1 & 8 & \ddots & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \ddots & 8 & -1 & -1 & -1 \\ & & & \ddots & \ddots & -1 & 8 & -1 & -1 \\ & & & & \ddots & -1 & -1 & 8 & -1 \\ & & & & & -1 & -1 & -1 & 8 \end{bmatrix}$$

and

$$y = \begin{bmatrix} (2+r, 2+r) \\ (2+r, 2+r) \\ \vdots \\ (2+r, 2+r) \end{bmatrix}.$$

Table 1
Iterations (IT) and Residual (RES) for Example 4.1

n	Jacobi				NJ			
	IT_{x_a}	RES_{x_a}	IT_{x_b}	RES_{x_b}	IT_{x_a}	RES_{x_a}	IT_{x_b}	RES_{x_b}
16	46	7.2625e-007	44	7.4434e-007	30	5.2909e-007	28	7.7411e-007
32	54	9.0938e-007	52	8.4110e-007	33	4.1344e-007	30	8.5933e-007
64	58	8.6441e-007	56	7.7656e-007	39	5.5691e-007	38	9.7607e-007
128	60	8.6662e-007	58	7.7297e-007	35	8.7319e-007	35	4.3660e-007
256	62	7.5204e-007	59	8.9320e-007	36	6.0221e-007	35	9.1825e-007
512	63	8.2970e-007	60	9.8431e-007	35	7.8261e-007	34	6.8072e-007
1024	64	8.9668e-007	62	7.9730e-007	32	7.4043e-007	30	8.4049e-007
2048	65	9.5986e-007	63	8.5334e-007	40	8.3323e-007	39	7.4473e-007

From Table 1, one can see that the NJ method needs less iterations and the number of iterations increases slow with the size of the system increasing. This shows that the combination technique of the NJ method works.

Example 4.2. Consider $n^2 \times n^2$ fuzzy linear system $Ax = y$ with

$$A = \begin{bmatrix} D & B^{\mathrm{T}} & & & \\ B & D & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & D & B^{\mathrm{T}} \\ & & & B & D \end{bmatrix},$$

where

$$B = \begin{bmatrix} 0.5 & -0.25 & & & \\ -0.25 & 0.5 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & 0.5 & -0.25 \\ & & & -0.25 & 0.5 \end{bmatrix}, \; D = \begin{bmatrix} 5 & -1 & & & \\ -1 & 5 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & 5 & -1 \\ & & & -1 & 5 \end{bmatrix},$$

and

$$y = \begin{bmatrix} (1+r, 1+r) \\ (2+r, 2+r) \\ \vdots \\ (n^2+r, n^2+r) \end{bmatrix}.$$

Table 2
Iterations (IT) and Residual (RES) for Example 4.2

n	Jacobi				NJ			
	IT_{x_a}	RES_{x_a}	IT_{x_b}	RES_{x_b}	IT_{x_a}	RES_{x_a}	IT_{x_b}	RES_{x_b}
10	52	8.9286e-007	36	8.2241e-007	44	9.4757e-007	36	7.3886e-007
15	68	9.9214e-007	49	9.6275e-007	45	8.2494e-007	35	8.7850e-007
20	68	9.7114e-007	48	9.0557e-007	46	8.3219e-007	35	8.8229e-007
25	75	9.5463e-007	51	8.1862e-007	47	8.1817e-007	35	9.5329e-007
30	76	9.4190e-007	51	8.5102e-007	48	7.6403e-007	36	6.3217e-007
35	79	8.5899e-007	51	9.7768e-007	49	6.7924e-007	36	6.9445e-007
40	80	9.4393e-007	52	8.4129e-007	49	9.6177e-007	36	7.6013e-007
45	82	8.3391e-007	52	9.1032e-007	50	7.9229e-007	36	8.2780e-007

Table 2 also shows that, as n increases, the classical Jacobi requires more iterations while the proposed NJ method needs less and the number of iterations is not sensitive to the size of the system.

5. Conclusion

A new improved method based on Jacobi iteration is presented for solving $n \times n$ fuzzy linear system. The numerical results show that the method is effective and improves the convergence, which is faster than the classical Jacobi method and seems not sensitive to the size of the system. Further work would be analyzing the reason of the stable iterations to improve the method much.

Acknowledgements

The authors would like to thank the referees for their invaluable comments and suggestions on the manuscript. This work was supported by Key Scientific Research Project of Colleges and Universities in Henan Province (20B110012), China.

References

[1] M. Friedman, M. Ming, A. Kandel, Fuzzy linear systems, Fuzzy Sets and Systems 96 (1998) 201–209.

[2] S. Abbasbandy, R. Ezzati, A. Jafarian, LU decomposition method for solving fuzzy system of linear equations, Appl. Math. Comput. 172 (2006) 633–643.

[3] S. Abbasbandy, A. Jafarian, Steepest descent method for system of fuzzy linear equations, Appl. Math. Comput. 175 (2006) 823–833.

[4] S. Abbasbandy, A. Jafarian, R. Ezzati, Conjugate gradient method for fuzzy symmetric positive definite system of linear equations, Appl. Math. Comput. 171 (2005) 1184–1191.

[5] M. Akram, T. Allahviranloo, W. Pedrycz, M. Ali, Methods for solving LR-bipolar fuzzy linear systems, Soft Comput. 25 (2021) 85–108.

[6] T. Allahviranloo, Numerical methods for fuzzy system of linear equations, Appl. Math. Comput. 155 (2004) 493–502.

[7] T. Allahviranloo, Successive over relaxation iterative method for fuzzy system of linear equations, Appl. Math. Comput. 162 (2005) 189–196.

[8] T. Allahviranloo, The Adomian decomposition method for fuzzy system of linear equations, Appl. Math. Comput. 163 (2005) 553–563.

[9] M. Dehghan, B. Hashemi, Iterative solution of fuzzy linear systems, Appl. Math. Comput. 175 (2006) 645–674.

[10] R. Ezzati, Solving fuzzy linear systems, Soft Comput. 15 (2011) 193–197.

[11] M.A. Fariborzi Araghi, A. Fallahzadeh, Inherited LU factorization for solving fuzzy system of linear equations, Soft Comput. 17 (2013) 159–163.

[12] A.N.A. Koam, M. Akram, G. Muhammad, N. Hussain, LU Decomposition Scheme for Solving m-Polar Fuzzy System of Linear Equations, Math. Probl. Eng. 2020 (2020) Article ID 8384593.

[13] J. Li, W. Li, X. Kong, A new algorithm model for solving fuzzy linear systems, Southeast Asian Bull. Math. 34 (2010) 121–132.

[14] S.-X. Miao, B. Zheng, K. Wang, Block SOR methods for fuzzy linear systems, J. Appl. Math. Comput. 26 (2008) 201–218.

[15] S.H. Nasseri, M. Matinfar, M. Sohrabi, QR-decomposition method for solving fuzzy system of linear equations, Int. J. Math. Comput. 4 (2009) 129–136.

[16] K. Wang, Y. Wu, Uzawa-SOR method for fuzzy linear system, International Journal of Information and Computer Science 1 (2012) 36–39.

[17] K. Wang, B. Zheng, Symmetric successive overrelaxation methods for fuzzy linear systems, Appl. Math. Comput. 175 (2006) 891–901.

[18] K. Wang, B. Zheng, Block iterative methods for fuzzy linear systems, J. Appl. Math. Comput. 25 (2007) 119–136.

[19] J.-F. Yin, K. Wang, Splitting iterative methods for fuzzy system of linear equations, Comput. Math. Model. 20 (2009) 326–335.

[20] D.M. Young, Iterative Solution of Large Linear Systems, Academic, New York, 1971.

[21] Y.-r. Wang, Y.-l. Chen, A modified Jacobi iterative method for large-size linear systems, Journal of Shandong University (Natural Science) 55 (2020) 122–126.