Journal of Quantum Technologies and Informatics Research
JQTAIR

KUANTUM TEKNOLOJİLERİ VE ENFORMATİK ARAŞTIRMALARI
JOURNAL OF QUANTUM TECHNOLOGIES AND INFORMATICS RESEARCH
Volume / Cilt: 3, Issue / Sayı: 1, 2025, e2786
E - ISSN: 3023-4735
URL: https://journals.gen.tr/index.php/jqtair/
DOI: https://doi.org/10.70447/ktve.2786

Received Date / Geliş Tarihi: 13/02/2025
Acccepted Date / Kabul Tarihi: 27/03/2025
Online Publication Date /
Online Yayın Tarihi: 07/04/2025

RESEARCH ARTICLE / ARAŞTIRMA MAKALESİ

Iterative Solutions for Certain Complex Coefficient Linear Systems: Jacobi and Gauss-Seidel Methods

Ahmet Zahid Küçük[1] **Bayram Köse**[2] **Bahriye Karaca**[3]

[1] Karabuk University, Faculty of Science, Department of Mathematics,Karabuk, Turkiye,
e-mail: azahidkucuk@karabuk.edu.tr
[2] İzmir Bakırcay University, Faculty of Engineering and Architecture, Department of Electrical and Electronics Engineering, Izmir, Turkiye, e-mail: bayram.kose@bakircay.edu.tr
[3] İzmir Bakırcay University, Faculty of Engineering and Architecture, Department of Basic Sciences, Izmir, Turkiye, e-mail: bahriye.karaca@bakircay.edu.tr

Abstract

In this study, the performance of the Jacobi and Gauss-Seidel iteration methods for solving systems of linear equations with complex coefficients is analyzed. The coefficient matrix of the system is transformed into a real coefficient system by separating the real and imaginary parts. The study aims to compare the accuracy and computational efficiency of these methods within the context of selected examples, while also evaluating their convergence behavior. The findings demonstrate that, for the examples considered, the Gauss-Seidel method converges faster and with lower initial errors compared to the Jacobi method

Keywords*:* Linear equations systems with complex coefficients, Jacobi iteration method, Gauss-Seidel iteration method, Complex-Real transform.

Citation/Atıf: KÜÇÜK, A.Z., KÖSE, B. & KARACA, B. (2025). Iterative Solutions for Certain Complex Coefficient Linear Systems: Jacobi and Gauss-Seidel Methods. *Kuantum Teknolojileri ve Enformatik Araştırmaları*. 3(1): e2786, DOI: 10.70447/ktve.2786

Corresponding Author/ Sorumlu Yazar:
Ahmet Zahid Küçük
E-mail: azahidkucuk@karabuk.edu.tr

1. INTRODUCTION

In many real-life problems, the relationships between variables are non-linear. However, linear approaches are widely employed in engineering and other scientific disciplines. Numerous studies are conducted based on linear models, which are frequently preferred due to their suitability for applying both approximate and exact methods in the analysis and resolution of mathematical problems [1]. Linear models, particularly in areas such as engineering, provide exact or approximate solutions by addressing problems with a systematic approach and facilitate analysis processes.

Some problems in science and technology are inherently more complex. Particularly in physics and engineering, where modeling and solving complex systems are essential, large-scale linear systems of equations are commonly utilized. For example, linear equations with a large number of variables can be used to describe scientific problems such as calculating the structural strength of a bridge or forecasting weather conditions. The extensive use of linear systems has consequently positioned their solutions as a central topic in numerical computation.

Problems involving linear systems with complex coefficients can be exemplified by the propagation of electromagnetic waves. Complex numbers are used to represent electromagnetic waves, with their two-dimensional structure enabling the simultaneous representation of both the wave's amplitude (magnitude) and phase (temporal shift). Additionally, sinusoidal functions used to describe electromagnetic waves can be converted into complex expressions through Euler's formula, $e^{i\theta} = cos\theta + isin\theta$. As emphasized with the example of electromagnetic waves, complex numbers play a significant role in expressing problems in physics and engineering.

Iterative methods for solving linear algebraic systems require fewer computational resources, such as memory, compared to direct methods. Consequently, iterative methods are often favored for solving large-scale systems [2]. Iterative methods used for linear systems generally fall into two categories: Stationary methods (e.g., Gauss-Seidel, Jacobi, SOR) and methods based on Krylov subspace (e.g., Conjugate Gradient, Generalized Minimal Residual) [2].

Building on the basic classification and efficiency of iterative methods, their practical applications and comparative performance have been extensively studied. Numerous studies have analyzed the performance of these methods for different system sizes and coefficient classes. These analyses contribute to a deeper understanding of the effectiveness and convergence properties of iterative methods, particularly when applied to systems with varied complexities. This foundation lays the way for facilitating specific studies that compare the behavior and convergence of these methods.

In the context of the methods discussed, a brief literature review on studies utilizing the Gauss-Seidel method will now be presented. Iterative methods for solving linear systems have evolved significantly throughout the 20th century, encouraged by increasing computational demands in science and engineering. Although their origins date back to the 19th century with Gauss's early work, recent advancements have resulted in a broad spectrum of specialized algorithms designed to address large and complex systems [3]. Among these, the Jacobi and Gauss-Seidel methods have remained central due to their simplicity and practical effectiveness.

The Gauss-Seidel method, while traditionally sequential due to variable dependencies, has inspired various parallel adaptations—such as the red-black Gauss-Seidel (RBGS) method for sparse systems [4]—and has been implemented in distributed computing frameworks [5]. Building on the strengths of both classical methods, [2] developed an algorithm that merges the convergence behavior of Gauss-Seidel with the inherent parallelism of Jacobi. Comparative studies continue to evaluate these methods under diverse conditions. For example, the solutions of two distinct linear systems with real coefficients (one of order 3×3 and the other 5×5) were examined in [6] using both Jacobi and Gauss-Seidel iterations. In [7], convergence conditions for both methods were presented and illustrated with numerical examples, along with a formula to estimate the required number of iterations. Other studies extend this comparison to fuzzy systems [8] or explore the convergence domains of these methods for systems with real and complex coefficients [9], further demonstrating the enduring relevance and adaptability of the Gauss-Seidel method across a range of problem domains.

In this study, the results obtained by using Jacobi and Gauss-Seidel iterative methods to solve linear equation systems with complex coefficients are analyzed (based on certain examples). The method used to conduct this study as described in the following. The approach used in this study includes several steps, starting with the extraction of two separate matrices from the $n \times n$ coefficient matrix of the system, representing the real and imaginary parts of the matrix entries. Thus, the initial $n \times n$ matrix with complex coefficients is extended to a real matrix of order $2n \times 2n$ and the initial system is effectively transformed into a real system. Subsequently, the Jacobi and Gauss-Seidel iterative methods were applied to this transformed real system, and the solutions were analyzed.

2. THE COMPLEX-REAL TRANSFORM APPROACH

Let $F^{m\times n}$ denote the set of matrices of order $m \times n$ defined over a field F. If the entries of a matrix K of order $m \times n$ are chosen from the field of complex numbers ($\mathbb{C}$), then this matrix is denoted by $K \in \mathbb{C}^{m\times n}$.

Let $A = [z_{jk}] \in \mathbb{C}^{n\times n}$, $X = [x_1 \quad x_2 \quad \cdots \quad x_n]^T \in \mathbb{C}^{n\times 1}$, and $B = [b_1 \quad b_2 \quad \cdots \quad b_n]^T \in \mathbb{C}^{n\times 1}$. We then consider a linear equation system with complex coefficients (in short LCC)

$$AX = B. \tag{4}$$

Following the transformation of a LCC into a linear equation system with reel coefficients (in short LRC) using the method detailed below, we will explore its solution through iterative methods.

The product of two complex numbers yields another complex number, i.e.,

$$(a+ib)(x+iy) = c + id \tag{5}$$

where $i^2 = -1$ and $a, b, x, y, c, d \in \mathbb{R}$. From Eq.(5), we obtain

$$ax - by = c \tag{6}$$

$$ibx + iay = id$$

which subsequently allows us to write

$$\begin{bmatrix} a & -b \\ b & a \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} c \\ d \end{bmatrix}. \tag{7}$$

In order to better analyze a linear system with complex coefficients, an approach has been proposed, modeled on the process of transforming the LCC given in Eq.(4) into its matrix form as shown in Eq.(7). This approach involves separating the real and complex components of the coefficients. A brief explanation of this method is provided below, while detailed information can be found in references [9] and [10].

Let us consider each entry of the coefficient matrix $A = [z_{jk}]$ in Eq.(4) as a complex number of the form

$$z_{jk} = Re(z_{jk}) + i.Im(z_{jk}). \tag{8}$$

Hence, we construct the matrices

$$A^{\mathbb{R}} = [Re(z_{jk})] \in \mathbb{R}^{n\times n}, \tag{9}$$

and

$$A^{\mathbb{C}} = [Im(z_{jk})] \in \mathbb{C}^{n\times n}. \tag{10}$$

Then we build the augmented matrix $\tilde{A} \in \mathbb{R}^{2n\times 2n}$ as follows:

$$\tilde{A} = \begin{bmatrix} A^{\mathbb{R}} & -A^{\mathbb{C}} \\ A^{\mathbb{C}} & A^{\mathbb{R}} \end{bmatrix}_{2n\times 2n} \tag{11}$$

For example, given the matrix $A = \begin{bmatrix} 2+3i & 4-5i \\ 7 & 1+9i \end{bmatrix}$, we separate it into its real and complex parts as follows: $A^{\mathbb{R}} = \begin{bmatrix} 2 & 4 \\ 7 & 1 \end{bmatrix}$ and $A^{\mathbb{C}} = \begin{bmatrix} 3 & -5 \\ 0 & 9 \end{bmatrix}$. We then form the matrix $\tilde{A}$ as

$$\tilde{A} = \begin{bmatrix} 2 & 4 & -3 & 5 \\ 7 & 1 & 0 & -9 \\ 3 & -5 & 2 & 4 \\ 0 & 9 & 7 & 1 \end{bmatrix}.$$

In a similar manner as the process used to obtain the matrix $\tilde{A}$, let us denote the real and complex parts of the elements of the column vector $X \in \mathbb{C}^{n\times 1}$ in Eq.(4) as $X^{\mathbb{R}}$ and $X^{\mathbb{C}}$, respectively, and build the following column vector:

$$\mathcal{X} = \begin{bmatrix} X^{\mathbb{R}} \\ X^{\mathbb{C}} \end{bmatrix}_{2n\times 1} \tag{12}$$

Similarly, let us denote the real and complex parts of the elements of the column vector $B \in \mathbb{C}^{n\times 1}$ in Eq.(4) as $B^{\mathbb{R}}$ and $B^{\mathbb{C}}$, respectively, and build the following column vector:

$$\mathcal{B} = \begin{bmatrix} B^{\mathbb{R}} \\ B^{\mathbb{C}} \end{bmatrix}_{2n\times 1} \tag{13}$$

Thus, the $n \times n$ complex linear system $AX = B$ is transformed into a $2n \times 2n$ real linear system

$$\tilde{A}\mathcal{X} = \mathcal{B} \tag{14}$$

as shown below:

$$\begin{bmatrix} A^{\mathbb{R}} & -A^{\mathbb{C}} \\ A^{\mathbb{C}} & A^{\mathbb{R}} \end{bmatrix} \begin{bmatrix} X^{\mathbb{R}} \\ X^{\mathbb{C}} \end{bmatrix} = \begin{bmatrix} B^{\mathbb{R}} \\ B^{\mathbb{C}} \end{bmatrix}. \tag{15}$$

This process is referred to as the *Complex-Real Transform* in [11]. After converting the LCC to LRC using the method described above, the following proposition, proven in [12], should be considered in the solution process.

Proposition: ([12]) The complex linear system $AX = B$ has a unique solution if and only if the real linear system $\tilde{A}\mathcal{X} = \mathcal{B}$ has a unique solution.

3. GENERAL ITERATION FORMULAE OF JACOBI AND GAUSS-SEIDEL METHODS

Let us consider n by n complex linear system $AX = B$, and split the coefficient matrix $A = [a_{ij}]$ into three parts as

$$A = L + D + U, \tag{16}$$

The following conditions hold for the matrices L, U and D in In Eq.(16):

- The lower triangular matrix L is obtained by replacing the elements $a_{ij} \in A$ with zero if $i \le j$.
- The upper triangular matrix U is obtained by replacing the elements $a_{ij} \in A$ with zero if $j \le i$.
- The diagonal matrix D is obtained by replacing the elements $a_{ij} \in A$ with zero if $i \ne j$.

By combining Eq.(4) and Eq.(16) yields the following equation:

$$DX = B - (L + U)X \tag{17}$$

Therefore, the following equation can be expressed for the matrix representation of the Jacobi iteration method.

$$DX^{(k+1)} = \mathrm{B} - (L + U)X^{(k)} \tag{18}$$

In Eq.(18), $X^{(k)}$ denotes the solution vector at the k^{th} iteration step. Let $X = [x_1 \quad x_2 \quad \cdots \quad x_n]^T$ and $B = [b_1 \quad b_2 \quad \cdots \quad b_n]^T$. According to the Jacobi iteration method outlined in Eq. (18), each x_i is calculated using the following formula [13].

$$x_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{\substack{j=1 \\ i\ne j}}^{n} a_{ij}x_j^{(k)} \right) \tag{19}$$

where $a_{ii} \ne 0$. Each variable in the linear system of equations with complex coefficients, as expressed in Eq. (19), converges iteratively toward the true solution or diverges, depending on the initial guess values provided.

With a simple modification on Eq.(18), we obtain the following equation, which represents the matrix form of the Gauss-Seidel iteration.

$$DX^{(k+1)} = \mathrm{B} - \left(LX^{(k+1)} + UX^{(k)}\right) \tag{20}$$

According to the Gauss-Seidel iteration method described in Eq.(20), each x_i is computed using the following formula [13].

$$x_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)} \right) \tag{21}$$

where $i \ne j$.

Apart from the formula given in Eq. (19), the formula presented in Eq. (21) accounts for the calculation of the i^{th} variable of the ith equation using the iteration values from the 1st variable to the $(j-1)^{\text{th}}$ variable at step $(k+1)$, as well as the values from the $(j+1)^{\text{th}}$ to the n^{th} variable at step k.

For further details, including MATLAB code related to the Jacobi and Gauss-Seidel iteration methods, see [14].

4. CONVERGENCE CONDITION OF GAUSS-SEIDEL AND JACOBI METHOD

Techniques used to find solutions in root-finding problems vary significantly in terms of how they ensure convergence. These variations stem from the convergence properties of the method, initial conditions, and the structure of the function being used.

The fixed-point technique, often considered the fundamental idea behind many iteration methods, aims to find a point where the function satisfies the condition of x=f(x). In this method, an iterative process is initiated based on an initial estimate, and the function's output at each step is used as the new estimate. However, fixed-point iteration may not always converge rapidly or reliably.

Jacobi iteration, a method developed based on the fixed-point technique, is commonly used for solving linear systems of equations. According to the Jacobi method, the new values of the unknowns in each iteration are updated based on the previous iteration values of the other unknowns. This technique can be particularly effective for solving large-scale and sparse matrices. The convergence speed and reliability of this method depend on the structure of the system and the properties of the matrix.

The Gauss-Seidel iteration method is a modified version of the Jacobi iteration. By using more updated solution values during each iteration, the Gauss-Seidel method improves convergence speed, enabling more efficient solutions for large and complex systems.

The sufficiency condition for the convergence of Jacobi and Gauss-Seidel methods is expressed by Theorem. 7.21 in [13], as shown below.

Teorem: ([13]) If A is strictly diagonally dominant, then for any choice of $x^{(0)}$, both the Jacobi and Gauss-Seidel methods give sequences $\left\{x^{(k)}\right\}_{k=0}^{\infty}$ that converge to the unique solution of $AX = B$.

Definition: The n by n matrix A is said to be diagonally dominant when

$$|a_{ii}| \geq \sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}| \tag{22}$$

holds for each $i = 1, 2, \cdots, n$.

A diagonally dominant matrix is said to be strictly diagonally dominant when the inequality in Eq.(22) is strict for each n, that is, when

$$|a_{ii}| > \sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}| \tag{23}$$

holds for each $i = 1, 2, \dots, n$.

Ensuring this condition guarantees faster and more reliable convergence of iterative methods. In the Jacobi and Gauss-Seidel methods, if the coefficient matrix exhibits this dominance, potential deviations during iterations are minimized, leading to a reliable convergence towards the solution. As frequently emphasized in the literature, systems that do not exhibit diagonal dominance may tend to diverge during iterations or require significantly more iterations to converge. This can reduce the efficiency of these methods. Therefore, satisfying the diagonal dominance of the coefficient matrix is considered an essential prerequisite for the successful application of these methods.

5. NUMERICAL EXAMPLES

In this section we present numerical examples analyzing the solutions obtained by applying the Jacobi and Gauss-Seidel iterations to the system transformed by the application of the complex-real transformation.

5.1. Example 1

Let us consider the equation

$$(4 + 2i)x = 20 - 10i. \tag{24}$$

Assuming $x = u + iv$, then we obtain

$$\begin{bmatrix} 4 & -2 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} 20 \\ -10 \end{bmatrix}. \tag{25}$$

The results obtained by applying Jacobi and Gauss-Seidel methods to the real system in Eq.(25) are given in Table-1.

Table 1. Values of variables depending on iterations

Iter No	Jacobi Method		Gauss Seidel Method	
	u	v	u	v
1	5	-2.5	5	-5
2	3.75	-5	2.5	-3.75
3	2.5	-4.375	3.125	-4.0625
.	.	.	.	.
8	2.9883	-3.9844	2.9999	-3.9999
9	3.0078	-3.9941	3	-4
.	.	.	.	.
16	3	-3.9999	3	-4
17	3	-4	3	-4
.	.	.	.	.
20	3	-4	3	-4

5.2. Example 2

Let us consider the following LCC.

$$\begin{aligned}(5+i)x + (-2+3i)y &= 13+14i \\ (2-i)x + (4+2i)y &= 13-6i\end{aligned} \quad (26)$$

The matrix representation of this system is as follows.

$$\begin{bmatrix} 5+i & -2+3i \\ 2-i & 4+2i \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 13+14i \\ 13-6i \end{bmatrix} \quad (27)$$

Applying complex-real transforms to the system in Eq.(27), we obtain

$$\begin{bmatrix} 5 & -2 & -1 & -3 \\ 2 & 4 & 1 & -2 \\ 1 & 3 & 5 & -2 \\ -1 & 2 & 2 & 4 \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \\ v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} 13 \\ 13 \\ 14 \\ -6 \end{bmatrix} \quad (28)$$

where $= u_1 + iv_1$, $y = u_2 + iv_2$. The results obtained by applying Jacobi and Gauss-Seidel methods to the system given by Eq.(28) are given in Table-2 and Table-3, respectively.

Table 2. Values of the variables depending on iterations

Iteration No	Jacobi Iteration Method			
	u_1	u_2	v_1	v_2
1	2.6	3.25	2.8	-1.5
2	3.56	0.5	-0.27	-3.875
3	0.421	-0.4	0.238	-0.725
.	.	.	.	.
12	1.5362	-0.0078	0.1578	-1.9098
13	1.4826	1.4876	1.7335	-1.191
.	.	.	.	.
38	1.955	1.13	1.1547	-1.8912
39	2.1482	1.0382	0.9745	-2.1536
40	1.9181	0.8555	0.886	-1.9693

Table 3. Values of the variables depending on iterations

Iteration No	Gauss-Seidel Iteration Method			
	u_1	u_2	v_1	v_2
1	2.6	1.95	1.11	-2.38
2	2.174	0.6955	0.9959	-1.8022
3	1.9961	1.1019	1.0188	-2.0613
.	.	.	.	.
9	2	1.0002	1	-2.0001
10	2	0.9999	1	-2
11	2	1	1	-2
.	.	.	.	.
39	2	1	1	-2
40	2	1	1	-2

In the following example, the solution of the system of complex linear equations given in [12] will be discussed. It is worth noting that during the iterative solution of this problem, as outlined in [12], a specialized storage (or compression) method referred to as VRC (value-row-column) was utilized for the coefficient matrix. In this study, solutions obtained through the direct application of Jacobi and Gauss-Seidel iteration methods to this problem will be presented. Additionally, different from the results reported in [12], detailed error analysis results and convergence graphs of the solutions toward the root will be provided.

5.3. Example 3

Let us consider the complex linear system $AX = B$ where matrix A and matrix B are as follows:

$$\begin{bmatrix} 19.73 & 12.11 - i & 5 & 0 & 0 \\ -0.51i & 32.3 + 7i & 23.07 & i & 0 \\ 0 & -0.51i & 70 + 7.3i & 3.95 & 19 + 31.83i \\ 0 & 0 & 1 + 1.1i & 50.17 & 45.51 \\ 0 & 0 & 0 & -9.351i & 55 \end{bmatrix} \quad (29)$$

$$B = \begin{bmatrix} 77.38 + 8.82i \\ 157.48 + 19.8i \\ 1175.62 + 20.69i \\ 912.12 - 801.75i \\ 550 - 1060.4i \end{bmatrix}. \quad (30)$$

Accordingly, the matrices obtained from the real and imaginary parts of the elements of the matrix $A \in \mathbb{C}^{5\times 5}$ are as

$$A^{\mathbb{R}} = \begin{bmatrix} 19.73 & 12.11 & 0 & 0 & 0 \\ 0 & 32.3 & 23.07 & 0 & 0 \\ 0 & 0 & 70 & 3.95 & 19 \\ 0 & 0 & 1 & 50.17 & 45.51 \\ 0 & 0 & 0 & 0 & 55 \end{bmatrix} \quad (31)$$

and

$$A^{\mathbb{C}} = \begin{bmatrix} 0 & 1 & -5 & 0 & 0 \\ 0.51 & -7 & 0 & -1 & 0 \\ 0 & 0.51 & -7.3 & 0 & -31.83 \\ 0 & 0 & -1.1 & 0 & 0 \\ 0 & 0 & 0 & 9.351 & 0 \end{bmatrix} \quad (32)$$

On the other hand, the matrices obtained from the real and imaginary parts of the elements of the matrix $B \in \mathbb{C}^{10\times 1}$ are as

$$B^{\mathbb{R}} = \begin{bmatrix} 77.38 \\ 157.48 \\ 1175.62 \\ 912.12 \\ 550 \end{bmatrix} \quad (33)$$

and

$$B^{\mathbb{C}} = \begin{bmatrix} 8.82 \\ 19.8 \\ 20.69 \\ -801.75 \\ -1060.4 \end{bmatrix}. \quad (34)$$

The matrices $A^{\mathbb{R}}$, $A^{\mathbb{C}}$, $B^{\mathbb{R}}$ ve $B^{\mathbb{C}}$ are written in Eq (15) to obtain the system $\tilde{A}\mathcal{X} = B$ with order 10×10. The exact solution of this transformed system is

$$\mathcal{X} = [3.3;\ 1\ ;\ 5.5\ ;\ 9\ ;\ 10\ ;\ -1\ ;\ 0.17\ ;\ 0\ ;\ 0\ ;\ -17.75\ ]^T.$$

Considering that $\mathcal{X} = [X^{\mathbb{R}} \quad X^{\mathbb{C}}]^T \in \mathbb{C}^{10\times 1}$ ve $X = X^{\mathbb{R}} + iX^{\mathbb{C}}$, the solution of the original system is obtained as follows:

$$X = \begin{bmatrix} 3.3 - i \\ 1 - 0.17i \\ 5.5 \\ 9 \\ 10 - 17.75i \end{bmatrix} \quad (34)$$

Considering that $\mathcal{X} = [u_1 \quad u_2 \quad u_3 \quad u_4 \quad u_5 \quad v_1 \quad v_2 \quad v_3 \quad v_4 \quad v_5]^T$, the numerical solution results of this problem based on the iteration methods Jacobi and Gauss-Seidel are given in Table-4 and Table-5.

Table 4. Values of the variables depending on iterations with Jacobi iteration method

	Jacobi method									
	u_1	u_2	u_3	u_4	u_5	v_1	v_2	v_3	v_4	v_5
1	3.9219	4.8755	16.7946	18.1806	10	0.447	0.613	0.2956	-15.980	-19.28
2	0.9732	-7.4888	4.3138	8.7812	12.717	-3.9382	-1.1557	0.1674	1.1344	-16.189
3	8.6195	1.6413	5.5118	6.5625	9.8071	-0.3164	1.8599	-1.6613	-1.3933	-17.787
.	.	.	.	.	.	.	.	.	.	.
7	3.2088	0.9627	5.5012	9.0568	10.0027	-0.9612	0.1287	0.0376	0.0317	-17.7473
8	3.3341	0.9904	5.5015	8.9981	9.9946	-0.9773	0.1478	-0.0042	-0.0032	-17.7402
9	3.3055	0.9935	5.5058	9.0046	10.0005	-0.9877	0.1754	-0.0003	-0.0088	-17.7502
10	3.3032	0.9964	5.4995	8.9992	10.0015	-1.0055	0.1713	-0.0004	0.0001	-17.7491
11	3.3016	1.0006	5.5	8.9984	10	-1.0013	0.1709	-0.0009	-0.0008	-17.75
12	3.2989	1	5.5	8.9998	10.0001	-1.0009	0.1703	0	0	-17.7501
13	3.2995	0.9999	5.4999	8.9997	10	-1.0007	0.1697	0	0.0001	-17.7499
.	.	.	.	.	.	.	.	.	.	.
39	3.2997	0.9998	5.5001	8.9998	10	-1.0004	0.1698	0	-0.0001	-17.7499
40	3.2997	0.9998	5.5001	8.9998	10	-1.0004	0.1698	0	-0.0001	-17.7499

Table 5. Values of the variables depending on iterations with Gauss-Seidel iteration method

	Gauss-Seidel method									
	u_1	u_2	u_3	u_4	u_5	v_1	v_2	v_3	v_4	v_5
1	3.9219	4.8755	16.7946	17.8458	10	-3.562	-0.9342	-5.9675	-16.2299	-16.2459
2	-0.5355	-7.7685	5.0705	8.8775	12.7594	-0.6583	6.2755	-0.7663	-1.3397	-17.7707
3	8.1779	2.5829	4.6242	6.4974	10.2278	-4.4457	0.5285	0.0805	0.0364	-18.1753
	.	.	.	.	.	.	.	.	.	.
7	3.3207	0.9828	5.5026	9.0044	10.0003	-0.998	0.1752	-0.0032	-0.0092	-17.7491
8	3.3091	0.9988	5.4997	8.9995	10.0016	-1.0036	0.1725	-0.0004	-0.0008	-17.7499
9	3.3	1.0006	5.4996	8.9984	10.0001	-1.0018	0.17	0	0	-17.7501
10	3.2992	1.0002	5.5	8.9997	10	-1.0004	0.1697	0	0.0002	-17.7499
11	3.2995	0.9998	5.5001	8.9998	10	-1.0003	0.1698	0	-0.0001	-17.7499
12	3.2997	0.9997	5.5001	8.9998	10	-1.0003	0.1698	0	-0.0001	-17.7499
.	.	.	.	.	.	.	.	.	.	.
39	3.2997	0.9998	5.5001	8.9998	10	-1.0004	0.1698	0	-0.0001	-17.7499
40	3.2997	0.9998	5.5001	8.9998	10	-1.0004	0.1698	0	-0.0001	-17.7499

6. ERROR ANALYSIS AND DISCUSSION

In Example 1, the Jacobi method starts with significant initial errors but shows steady convergence, reaching near-zero errors by the iteration, though with oscillatory behavior. The Gauss-Seidel method, while starting with similar errors, achieves faster convergence, reducing errors to near-zero by the iteration.

In Example 2, the Jacobi method experiences larger initial fluctuations, with errors persisting until the iteration despite some reduction after the iteration. The Gauss-Seidel method, in contrast, begins with smaller initial errors and rapidly eliminates them, achieving near-zero errors by the iteration.

In Example 3, the Jacobi method results in considerable errors initially and fails to fully converge even after 10 iterations. The Gauss-Seidel method, however, converges significantly faster, stabilizing the solution within the first few iterations and requiring fewer steps overall.

Figures 1, 2, and 3 illustrate the error reduction per iteration for each method. The graphs in these figures clearly show that the Jacobi method (blue line) converges more slowly than the Gauss-Seidel method (orange line). The Gauss-Seidel method required fewer iterations to approach the solution and delivered more stable results.

Table 6. Errors in the iteration of Jacobi and Gauss-Seidel methods for Example 1

Iteration No	**Errors in Jacobi iteration**		**Errors in Gauss-Seidel iteration**	
1	2	1.5	2	-1
2	0.75	-1	-0.5	0.25
3	-0.5	-0.375	0.125	-0.0625
.	.	.	.	.
8	-0.0117	0.0156	-0.0001	0.0001
9	0.0078	0.0059	0	0
10	0.0029	-0.0039	0	0
11	-0.002	-0.0015	0	0
12	-0.0007	0.001	0	0
13	0.0005	0.0004	0	0
14	0.0002	-0.0002	0	0
15	-0.0001	-1E-04	0	0

Table 7. Errors in the iteration of Jacobi and Gauss-Seidel methods for Example 2

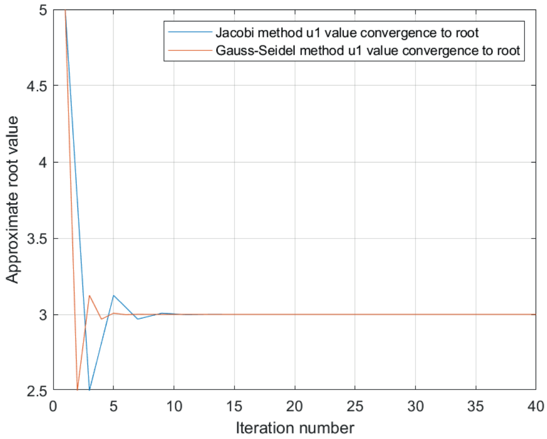
Iter.	**Errors in Jacobi iteration**				**Errors in Gauss-Seidel iteration**			
No	u_1	u_2	v_1	v_2	u_1	u_2	v_1	v_2
1	0.6	2.25	1.8	0.5	0.6	0.95	0.11	-0.38
2	1.56	-0.5	-1.27	-1.875	0.174	-0.3045	-0.0041	0.1978
3	-1.579	-1.4	-0.762	1.275	-0.0039	0.1019	0.0188	-0.0613
7	-0.2568	1.1722	1.3322	0.8122	-0.0003	0.0017	0.0002	-0.001
8	1.2226	0.2015	-0.327	-1.3164	0.0001	-0.0006	-1E-04	0.0004
9	-0.7747	-1.1878	-0.892	0.3685	0	0.0002	0	-0.0001
10	-0.4324	0.7945	1.015	0.8462	0	-1E-04	0	0
.	.	.	.	.	.	.	.	.
38	-0.045	0.13	0.1547	0.1088	0	0	0	0
39	0.1482	0.0382	-0.0255	-0.1536	0	0	0	0
40	-0.0819	-0.1445	-0.114	0.0307	0	0	0	0

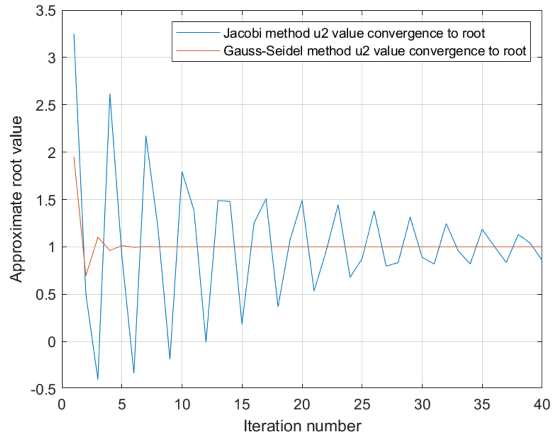
Table 8. Errors in the iteration of the Jacobi method for Example 3.

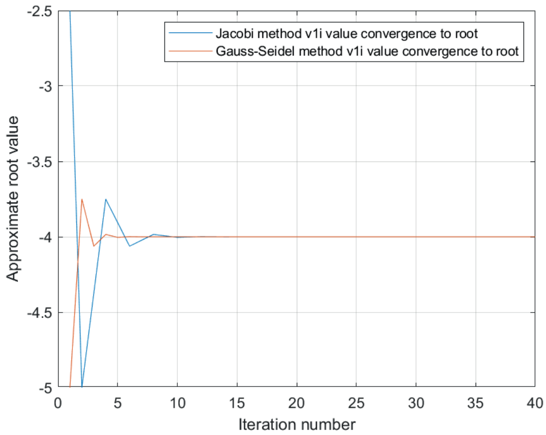
Iter. No.	Errors in Jacobi Iteration									
	u_1	u_2	u_3	u_4	u_5	v_1	v_2	v_3	v_4	v_5
1	0.6223	3.8758	11.2945	9.1808	0	1.4474	0.4432	0.2956	15.9806	1.5301
2	2.3264	8.4886	1.1862	0.2186	2.717	2.9378	1.3255	0.1675	1.1345	1.5609
3	5.3198	0.6415	0.0118	2.4373	0.1929	0.684	1.69	1.6613	1.3932	0.0372
.	.	.	.	.	.	.	.	.	.	.
7	0.0909	0.0371	0.0011	0.057	0.0027	0.0392	0.0411	0.0377	0.0318	0.0026
8	0.0344	0.0093	0.0015	0.0017	0.0054	0.0231	0.0221	0.0041	0.0031	0.0097
9	0.0058	0.0063	0.0057	0.0048	0.0005	0.0127	0.0056	0.0002	0.0087	0.0003
10	0.0035	0.0033	0.0006	0.0006	0.0015	0.0052	0.0015	0.0003	0.0001	0.0008
11	0.0019	0.0008	0	0.0013	0	0.0009	0.001	0.0009	0.0007	0.0001
12	0.0008	0.0002	0.0001	0	0.0001	0.0006	0.0005	0.0001	0.0001	0.0002
13	0.0002	0.0002	0.0001	0.0001	0	0.0003	0.0001	0	0.0002	0
14	0.0001	0.0001	0	0	0	0.0001	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0

Table 9. Errors in the iteration of the Gauss-Seidel method for Example 3

Iter No.	Errors in Gauss-Seidel Iteration									
	u_1	u_2	u_3	u_4	u_5	v_1	v_2	v_3	v_4	v_5
1	0.6223	3.8758	11.2945	8.846	0	2.5616	1.104	5.9674	16.2299	1.504
2	3.8352	8.7683	0.4296	0.1223	2.7594	0.3421	6.1057	0.7662	1.3396	0.0208
3	4.8782	1.5831	0.8759	2.5024	0.2278	3.4454	0.3587	0.0805	0.0365	0.4255
.	.	.	.	.	.	.	.	.	.	.
7	0.021	0.017	0.0025	0.0046	0.0003	0.0024	0.0054	0.0032	0.0092	0.0008
8	0.0094	0.0009	0.0003	0.0003	0.0016	0.0033	0.0026	0.0004	0.0007	0.0001
9	0.0003	0.0008	0.0005	0.0014	0.0001	0.0014	0.0001	0.0001	0.0001	0.0002
10	0.0005	0.0004	0.0001	0.0001	0	0	0.0001	0.0001	0.0002	0
11	0.0002	0	0	0	0	0.0001	0.0001	0	0	0
12	0	0	0	0	0	0	0	0	0	0
.	.	.	.	.	.	.	.	.	.	.
15	0	0	0	0	0	0	0	0	0	0







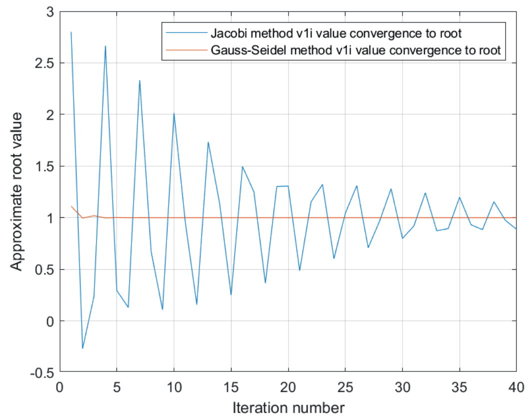
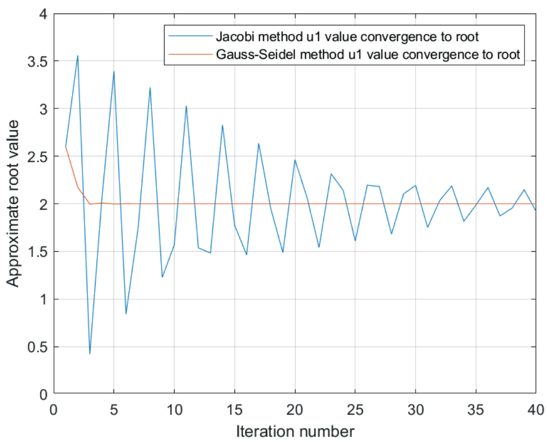
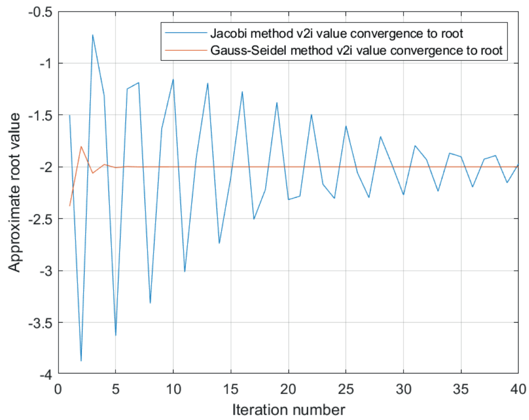
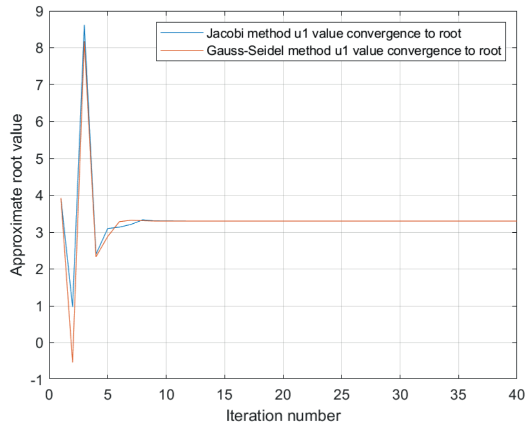


Figure 1. Root convergence graphs; Example 1.







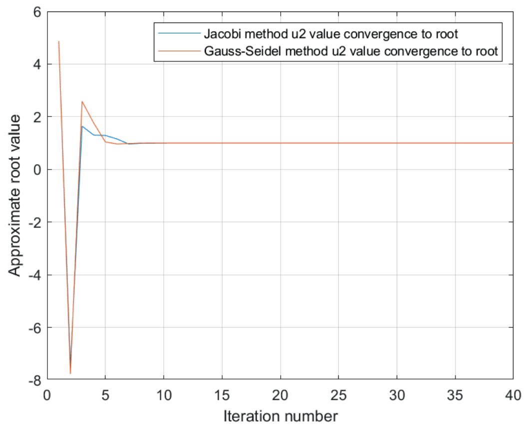
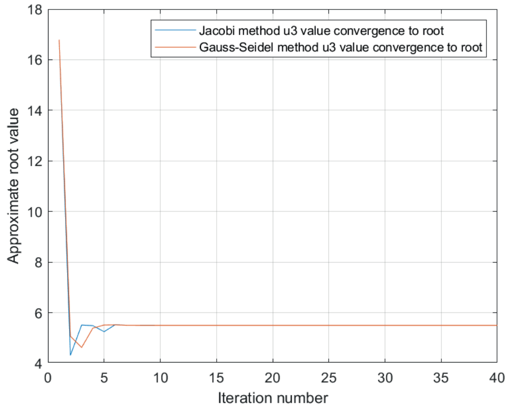
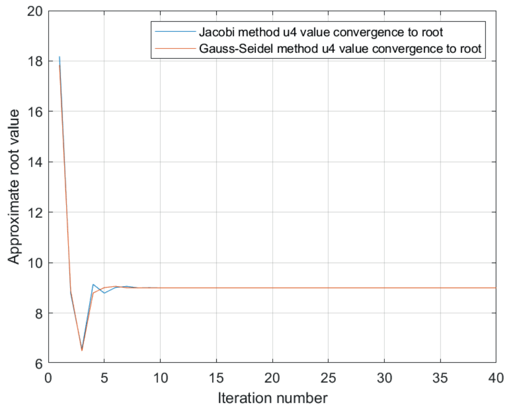
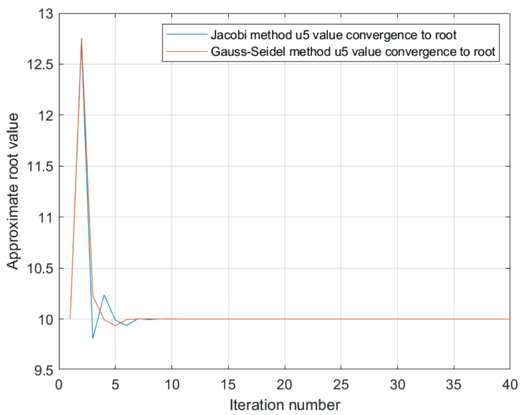
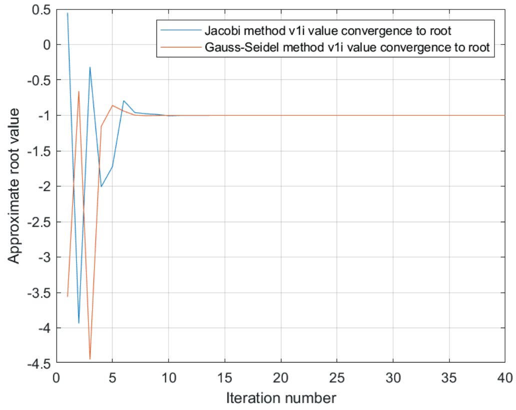


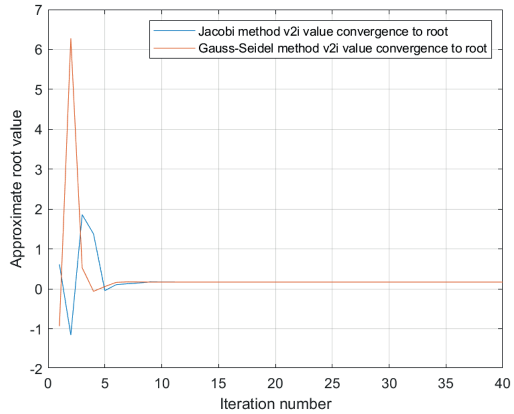
Figure 2. Root convergence graphs; Example 2.

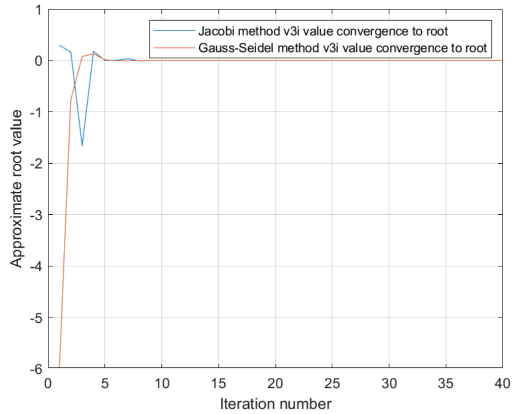


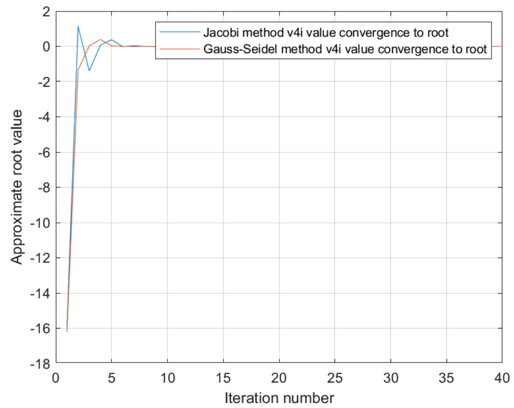












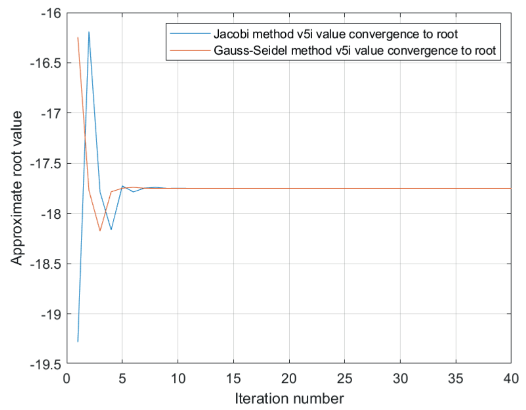


Figure 3. Root convergence graphs; Example 3.

7. CONCLUSION

This study compares the Jacobi and Gauss-Seidel methods for solving systems of linear equations with complex coefficients, focusing on convergence speed and error reduction. Numerical analyses reveal that the Gauss-Seidel method consistently outperforms the Jacobi method across various system sizes, including small (2x2) and larger (5x5) systems. The Gauss-Seidel method achieves faster convergence and greater stability, with errors diminishing to zero in fewer iterations, regardless of system size. In contrast, the Jacobi method shows slower convergence and fluctuating error reductions, particularly for larger systems.

These findings emphasize the efficiency and reliability of the Gauss-Seidel method for solving complex linear systems, particularly in large-scale applications where computational efficiency and stability are critical. While the Jacobi method remains applicable for smaller or less complex systems, the superior performance of the Gauss-Seidel method makes it the preferred choice for engineering and scientific computations.

Future work could explore further enhancements to both methods, focusing on optimizing their application to larger, more complex systems and assessing their performance in parallel computational environments. This study highlights the Gauss-Seidel method as an effective and practical tool for tackling computational challenges in linear algebra.

Author Contribution Statement:

The contributions of the authors in the study were as follows: Author 1 contributed to the creation of the idea, preparation of the theoretical background of the article, literature review, evaluation of the results and writing of the article, Author 2 contributed to the calculations and analyses, preparation of the theoretical background of the article, literature review, evaluation of the results and spell check, and Author 3 contributed to the content control of the article.

REFERENCES

[1] Wong SM. Computational methods in physics and engineering. 2nd ed. World Scientific; 2003.

[2] Ahmadi A, Manganiello F, Khademi A, Smith MC. A parallel Jacobi-embedded Gauss-Seidel method. IEEE Trans. Parallel Distrib. Syst. 2021;32(6):1452-1464.

[3] Saad, Y., Van Der Vorst, H. A. (2000). Iterative solution of linear systems in the 20th century. Journal of Computational and Applied Mathematics, 123(1-2), 1-33.

[4] Kawai, M., Iwashita, T., Nakashima, H., & Marques, O. (2012). Parallel smoother based on block red-black ordering for multigrid Poisson solver. In International Conference on High Performance Computing for Computational Science (pp. 292-299). Berlin, Heidelberg: Springer Berlin Heidelberg.

[5] Shang, Y. (2009). A distributed memory parallel Gauss–Seidel algorithm for linear algebraic systems. Computers & Mathematics with Applications, 57(8), 1369-1376.

[6] Chikaedum AP, Grace EN. Comparative analysis of the Gauss-Seidel and Jacobi methods for solving linear systems. COOU J Phys Sci. 2022;5(1):60-79.

[7] Liu H, Feng T. Study on the convergence of solving linear equations by Gauss-Seidel and Jacobi method. In: Proceedings of the 11th International Conference on Computational Intelligence and Security (CIS). IEEE; 2015. p. 100-103.

[8] Dihoum, B. A., Flijah, L. A. A., Qiblawi, S. S., Owen, S. A. (2022). Comparison Of Jacobi Iteration Method And Gauss-Seidel Iteration Method In Solving Fuzzy

Linear Equation Systems Using A Computer, 13(2), 711-722.

[9] Khrapov, P. V., & Volkov, N. S. (2024). Comparative analysis of Jacobi and Gauss-Seidel iterative methods. International Journal of Open Information Technologies, 12(2), 23-34.

[10] Axelsson O, Neytcheva M, Ahmad B. A comparison of iterative methods to solve complex-valued linear algebraic systems. Numer Algorithms. 2014; 66:811-841.

[11] Ercegovac M, Muller JM. (2007). Solving systems of linear equations in complex domain: Complex E-method. ⟨ensl-00125369v2⟩.

[12] Militaru R, Popa I. On the numerical solving of complex linear systems. Int J Pure Appl Math. 2012;76(1):113-122.

[13] Burden RL, Faires JD, Numerical analysis. 9th ed. Brooks/Cole Learning: Boston; 2010.

[14] Bhartendu [Internet]. Gauss-Seidel Method, Jacobi Method, 2024 [cited 2024 Nov 25]. Available from: https://www.mathworks.com/matlabcentral/fileexchange/63167-gauss-seidel-method-jacobi-method