

Published by: IOCSCIENCE

International Journal of Basic and Applied Science

Journal homepage: www.ijobas.pelnus.ac.id



Stationary and non-stationary method for solving system of linear equation

Ferdinan Rinaldo Tampubolon¹, Sinta Marito Siagian², Samaria Chrisna³, Rischa Devita⁴, Indah Nurhidayati⁵

^{1,4}Jurusan Teknik Elektro, Politeknik Negeri Medan, Indonesia ⁵Nautika, Politeknik Ilmu Pelayaran, Indonesia

Article Info

Article history:

Received Apr 13, 2023 Revised May 18, 2023 Accepted May 30, 2023

Keywords:

Conjugate; Gauss-seidel; Indirect; Jacobi; Non-stationary.

ABSTRACT

System of Linear Equation Ax = b where A is a non-singular and square matrix $(det(A) \neq 0)$. Method for solving System of Linear Equation consist of direct method and indirect method. Further indirect methods were divided into two, that is stationary and non-stationary. This research will conduct a comparative study of several indirect methods and direct methods in solving several cases of Linear equation systems. Some methods that will be compared in this research are jacobi, gauss-seidel, SOR, conjugate and biconjugate gradient. Testing several methods for some kind of matrix is useful to understand the characteristics of each method in solving different types of matrices. The result show that non-stationary such as conjugate and biconjugate has a less computation and faster to convergence compared to stationary method for several symmetric and non-symmetric matrices.

This is an open access article under the CC BY-NC license.



Corresponding Author:

Ferdinan Rinaldo Tampubolon, Department of Electrical Engineering, Politeknik Negeri Medan, Jl. Almamater No.1, Padang Bulan, Kota Medan, Sumatera Utara 20155, Indonesia, Email: ferdinantampubolon@polmed.ac.id

1. INTRODUCTION

System of Linear Equation Ax = b where A is a non-singular and square matrix ($det(A) \neq 0$). The n denoted a vector, and x denoted solution vector. Each entries of A and b could be either a real number or complex number. Then the solution of x-vector:

$$x = A^{-1}b$$

 A^{-1} is the invers of A. Unfortunately in so many cases A^{-1} had been known. [1]. System of Linear Equation have so many application in some fields that is electrical circuit [2], chemical balance [3], and economic or financial [4, 5]. Method for solving System of Linear Equation consist of direct method and indirect method. *Direct method* provide a solution after performing amount of calculation which is known exactly, some popular direct methods are elimination, cramer and gauss elimination. Gauss Elimination transform a whole system of linear equation to be upper triangular matrix by doing only row elementary operation [6]. Gauss Elimination required less computation, easier to get the rank of a matrix [7], and more effective than the other elimination method for solving large variables [8]. In contrast with direct method, indirect method make an approach to the solution, doing some iteration process for making the solution converge based on the desirable accuracy. Therefore the amount of computation for indirect method depends on desirable accuracy.

Generally, indirect method is preferable in solving System of Linear Equation with large variable because direct method such as Gauss Elimination allows some zero elements to be non-zero, which requires additional memory [9]

Further indirect methods were divided into two, that is stationary and non-stationary. Stationary method is easier to understand because it only involves one or two information that changes at each iteration, hence less computing resources are used [10], some stationary methods are Jacobi [11], Gauss-Seidel [12], and SOR [13]. In general, the SOR method has faster convergence than Gauss-Seidel but it is hard to determine the relaxation value [14]. Meanwhile, non-stationary involves more information that changes at each iteration and also multiplication of residuals or other vectors that appear at each iteration [15]. Some examples of Non-Stationary methods are Conjugate Gradient [16] and biconjugate gradient [17]. This method solves a system of linear equations with a number of iterations smaller than the matrix dimension, making it suitable for large matrices [18]. Conjugate Gradient is known as one of the best iteration methods in solving linear equation systems for symmetric matrices [19], This method works not only with real numbers but also with complex numbers [20]. Biconjugate Gradient is a generalized form of Conjugate Gradient that can solve the non-symmetric matrix.

This research will conduct a comparative study of several indirect methods and direct methods in solving some cases of Linear equation systems. Some Researches related to the use of direct methods: [21] which discussed the comparison of gauss-seidel and jacobi methods, [22] discusses the jacobi, gauss-seidel and SOR methods in the case of a 4x4 matrix, But, both studies did not involve non-stationary methods, research related to comparisons involving non-stationary, was done by [23] which discusses the conjugate gradient, gauss-seidel and jacobi methods. Unfortunately that research has not involved a test for non-symmetrical matrix, larger scale matrix and sparse matrix. Some methods that will be compared in this research are jacobi, gauss-seidel, SOR, conjugate and biconjugate gradient. Testing several methods for some kind of matrix is useful to understand the characteristics of each method in solving different types of matrices.

2. RESEARCH METHOD

This research will utilize five methods to solve several cases of Linear Equation System. That five methods are:

2.1. Jacobi Method

Suppose a System of Linear Equation Ax = b, where A is an $n \times n$ matrix, x is $n \times 1$ a solution vector and dan b is a $n \times 1$ vector. Then the solution will be updated as follow:

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

This method will be converge if the diagonal of A matrix has non-zero entries and diagonally strong dominant. $|a_{ii}| \ge \left|\sum_{j=1}^n a_{ij}\right|$ [24]

The step of Jacobi Method

```
Metode Jacobi

Determine the initial value x_0, maximum iteration k_n, and desirable accuracy \varepsilon

for k=1: k_n

for i=1: n

sum=0

for j=1: n; j\neq i

sum=sum+a(i,j)*x_0(j)

end

x\_new=(b(i)-sum)/a(i,i)
end

if |x\_new-x_0|<\varepsilon

break

end

x_0=x\_new
end
```

2.2. Gauss-Seidel Method

Gauss-Seidel Method is a development of jacobi method, where the x-vector will be updated using this formula:

$$x_i(k+1) = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j(k+1) - \sum_{j=i+1}^{n} a_{ij} x_j(k) \right)$$
 (2)

Gauss Seidel will be converged if diagonal matrix A have non-zero entry

The step of Gauss-Seidel

```
Gauss-Seidel Method
Determine the initial value x_0, maximum iteration k_n, and \overline{\text{desirable accuracy } \varepsilon}
for k = 1: k_n
    for i = 1:n
         sum = 0
         for j = 1: n ; j \neq i
              sum = sum + a(i,j) * x(j)
         end
         for j = 1: n ; j \neq i
              sum = sum + a(i,j) * x_0(j)
         end
    x(i) = (b(i) - sum)/a(i, i)
    end
   if |x - x_0| < \varepsilon
         break
     x_0 = x
end
```

2.3. Succesive Over Relaxation Method (SOR)

Succesive Over Relaxation Method is another form of Gauss Seidel Method with the additional variable w, so the update vector x will be:

$$x_i(k+1) = (1-w)x_i(k) + \frac{w}{a_{ii}} \left(b_i - \sum_{j < i} a_{ii}x_j(k+1) - \sum_{j > i} a_{ii}x_j(k) \right)$$
(3)

If the relaxation value w=1 then SOR method will have the same formula with Gauss-Seidel Method. This method will be converge if the diagonal matrix A has non zero entries. The relaxation value w has the inverval value 0 < w < 2. [25]

The Step of SOR Method.

```
Succesive Over Relaxation Method
Determine the initial value x_0 = x, maximum iteration k_n, relaxation value w, and desirable accuracy \varepsilon
for k = 1: k_n
    for i = 1:n
         sum = 0
         for j = 1: n ; j \neq i
             sum = sum + a(i,j) * x(j)
         for j = 1: n ; j \neq i
             sum = sum + a(i,j) * x_0(j)
    sum1 = (b(i) - sum)/a(i, i)
    x_i = x_0(i) + (w * (sum1 - x_0(i)))
    end
   if |x - x_0| < \varepsilon
         break
    end
     x_0 = x
end
```

2.4. Conjugate Gradient Method

The solution vector *x* is updated as follows:

$$x_i(k+1) = x_i + \alpha_i d_i \tag{4}$$

where α_i

$$\alpha_i = \frac{r_i^T d_i}{d_i^T A d_i} \tag{5}$$

Residual value r_i is updated as follows:

$$r_i(k+1) = r_i + \alpha_i A d_i \tag{6}$$

Search direction d_i is updated as follows:

$$d_i(k+1) = r_i(k+1) + \beta_i d_i \tag{7}$$

The value of β_i

$$\beta_i = -\frac{r_i^T (k+1) A d_i}{d_i^T A d_i} \tag{8}$$

This method will be converge on symmetric matrices [26]

The Step of Conjugate Gradient Method.

```
Metode Conjugate Gradient
Determine the initial value x_0, maximum iteration k_n, and desirable accuracy \varepsilon
r_0 = b - A * x_0
d = r_0
      for k = 1: k_n
           \alpha = (r_0^T * r_0)/d_0^T * A * d
           x_{new} = x_0 + \alpha * d
           r_{new} = r_0 - \alpha * d
           if r_{new} < \varepsilon
                 break
           \beta = (r_{new}^T * r_{new})/(r_0^T * r_0)
d = r_{new} + \beta * d
           r=r_{new}
           x_0 = x_{new}
      end
```

2.5. Biconjugate Gradient Method

The Biconjugate Gradient method can be applied to solve non-symmetric Linear Equation Systems. This method has the same process as Conjugate Gradient but has additional value of x_i^T , r_i^T , d_i^T

$$x_{i}(k+1) = x_{i} + \alpha_{i}d_{i} , x_{i}^{T}(k+1) = x_{i}^{T} + \alpha_{i}d_{i}^{T}$$

$$r_{i}(k+1) = r_{i} - \alpha_{i}Ad_{i}, r_{i}^{T}(k+1) = r_{i}^{T} + \alpha_{i}^{T}A_{i}^{T}d_{i}^{T}$$

$$(9)$$

$$(10)$$

$$r_i(k+1) = r_i - \alpha_i A d_i, \quad r_i^T(k+1) = r_i^T + \alpha_i^T A^T d_i^T$$
(10)

$$d_i(k+1) = r_i(k+1) + \beta_i d_i, \qquad d_i^T(k+1) = r_i^T(k+1) + \beta_i d_i^T$$
(11)

Where,

$$\alpha_i = \frac{r_i^T (k+1) r_i (k+1)}{d_i^T A d_i}, \qquad \beta_i = \frac{r_i^T (k+1) r_i (k+1)}{r_i^T r_i}$$
(12)

For symmetrical matrix the biconjugate gradient method will produce the same results as the conjugate gradient method but has twice as much calculation. [27]

The Step of Biconjugate Gradient

```
Biconjugate Gradient Method
Determine the initial value x_0, maximum iteration k_n, and desirable accuracy \varepsilon
r_0 = b - A * x_0
r_0^T = b^T - x_0^T * A
d = r_0d^T = r^T
       \mathbf{for}\: k=1{:}\: k_n
              \alpha = (r_0^T * r_0)/d_0^T * A * d
              x_{new} = x_0 + \alpha * d
x_{new}^T = x_0^T + \alpha * d^T
              r_{new} = r_0 - \alpha * A * d
r_{new}^T = r_0^T - \alpha * d^T * A
              if |r_{new}| < \varepsilon or |r_{new}^T| < \varepsilon
                     break
```

```
end
\beta = (r_{new}^T * r_{new})/(r_0^T * r_0)
d = r_{new} + \beta * d
d^T = r_{new}^T + \beta * d^T
r_0 = r_{new}
r_0^T = r_{new}^T
x_0 = x_{new}
x_0^T = x_{new}^T
end
```

3. RESULTS AND DISCUSSIONS

Some System of Linear Equation cases will be investigated in this research. Five methods will be used for solving System of Linear Equation with desirable accuracy e=0,00001, initial value $x_1, x_2, x_3, x_4 = 0$ using Matlab R2021a.

3.1. Case 1

The case of 4x4 matrix with non-symmetrical matrix as follows:

$$4x_1 + x_2 + 2x_3 - x_4 = 2$$

$$3x_1 + 6x_2 - x_3 + 2x_4 = -1$$

$$2x_1 - x_2 + 5x_3 - 3x_4 = 3$$

$$4x_1 + x_2 - 3x_2 - 8x_4 = 2$$

The solution obtained using Jacobi and Gauss-Seidel method is presented on Table 1 and Table 2

Table 1. The Solution using Jacobi Method (Case 4 x 4)

Iterasi	x_1	x_2	x_3	χ_4
1	0.5000	-0.1667	0.6000	-0.2500
2	0.1792	-0.2333	0.2167	-0.2458
	•			
22	0.3650	-0.2338	0.2851	-0.2036
23	0.3650	-0.2338	0.2851	-0.2036

Table 2. The Solution using Jacobi Method Gauss-Seidel (Case 4 x 4)

Iterasi	x_1	x_2	x_3	x_4
1	0.5000	-0.4167	0.3167	-0.1708
2	0.4031	-0.2585	0.2845	-0.1875
8	0.3650	-0.2338	0.2851	-0.2036
9	0.3650	-0.2338	0.2851	-0.2036

The Gauss-Seidel method has better performance compared to Jacobi method, Table 2 shows that Gauss-Seidel converges at the 9th iteration while the Jacobi method converged at 23th iteration as shown in Table 1. The solution with the SOR method with a value of w = 1.25 is presented in Table 3

Table 3. The Solution using SOR Method (Case 4 x 4)

Iterasi	x_1	x_2	x_3	x_4
1	0.6000	-0.5600	0.2976	-0.1579
2	0.4221	-0.2186	0.2917	-0.1792
14	0.3650	-0.2338	0.2851	-0.2036
15	0.3650	-0.2338	0.2851	-0.2036

Table 3 shows that SOR Method converge at 15th iteration which is worse than the Gauss-Seidel Method. But the number of iterations obtained by the SOR method is affected by the relaxation value w, o<w<2. Here are some numbers of iterations with various values of w within the interval [0,2], and

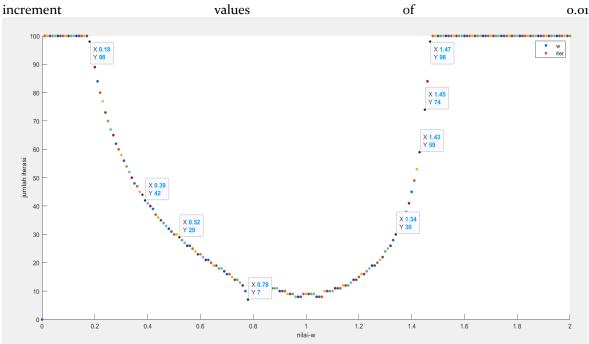


Figure 1. The Amount of Iteration of each value w (Case 4 x 4)

Figure 1 shows the number of iterations in the SOR method is significantly affected by the relaxation value. At most relaxation values o < w < 2 (denoted as x in figure 1) the SOR method has a larger number of iterations than gauss-seidel (denoted as y in figure 1). SOR method has fewer iterations than gauss-seidel when using the relaxation value w = 0.78 that is 7 iterations as compare to 9 iterations in gauss-seidel. Since the matrix is a non-symmetrical matrix, conjugate gradient cannot be used, so biconjugate gradient is used to solve that cases as in Table 4.

Table 4. The Solution using Biconjugate Method (Case 4 x 4)

Iterasi	x_1	x_2	x_3	χ_4
1	1.3333	-0.6667	2.0000	1.3333
2	0.2789	-0.0198	0.3724	-0.2548
3	0.2766	-0.0385	0.3675	-0.2554
4	0.3650	-0.2338	0.2851	0.2036

Table 4 shows that Biconjugate Gradient has the least iteration that is 4 iteration

3.2. Case 2

This case is a 5x5 matrix, which is a non-symmetric matrix. The results obtained are shown in Figure 2.

$$6x_1 - 4x_2 - 2x_4 = 12$$

$$15x_1 - 31x_2 + 10x_3 + 6x_5 = 57$$

$$2x_2 - 11x_3 + 6x_4 = -12$$

$$3x_1 + 12x_3 - 19x_4 = 0$$

$$4x_2 - 9x_5 = -8$$

Figure 3. The Amount of Iteration of each method (Case 5 x 5)

Figure 2 shows the number of iterations and the accuracy value obtained. Each method will be considered convergent when the accuracy value is 0.0001. Biconjugate Gradient method is quicker to converge at the 4th iteration, followed by SOR method (w=1.25) at the 19th iteration, Gauss-Seidel method at the 35th iteration and Jacobi at the 70th iteration. This indicates for this 5 x 5 matrix case, non-stationary method, biconjugate gradient, is much faster to converge than the stationary method. While for the Stationary method, the SOR method is faster to converge than the Gauss-Seidel and Jacobi methods.

The other cases are symmetric matrix which obtained from some research such as 3x3 by [28],8 x 8 sparse matrix by [13], dan several symmetric matrix with random variable generated using sprand function in Matlab. The result is summarized on Table 5.

Kasus		Jacobi	Gauss Seidel	SOR	Conjugate/Biconjugate
Matrix 3 x 3	Iteration	53	27	13	3
	Time	0.008913	0.016742	0.003613	0.002780
Matrix 4 x 4	Iteration	23	9	7	4
Non-Symmetric	Time	0.002976	0.002331	0.007187	0.009751
Matrix 5 x 5	Iteration	70	35	19	5
Non-Symmetric	Time	0.020981	0.006796	0.006008	0.001220
Matrix 6 x 6	Iteration	-	5274	3289	6
	Time	-	1.007868	1.778296	0.002618
Matrix 8 x 8	Iteration	-	13	13	4
Sparse Matrix	Time	-	0.002447	0.001114	0.004604
Matrix 10 x 10	Iteration	-	4240	2638	11
	Time	-	1.488664	2.812353	0.003167
Matrix 13 x 13	Iteration	-	3951	2431	14
	Time	-	1.859279	3.358919	0.002826
Matrix 15 x 15	Iteration	-	1502	901	16
	Time	-	0.878366	1.903691	0.002259
Matrix 20 x 20	Iteration	-	3655	2093	24
	Time	-	3.620163	5.107883	0.003713

Table 5. The Solution of Several Cases

From the Table 5, it can be observed that the non-stationary method have the least number of iterations and computation time except in the case of 4x4 matrices. In this case the biconjugate method

which have more computation than conjugate gradient is used because conjugate gradient cannot be implemented for non-symmetric matrix.

Meanwhile, the SOR method with w = 1.25 has fewer iterations than the Gauss-Seidel method but in some cases have longer computation time, this happens because the SOR method has more variables and more calculations.

The SOR method has different amount of iteration when calculated with various relaxation values of w, however, with several relaxation values which have been used, the number of iterations obtained is still larger compared to the number of iterations of non-stationary method. Figure 3 shows the results obtained for the 20x20 case. The least number of iterations in SOR is 1127 with a relaxation value of 1.38. This result is much fewer than the gauss-seidel method which is 3655 iteration. Nevertheless, this still larger than number of iterations using the non-stationary method, conjugate gradient. This method has only 24 iterations. For this case, it appears that the non-stationary method converges faster than the stationary method.

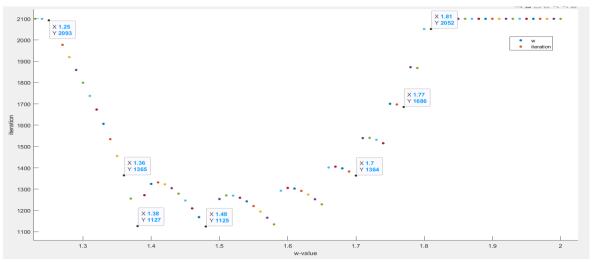


Figure 3. The Amount of Iteration of each value w (Case 20 x 20)

4. CONCLUSION

Although having a more complex computation, Non-Stationary methods such as Conjugate and Biconjugate have less iteration and faster convergence compared to stationary methods in several cases. For stationary method, SOR method has the least computation followed by Gauss-Seidel, and Jacobi. The performance of SOR method is affected by the relaxation factor w. But experiments show that even if we used the best relaxation value, it is still inferior to the non-stationary method. However, the use of a relaxation factor that is not static but changes over several iterations may give different results and is worth to be tested for future research. In addition, despite Biconjugate Gradient has less iterations than other non-stationary methods and can be used on both symmetric and non-symmetric matrices, it still has drawbacks. This method uses a lot of computation so it is possible to consume more time than stationary methods in particular cases. This method seems to perform the conjugate gradient calculation twice, so for future research it is recommended to explore other non-stationary methods and compare the number of iterations and time used in solving the system of linear equations.

REFERENCES

- [1] J. C. Butcher, The numerical analysis of ordinary differential equations: Runge-Kutta and general linear methods. Wiley-Interscience, 1987.
- [2] N. Nurullaeli, "Media Analisis Rangkaian Listrik Menggunakan Pendekatan Numerik Gauss-Jordan, Gauss-Seidel, dan Cramer," *Navigation Physics: Journal of Physics Education*, vol. 2, no. 1, pp. 1-8, 2020.

- [3] I. Hamid, "Balancing Chemical Equations by Systems of Linear Equations," *Applied Mathematics*, vol. 10, no. 7, pp. 521-526, 2019.
- [4] R. Saikia and D. Sarma, "A Case study on an Economic problem by using Fuzzy linear Equations," *IJSRSET*, vol. 1, no. 6, pp. 391-394, 2015.
- [5] A. Mandal, B. Cherukuru, and P. Y. Rani, "A STUDY OF INVESTIGATING THE BEST METHOD TO SOLVE LINEAR SYSTEM OF EQUATIONS AND ITS APPLICATIONS."
- [6] L. N. Trefethen and D. Bau, Numerical linear algebra. Siam, 2022.
- [7] S. W. a. N. X. Haoyu Luo, "Three methods for solving systems of linear equations: Comparing the advantages and disadvantages," *Journal of Physics: Conference Series*, vol. 2012, 2021.
- [8] R. Khan, S. Gharib, and S. R. Ali, "System of linear equations, Guassian elimination," *Global Journal of Computer Science and Technology*, vol. 15, no. C5, pp. 23-26, 2015.
- [9] R. Mittal and A. Al-Kurdi, "Application of the Cramer rule in the solution of sparse systems of linear algebraic equations," *Journal of computational and applied mathematics*, vol. 136, no. 1-2, pp. 1-15, 2001.
- [10] S. A. Meligy and I. Youssef, "Relaxation parameters and composite refinement techniques," *Results in Applied Mathematics*, vol. 15, p. 100282, 2022.
- [11] F. Aryani and L. T. Lestari, "Metode Gauss-Seidel dan Generalisasi Gauss-Seidel untuk Menyelesaikan Sistem Persamaan Linear Kompleks (Contoh Kasus: SPL Kompleks dengan 4 persamaan dan 4 variabel)," *Jurnal Sains Matematika dan Statistika*, vol. 2, no. 2, pp. 21-31, 2016.
- [12] A. Ramadhan and A. Sirait, "Generalisasi metode Gauss-Seidel Untuk Menyelesaikan Sistem Persamaan Linear," *Jurnal Online Mahasiswa (JOM) Bidang Matematika dan Ilmu Pengetahuan Alam,* vol. 1, no. 2, p. 8, 2014.
- [13] T. Mayooran and E. Light, "Applying the successive over-relaxation method to a real world problems," *American Journal of Applied Mathematics and Statistics*, vol. 4, no. 4, pp. 113-117, 2016.
- [14] S. Zhu, L. Wu, P. Cheng, and J. Zhou, "Application of modified iterative method to simulate rainfall infiltration in unsaturated soils," *Computers and Geotechnics*, vol. 148, p. 104816, 2022.
- [15] G. Dessalew, "Some Modified Stationary and Non-Stationary Iterative Methods for Solving System of Linear Equations," 2022.
- [16] S. G. Shareef and D. A. Sulaiman, "Conjugate Gradient Method for System of Linear Algebraic Equations," Turkish Journal of Computer and Mathematics Education (TURCOMAT), vol. 12, no. 14, pp. 5854-5865, 2021.
- [17] G. Ortega, E. M. Garzón, F. Vázquez, and I. García, "The BiConjugate gradient method on GPUs," *The Journal of Supercomputing*, vol. 64, pp. 49-58, 2013.
- [18] H. Ping, Y. Wang, C. Wei, J. Xi, T. Zhang, and Y. Gao, "DCG: An efficient Distributed Conjugate Gradient algorithm for solving linear equations in multi-agent networks," *Results in Control and Optimization*, vol. 10, p. 100213, 2023.
- [19] P. J. Martinez-Ferrer, T. Arslan, and V. Beltran, "Improving the performance of classical linear algebra iterative methods via hybrid parallelism," *Journal of Parallel and Distributed Computing*, 2023.
- [20] M. Tiwari and S. Vadhiyar, "Pipelined Preconditioned Conjugate Gradient Methods for real and complex linear systems for distributed memory architectures," *Journal of Parallel and Distributed Computing*, vol. 163, pp. 147-155, 2022.
- [21] A. Bakari and I. Dahiru, "Comparison of Jacobi and Gauss-Seidel iterative methods for the solution of systems of linear equations," *Asian Research Journal of Mathematics*, vol. 8, no. 02, p. 2018, 2018.
- [22] S. Karunanithi, N. Gajalakshmi, M. Malarvishi, and M. Saileshwari, "A Study on comparison of Jacobi, Gauss-Seidel and SOR methods for the solution in system of linear equations," *Int. J. of Math. Trends and Technology, (IJMTT)*, vol. 56, no. 4, 2018.
- [23] H. Ibrahim, H. Chinwenyi, and H. Ude, "On the Algorithmic Iterative Solutions of Conjugate Gradient, Gauss-Seidel and Jacobi Methods for Solving Systems of Linear Equations," *International Journal of Mathematical and Computational Sciences*, vol. 15, no. 11, pp. 113-118, 2021.
- [24] K. HarpinderKaur, "Convergence of Jacobi and Gauss-Seidel Method and Error Reduction Factor," *IOSR Journal of Mathematics (IOSRJM)*, 2012.
- P. Wang, S. Mou, J. Lian, and W. Ren, "Solving a system of linear equations: From centralized to distributed algorithms," *Annual Reviews in Control*, vol. 47, pp. 306-322, 2019.
- [26] P. S. S.P. Venkateshan, P. S. S.P. Venkateshan, Ed. *Computational Methods in Engineering* (Chapter 2 Solution of Linear Equations,). Academic Press, 2014.
- [27] A. El Akkraoui, Y. Trémolet, and R. Todling, "Preconditioning of variational data assimilation and the use of a bi-conjugate gradient method," *Quarterly Journal of the Royal Meteorological Society*, vol. 139, no. 672, pp. 731-741, 2013.

[28] O. T. Olu, "On The Direct and Indirect Methods of Solving Systems of Linear Equations" *Scholars Journal of Physics, Mathematics and Statistics* 2018.