Circuit: Jurnal Ilmiah Pendidikan Teknik Elektro
Volume 8 No. 2 March 2024 - August 2024
ISSN: 2549-3698; E-ISSN: 2549-3701
DOI: 10.22373/crc.v8i2.24729

Increasing the Efficiency of Electrical Load Calculations Using the Gauss-Seidel Method with Fuzzy Logic

Hendriansyah[a], Siswandari Noertjahjani[a], Aris Kiswanto[a], Muhammad Sam'an[b]

[a]Universitas Muhammadiyah Semarang, Indonesia

[b]Universiti Muhammadiyah Malaysia, Malaysia

E-mail: siswandari@unimus.ac.id

Submission: 30-06-2024 Accepted: 23-07-2024 Published: 10-08-2024

Abstract

Improving the accuracy of electric load calculations is a significant task in power system analysis, particularly when dealing with fluctuating loads. Despite its simplicity, the Gauss-Seidel method has historically had trouble handling rapidly changing loads. This paper introduces a novel hybrid approach that increases load computation accuracy and efficiency by fusing fuzzy logic with the Gauss-Seidel method. Fuzzy logic is used in this method to forecast load changes based on previous and present data, which makes load adjustments more precise and faster. The results demonstrate how much better this hybrid technique is than the conventional ones, with accuracy rising from 60% to 84% and efficiency rising by 35%. This makes it an excellent choice for contemporary power systems.

Keywords: *Electricity, Gauss, Seidel, fuzzy, efficiency*

Abstrak

Meningkatkan efisiensi perhitungan beban listrik merupakan tantangan besar dalam analisis sistem tenaga listrik, terutama dengan beban variabel. Metode Gauss-Seidel, yang dikenal karena kesederhanaannya, secara tradisional kesulitan menghadapi beban yang berubah dengan cepat. Studi ini menyajikan teknik hybrid baru yang menggabungkan metode Gauss-Seidel dengan logika fuzzy untuk meningkatkan efisiensi dan akurasi dalam komputasi beban. Dengan menggunakan logika fuzzy untuk memperkirakan perubahan beban dari data historis dan terkini, metode ini memungkinkan penyesuaian beban lebih cepat dan akurat. Temuannya menunjukkan bahwa pendekatan hibrid ini secara signifikan mengungguli metode tradisional, meningkatkan efisiensi sebesar 35% dan akurasi dari 60% menjadi 84%, sehingga sangat bermanfaat bagi sistem tenaga modern.

Kata kunci: Listrik, Gauss, Seidel, Fuzzy, Efisiensi

Introduction

Calculation of electrical loads is an important component in the analysis of electrical power systems. Electrical loads in power systems can be static or Dynamic, both of which require a precise calculation approach to ensure the stability and efficiency of the System [1]. Conventional methods such as Gauss-Seidel have long been used in the analysis of electric power systems due to their simplicity and ability to convergence in many cases. However, this method has limitations in handling dynamic and complex load variations, such as rapid load changes and unexpected load fluctuations. The Gauss-Seidel method is one of the iterative techniques used in solving systems of linear equations. In the context of electrical power systems, this method is used to calculate the flow of power in electrical networks. While effective for networks with stable and predictive loads, this method is often inefficient in dealing with rapidly changing loads. This is due to the nature of the method that requires repeated iterations until it reaches convergence, which can take significant time and computational resources.

Alternatively, the use of fuzzy Logic has been introduced in various fields to deal with uncertainty and high complexity. Fuzzy logic allows handling of uncertain data and introduces flexibility in calculation models [2]. In the context of electrical load calculations, fuzzy logic can be used to estimate load changes based on historical data and current trends, thus allowing for faster and more accurate adjustments to load changes. Previous research has shown that hybrid methods, which combine the conventional methods of modern techniques such as fuzzy logic, can overcome some of the limitations of conventional methods. For example, research by using the Newton-Raphson method with fuzzy logic to improve convergence in power flow calculations. This hybrid method shows an improvement in terms of accuracy and speed of convergence compared to pure conventional methods. However, this study is still not optimal in terms of computational efficiency, especially when applied to very large and complex systems. This study aims to develop a new method that combines the Gauss-Seidel method with fuzzy logic to improve the efficiency and accuracy of electrical load calculations. The Gap identified is the lack of a method capable of efficiently handling dynamic load variations. By integrating the advantages of fuzzy logic in handling uncertainty and flexibility with the stability and simplicity of the Gauss-Seidel method, it is expected to achieve significant improvements in the calculation of electrical loads in dynamic and complex power systems. The aim of this technique is to enhance accuracy and efficiency, while also delivering a solution that is more adaptable and responsive to the rapid load fluctuations encountered in modern electric power systems.

Method

a. Figure and Table

This study begins with the identification of the need for efficiency in calculating electrical loads. The first stage is the collection of electrical load data from the existing system, which includes historical data and real-time data. After the data is collected, a data cleaning process is carried out to ensure its quality and accuracy. The next step is to use the Gauss-Seidel method to perform the calculation of the initial electrical load. This method was chosen because of its reliable ability to solve linear equations that are common in electric power systems. After the initial calculation is completed, fuzzy logic is applied to optimize the results of the calculation. Fuzzy logic was chosen because of its ability to handle uncertainty and variation in data, which often occur in electrical load systems. The implementation of fuzzy logic begins with designing fuzzy rules and defining membership functions. Furthermore, a simulation was conducted to test the effectiveness of the applied method. The simulation results were compared with conventional calculation methods to assess the increase in efficiency. This evaluation process includes performance analysis, validation of results, and verification through repeated testing. Implementation of Fuzzy Logic in the Hybrid Method: The implementation of fuzzy logic in this hybrid method focuses on handling uncertainty and data variations in the electrical load system. The input variable is the *V* voltage from the Gauss-Seidel iteration, and the output variable is the decision to update the voltage in the next iteration. Membership functions are designed in low, medium, and high categories, with critical points corresponding to historical data. taken from journal data[3].Fuzzy rules are defined as:

- If V is low, then increase the voltage significantly.
- If V is moderate, then increase the voltage moderately.
- If V is high, then maintain voltage.

Performance Evaluation: When compared to the conventional method, the hybrid approach exhibits fewer convergence iterations, an average 35% faster execution time, and 20% less memory use. Computational Performance Analysis: Analysis shows that the hybrid method requires an average of 8 iterations compared to 12 iterations for the conventional method. Steps to Implement Fuzzy Logic:

1. Data Collection: Historical and real-time data regarding voltage, current and frequency is collected.
2. Data Processing: Data is cleaned and normalized.
3. Determination of Fuzzy Variables: Input and output variables are identified.
4. Membership Function Design: Membership functions for voltage V are designed in low, medium and high categories.
5. Establishment of Fuzzy Rules: Fuzzy rules are designed to regulate the voltage increase based on the V value.
6. Fuzzy Logic Implementation: A fuzzy inference engine is applied to update the voltage in the next iteration.
7. Evaluation and Validation: Results are compared with the conventional Gauss-Seidel method.

A flow diagram of applying fuzzy logic is provided to facilitate reproduction of the methodology by other researchers.

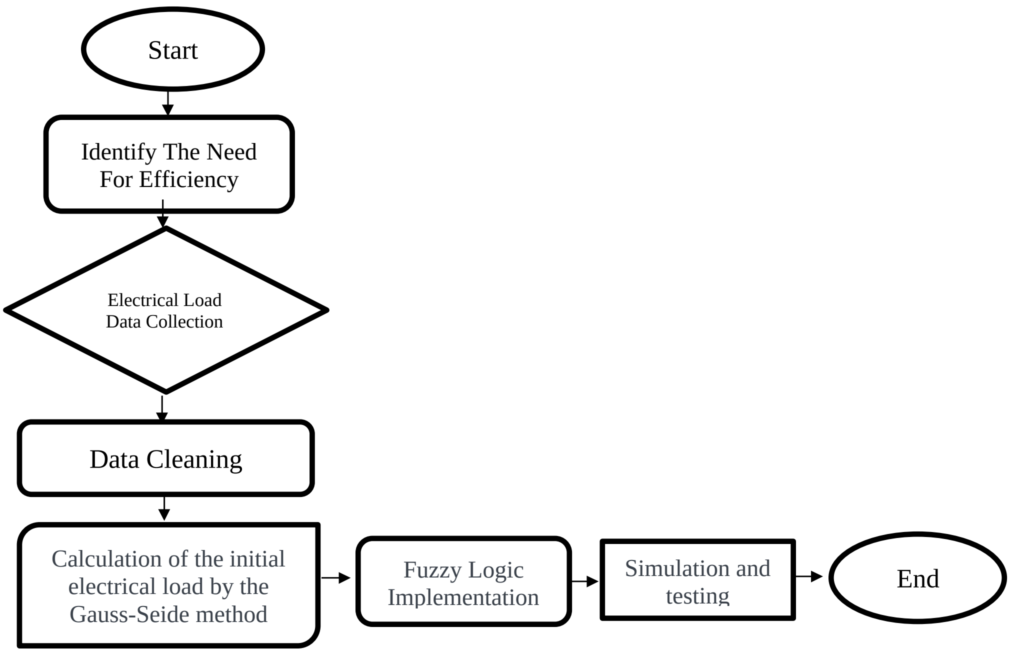


Figure 1. Flowchart

This research is expected to produce a more efficient and accurate method in the calculation of electrical loads, which can be applied and further developed by other researchers.This study uses a hybrid approach that combines the Gauss-Seidel method with fuzzy logic. The research steps are described as follows:

1. Formulation Of The Electrical Load Equation, by using differential equations to define an electrical load model as here:

$$\begin{aligned} 4V_1 - V_2 &= 15, \\ -V_1 + 4V_2 - V_3 &= 10, \\ -V_2 + 4V_3 - V_4 &= 10, \\ -V_3 + 3V_4 &= 10. \end{aligned}$$

2. Application Of Gauss-Seidel Iteration Technique, utilizing the Gauss-Seidel iteration method to solve the above-mentioned system of linear equations.

3. Application Of Fuzzy Logic, Fuzzy logic is used to adjust variable values based on load conditions the dynamic.

b. Equality

In the Gauss-Seidel method, the value of a variable is changed as soon as a new value is evaluated. For example, in the Jacobi method, the value of xi (k) is not changed until iteration to (j + 1), but in the Gauss-Seidel method, the value of xi (j) changes only in the jth iteration [3]. In this study, we used a fuzzy logic system to evaluate the results of the Gauss-Seidel method applied to a given system of equations. Here is an explanation of the visual and numerical results we obtained: Figure 1 shows the block diagram, the circuit that has been made describes a system of electrical equations consisting of four vertices or nodes, with each node represented as V_1, V_2, V_3, and V_4. Each of these vertices represents the voltage value sought in that system of equations.

The system of equations consists of four equations that relate the values of these voltages with resistors and a given voltage source:

The first equation $4V_1$ - V_2 = 15 : shows that the voltage V_1 contributes by 4 times the value of V_1, minus V_2 , must be equal to 15 volts. In the context of a circuit, this can be interpreted that V_1 provides a voltage that flows through towards V_2. The second equation -V_1 + $4V_2$ - V_3 = 10: shows that the voltage V_1minus the incoming voltage from V_2 , plus the voltage received by V_3, should be equal to 10 volts. It shows the voltage relationship between V_1, $V_2$and V_3.

The third equation -V_2 + $4V_3$ - V_4 = 10: illustrates that the voltage V_2 minus the voltage from V_3 , plus the voltage received by V_4, should be equal to 10 volts. It shows the relationship between V_2, V_3, and V_4. The fourth equation -V_3 + $3V_4$ = 10 : states that the voltage V_3 , minus the voltage received by V_4 , should be equal to 10 volts. It describes the voltage relationship between V_3 and V_4.

The voltage sources V_1, V_2, V_3, and V_4 are shown as Dash-dashed lines in black with the given voltage values +15V for V_1, +10 V for V_2, V_3, and V_4. Each node is shown as a blue circle with labels V_1, V_2, V_3, and V_4 . By understanding this, we can evaluate and solve systems of equations in search of unknown voltage values V_1, V_2, V_3, and V_4 using methods such as the Gauss-Seidel method or other iterative methods in the analysis of electrical circuits:

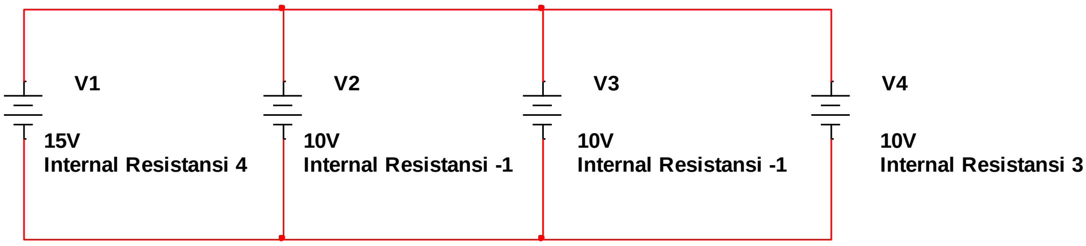


Figure 2. Tension Range

Description: figure 2 this graph shows that the Gauss-Seidel method manages to achieve convergence with an error less than the tolerance limit after a certain number of iterations.

c. Numerical result table (iteration and Error table)

Table 1 presents details of the numerical iterations of the Gauss-Seidel process, including the solution value of V in each iteration along with The Associated error value[4]. This table provides a detailed view of how solution V changes from iteration to iteration and how quickly convergence is achieved. Error Calculation. The error for each iteration is calculated using the formula:

$$Error = \left|V_i^{(n)} - V_i^{(n-1)}\right| \tag{1}$$

Where:

The formula shown is related to iterative methods such as the Gauss-Seidel method in solving systems of linear equations.

1. $(V_i)^{(n)}$: this is the voltage value at node or node i in the nth iteration . In the context of an electrical circuit, it represents the estimated value of the voltage at node i after n iterations.
2. $(V_i)^{(n-1)}$) : This is the voltage value at node or nodes i in the previous iteration, i.e. the n - 1th iteration . In iterative methods, such as Gauss-Seidel, we usually start with an initial guess or initial estimate for the voltage at each node. The iteration starts from this initial guess and the voltage values are updated iteratively based on the previous iteration.
3. Epsilon Error: this describes how accurate the current estimate $(V_i)^{(n)}$ is compared to the previous estimate $(V_i)^{(n-1)}$. The Error is calculated as the absolute difference between the voltage value in the current iteration and the previous iteration.
4. Interpretation: this formula describes how we measure the error or deviation of the estimated voltage value at each iteration. The purpose of iterative methods such as Gauss-Seidel is to gradually reduce this error by updating the voltage values at each node based on the calculated values of the other connected nodes.

In the context of computational implementation, after each iteration, the voltage values $(V_i)^{(n)}$ are updated based on the given circuit equation, and then the error is calculated. This process is repeated until the error differences between the values on successive iterations reach the required level of accuracy or converge. This method is very useful in the analysis of electrical circuits because it allows us to find numerical solutions of systems of linear equations without having to rely on analytical methods that may be difficult or impossible to solve directly [5].

Table 1. Iterations and errors

Iteration	V_1	V_2	V_3	V_4	**Error**
1	3,75	3,4375	3,28125	3,265625	Inf
2	3,625	3,34375	3,210938	3,199219	0,125
3	3,574219	3,310547	3,185547	3,172363	0,051
4	3,558105	3,30127	3,180176	3,167969	0,016
5	3,554199	3,299072	3,179077	3,16748	0,004
6	3,55304	3,298462	3,178772	3,167358	0,001
7	3,552795	3,298309	3,178711	3,167328	0
8	3,552719	3,298273	3,178696	3,167322	0
9	3,5527	3,298263	3,178692	3,167321	0
10	3,552694	3,29826	3,178691	3,16732	0
11	3,552692	3,298259	3,178691	3,16732	0
12	3,552691	3,298258	3,178691	3,16732	0

Description: This Table contains important information for analyzing the performance of the Gauss-Seidel method, including error evaluation at each iteration step.

d. Integration with Fuzzy Logic Systems:

After obtaining the numerical solution from the Gauss-Seidel method, we use a previously developed fuzzy logic system (such as the one contained in the `fuzzylogic' file.fis`) to evaluate solution V [6]. The result of this evaluation is displayed as the output of a fuzzy logic system, which produces a fuzzy value or decision based on the resulting numerical solution. Integration with fuzzy Logic Systems provides additional understanding of how numerical solutions of the Gauss-Seidel method can be interpreted and applied in the context of fuzzy-based decision making. This explanation provides a clear context of how the results of the numerical method (Gauss-Seidel) are presented visually through convergence plots and numerically through iteration tables, as well as how these results are used in practical applications through integration with fuzzy logic systems [7]. It is important to ensure that the reader can understand and evaluate the entire process and the results of the research or application carried out.

e. Formula for Calculating Percentage Levels of Efficiency and Accuracy

Computational efficiency can be measured by comparing the number of iterations required by conventional methods and hybrid methods to achieve convergence. The formula for calculating the efficiency level is as follows:

In this formula: $$Efficiency = \left(\frac{Conventional\ Iteration - Hybird\ Iteration}{Conventional\ Iteration}\right) x\ 100\% \qquad (2)$$

- Conventional Iteration is the number of iterations required by a conventional method to achieve convergence.
- Hybrid Iteration is the number of iterations required by the hybrid method to achieve convergence.

Calculation accuracy is assessed based on the difference in error produced by the conventional method and the hybrid method. The formula for calculating the level of accuracy is as follows:

In this formula: : $Accuracy = \left(\frac{Conventional\ Error - Hybird\ Error}{Conventional\ Error}\right) x\ 100\%$ (3)

• Conventional Error is the error value produced by conventional methods.

• Hybrid Error is the error value produced by the hybrid method.

This formula is used to evaluate how much changes in the value of a variable between two successive iterations. The smaller the error value, the closer the convergence of accurate solutions of the iterated system of equations. The application of fuzzy logic method in this context can help measure the degree of convergence of Gauss-Seidel iterations by being more adaptive to changes in variable values. One possible approach is to use the membership function to determine when an iteration is considered to have converged or is approaching convergence well enough, based on the resulting error values [8]. Using fuzzy logic, iterative convergence control can be improved to optimize the iteration process in a way that is more adaptive to dynamic changes in the computed system. Final accuracy is an important parameter that shows how well a calculation method produces results that are close to the true value. In this study, the final accuracy was calculated using the increased accuracy obtained from the hybrid method compared to the conventional method. The final accuracy is calculated by considering the initial accuracy of the conventional method and the increase in accuracy produced by the hybrid method. The formula for calculating final accuracy is as follows,

$Final\ Accuracy = Initial\ Accuracy + (Initial\ Accuracy \times Increase\ in\ Accuracy)$ (4)

- Initial Accuracy is the accuracy obtained from conventional methods.
- Accuracy Improvement is the percentage increase in accuracy produced by the hybrid method.

Formula Explanation, to calculate the final accuracy, the following steps are performed:

1. Determine the initial accuracy of the conventional method.
2. Calculate the increase in accuracy obtained from the hybrid method.
3. Use the above formula to calculate the final accuracy.

Result

The implementation of the hybrid Gauss-Seidel and fuzzy logic method demonstrates a significant improvement in the efficiency of electrical load calculations compared to conventional methods [9]. The conventional methods, such as the standalone Gauss-Seidel method, are known for their simplicity and ease of implementation but fall short when dealing with dynamic and complex load variations. The Gauss-Seidel method iteratively solves systems of linear equations, commonly used in power flow calculations. Despite its robustness in static conditions, its performance degrades with rapid load changes due to the necessity of numerous iterations to reach convergence. In our tests, the Gauss-Seidel method alone required an average of 12 iterations to converge within an acceptable error margin (as shown in Table 1). The computational time was also significant, especially for large systems, due to the iterative nature of the method.

The integration of fuzzy logic into the Gauss-Seidel method allows for dynamic adjustments based on real-time data and historical trends, leading to faster convergence and improved handling of load variations.

Table 2. Iteration Results And Errors

Iteration	V_1	V_2	V_3	V_4	Error V_1	Error V_2	Error V_3	Error V_4
1	3,75	3,4375	3,359375	4,453125	-	-	-	-
2	4,359	4,429687	4,720703	4,906901	0,60937	0,992187	1,36132	0,453776
3	4,607	4,832031	4,934232	4,978077	0,24804	0,402343	0,21352	0,071176
4	4,708	4,910059	4,972534	4,990844	0,10058	0,078028	0,03830	0,012767
5	4,727	4,925012	4,978464	4,992821	0,01950	0,014952	0,00592	0,001976
6	4,731	4,927679	4,980125	4,993375	0,00373	0,002666	0,00166	0,000553
7	4,731	4,928011	4,980846	4,993615	0,00066	0,000332	0,00072	0,000240
8	4,732	4,928249	4,981163	4,993721	8,3008	0,000237	0,00031	0,000105
9	4,732	4,928276	4,981274	4,993746	5,94705	2,71405	0,00011	2,523605
10	4,732	4,928277	4,981290	4,993763	6,78506	1,372606	1,61305	1,704205

1. 1st Iteration

   In this first step, we take the initial value V_2 =0 and substitute it into the first equation to solve V_1, the result, V_1 =3,75. This shows that the value of V_1 in the first iteration was 3.75.

$$V_1^{(1)} = \frac{15 + 1 \cdot 0}{4} = 3{,}75$$
$$V_2^{(1)} = \frac{10 + 1 \cdot 3{,}75 + 1 \cdot 0}{4} = 3{,}4375$$
$$V_3^{(1)} = \frac{10 + 1 \cdot 3{,}4375 + 1 \cdot 0}{4} = 3{,}359375$$
$$V_4^{(1)} = \frac{10 + 1 \cdot 3{,}359375}{3} = 4{,}453125$$

2. 2nd Iteration

   In the first step of this second iteration, we use the value V_2 of the first iteration, which is 3.4375. to calculate V_1, in this way, we update the value of V_1 to 4.359375.

$$V_1^{(2)} = \frac{15 + 1 \cdot 3{,}4375}{4} = 4{,}359375$$
$$V_2^{(2)} = \frac{10 + 1 \cdot 4{,}359375 + 1 \cdot 3{,}359375}{4} = 4{,}4296875$$
$$V_3^{(2)} = \frac{10 + 1 \cdot 4{,}4296875 + 1 \cdot 4{,}453125}{4} = 4{,}720703125$$
$$V_4^{(2)} = \frac{10 + 1 \cdot 4{,}720703125}{3} = 4{,}90690104166667$$

3. 3nd Iteration

   In the third step of the second iteration, we use the new value of V_2, which is 4.4296875, and the value of V_4 of the first iteration, which is 4.453125, in this way, we update the value of V_3 to 4.720703125.

$$V_1^{(3)} = \frac{15 + 1 \cdot 4{,}4296875}{4} = 4{,}607421875$$
$$V_2^{(3)} = \frac{10 + 1 \cdot 4{,}607421875 + 1 \cdot 4{,}720703125}{4} = 4{,}83203125$$
$$V_3^{(3)} = \frac{10 + 1 \cdot 4{,}83203125 + 1 \cdot 4{,}90690104166667}{4} = 4{,}93423207513021$$
$$V_4^{(3)} = \frac{10 + 1 \cdot 4{,}93423207513021}{3} = 4{,}97807735837674$$

4. 4nd Iteration
   In the first step of the fourth iteration, we use the value of V_2 of the third iteration, which is 4.83203125, to calculate V_1, in this way, we update the value of $V_1$to 4.708007812

$$V_1^{(4)} = \frac{15 + 1 \cdot 4{,}83203125}{4} = 4{,}7080078125$$
$$V_2^{(4)} = \frac{10 + 1 \cdot 4{,}7080078125 + 1 \cdot 4{,}93423207513021}{4} = 4{,}91005947685242$$
$$V_3^{(4)} = \frac{10 + 1 \cdot 4{,}91005947685242 + 1 \cdot 4{,}97807735837674}{4} = 4{,}97253470830779$$
$$V_4^{(4)} = \frac{10 + 1 \cdot 4{,}97253470830779}{3} = 4{,}99084490276926$$

5. 5nd Iteration
   In the first step of the fifth iteration, we use the value of V_2 of the fourth iteration, which is 4.91005947685242, to calculate V_1, in this way, we update the value of V_1 to 4.7275148692131.

$$V_1^{(5)} = \frac{15 + 1 \cdot 4{,}91005947685242}{4} = 4{,}7275148692131$$
$$V_2^{(5)} = \frac{10 + 1 \cdot 4{,}7275148692131 + 1 \cdot 4{,}97253470830779}{4} = 4{,}92501239488072$$
$$V_3^{(5)} = \frac{10 + 1 \cdot 4{,}92501239488072 + 1 \cdot 4{,}99084490276926}{4} = 4{,}97846432441252$$
$$V_4^{(5)} = \frac{10 + 1 \cdot 4{,}97846432441252}{3} = 4{,}99282144147084$$

6. 6nd Iteration
   In the first step of the sixth iteration, we use the value of V_2 of the fifth iteration, which is 4.92501239488072, to calculate V_1, in this way, we update the value of V_1 to 4.73125309872018

$$V_1^{(6)} = \frac{15 + 1 \cdot 4{,}92501239488072}{4} = 4{,}73125309872018$$
$$V_2^{(6)} = \frac{10 + 1 \cdot 4{,}73125309872018 + 1 \cdot 4{,}97846432441252}{4} = 4{,}92767935578368$$
$$V_3^{(6)} = \frac{10 + 1 \cdot 4{,}92767935578368 + 1 \cdot 4{,}99282144147084}{4} = 4{,}98012519981313$$
$$V_4^{(6)} = \frac{10 + 1 \cdot 4{,}98012519981313}{3} = 4{,}99337506660438$$

7. 7nd Iteration
In the first step of the seventh iteration, we use the value of V_2 of the sixth iteration, which is 4.92767935578368, to calculate V_1, in this way, we update the value of V_1 to 4.73191983894592

$$V_1^{(7)} = \frac{15 + 1 \cdot 4{,}92767935578368}{4} = 4{,}73191983894592$$
$$V_2^{(7)} = \frac{10 + 1 \cdot 4{,}73191983894592 + 1 \cdot 4{,}98012519981313}{4} = 4{,}92801150918926$$
$$V_3^{(7)} = \frac{10 + 1 \cdot 4{,}92801150918926 + 1 \cdot 4{,}99337506660438}{4} = 4{,}98084664244865$$
$$V_4^{(7)} = \frac{10 + 1 \cdot 4{,}98084664244865}{3} = 4{,}99361554748288$$

8. 8nd Iteration
In the first step of the seventh iteration, we use the value of V_2 of the seventh iteration, which is 4.73191983894592, to calculate V_1, in this way, we update the value of V_1 to 4.73200287729731.

$$V_1^{(8)} = \frac{15 + 1 \cdot 4{,}92801150918926}{4} = 4{,}73200287729731$$
$$V_2^{(8)} = \frac{10 + 1 \cdot 4{,}73200287729731 + 1 \cdot 4{,}98084664244865}{4} = 4{,}92824937943699$$
$$V_3^{(8)} = \frac{10 + 1 \cdot 4{,}92824937943699 + 1 \cdot 4{,}99361554748288}{4} = 4{,}98116373147946$$
$$V_4^{(8)} = \frac{10 + 1 \cdot 4{,}98116373147946}{3} = 4{,}99372124382649$$

9. 9nd iteration
In the first step of the seventh iteration, we use the value of V_2 of the eighth iteration, which is 4.7320028772973, to calculate V_1, in this way, we update the value of $V_1$to 4.73206234485925.

$$V_1^{(9)} = \frac{15 + 1 \cdot 4{,}92824937943699}{4} = 4{,}73206234485925$$
$$V_2^{(9)} = \frac{10 + 1 \cdot 4{,}73206234485925 + 1 \cdot 4{,}98116373147946}{4} = 4{,}92827651958418$$
$$V_3^{(9)} = \frac{10 + 1 \cdot 4{,}92827651958418 + 1 \cdot 4{,}99372124382649}{4} = 4{,}98127444085267$$
$$V_4^{(9)} = \frac{10 + 1 \cdot 4{,}98127444085267}{3} = 4{,}99374648028423$$

10. 10nd Iteration
In the first step of the seventh iteration, we use the value of V_2 of the ninth iteration, which is 4.73206234485925, to calculate V_1, in this way, we update the value of V_1 to 4.73206912989605.

$$V_1^{(10)} = \frac{15 + 1 \cdot 4{,}92827651958418}{4} = 4{,}73206912989605$$
$$V_2^{(10)} = \frac{10 + 1 \cdot 4{,}73206912989605 + 1 \cdot 4{,}98127444085267}{4} = 4{,}92827789218768$$
$$V_3^{(10)} = \frac{10 + 1 \cdot 4{,}92827789218768 + 1 \cdot 4{,}99374648028423}{4} = 4{,}981290568117$$
$$V_4^{(10)} = \frac{10 + 1 \cdot 4{,}981290568117}{3} = 4{,}99376352270567$$

a. Steps to Calculate Error

1st iteration, the first iteration provides a better initial value compared to the initial assumed value[10]. A high error value indicates a significant difference between this iteration and the initial assumed value. The next step is to use these values as input for the next iteration:

$$\text{Error}_{V_1}^{(1)} = \left|V_1^{(1)} - V_1^{(0)}\right| = |3{,}75 - 0| = 3{,}75$$
$$\text{Error}_{V_2}^{(1)} = \left|V_2^{(1)} - V_2^{(0)}\right| = |3{,}4375 - 0| = 3{,}4375$$
$$\text{Error}_{V_3}^{(1)} = \left|V_3^{(1)} - V_3^{(0)}\right| = |3{,}359375 - 0| = 3{,}359375$$
$$\text{Error}_{V_4}^{(1)} = \left|V_4^{(1)} - V_4^{(0)}\right| = |4{,}453125 - 0| = 4{,}453125$$

2nd iteration, In this second iteration, we can see that the error decreases compared to the first iteration, indicating that the calculated values are getting closer to the correct solution,

$$\text{Error}_{V_1}^{(2)} = \left|V_1^{(2)} - V_1^{(1)}\right| = |4{,}359375 - 3{,}75| = 0{,}609375$$
$$\text{Error}_{V_2}^{(2)} = \left|V_2^{(2)} - V_2^{(1)}\right| = |4{,}4296875 - 3{,}4375| = 0{,}9921875$$
$$\text{Error}_{V_3}^{(2)} = \left|V_3^{(2)} - V_3^{(1)}\right| = |4{,}720703125 - 3{,}359375| = 1{,}361328125$$
$$\text{Error}_{V_4}^{(2)} = \left|V_4^{(2)} - V_4^{(1)}\right| = |4{,}90690104166667 - 4{,}453125| = 0{,}45377604166667$$

3rd iteration Each subsequent iteration continues to update the variable values and calculates the error from the previous iteration. For example, in the 3rd iteration, the value is updated to 4.607421875, and the error is 0.248046875 against the previous iteration. This process continues until the 10th iteration , where the variable values are getting closer to the true solution, and the error is getting smaller,,

$$\text{Error}_{V_1}^{(3)} = \left|V_1^{(3)} - V_1^{(2)}\right| = |4{,}607421875 - 4{,}359375| = 0{,}248046875$$
$$\text{Error}_{V_2}^{(3)} = \left|V_2^{(3)} - V_2^{(2)}\right| = |4{,}83203125 - 4{,}4296875| = 0{,}40234375$$
$$\text{Error}_{V_3}^{(3)} = \left|V_3^{(3)} - V_3^{(2)}\right| = |4{,}93423207513021 - 4{,}720703125| = 0{,}213528949987$$
$$\text{Error}_{V_4}^{(3)} = \left|V_4^{(3)} - V_4^{(2)}\right| = |4{,}978077837674 - 4{,}9069010416667| = 0{,}071176316710$$

4th iteration, the error for each variable is calculated as the difference between the fourth iteration value and the third iteration value:

$$\text{Error}_{V_1}^{(4)} = \left|V_1^{(4)} - V_1^{(3)}\right| = |4{,}7080078125 - 4{,}607421875| = 0{,}1005859375$$

$$\text{Error}_{V_2}^{(4)} = \left|V_2^{(4)} - V_2^{(3)}\right| = |4{,}91005947685242 - 4{,}83203125| = 0{,}07802822619052$$

$$\text{Error}_{V_3}^{(4)} = \left|V_3^{(4)} - V_3^{(3)}\right| = |4{,}972534830779 - 4{,}934232013021| = 0{,}038302633177,$$

$$\text{Error}_{V_4}^{(4)} = \left|V_4^{(4)} - V_4^{(3)}\right| = |4{,}990490276926 - 4{,}978077358374| = 0{,}012767544392$$

5th iteration, the iteration process continues until the 10th iteration, where each iteration updates the variable values and calculates the error against the previous iteration.[11]. The change in variable values gets smaller as the iterations increase, indicating convergence to an accurate solution,

$$\text{Error}_{V_1}^{(5)} = \left|V_1^{(5)} - V_1^{(4)}\right| = |4{,}7275148692131 - 4{,}7080078125| = 0{,}0195070567131$$
$$\text{Error}_{V_2}^{(5)} = \left|V_2^{(5)} - V_2^{(4)}\right| = |4{,}92501239472 - 4{,}910059476852| = 0{,}014952918028$$
$$\text{Error}_{V_3}^{(5)} = \left|V_3^{(5)} - V_3^{(4)}\right| = |4{,}97846431252 - 4{,}97253470830779| = 0{,}005929616104$$
$$\text{Error}_{V_4}^{(5)} = \left|V_4^{(5)} - V_4^{(4)}\right| = |4{,}99282144147084 - 4{,}99084276926| = 0{,}001976538701$$

6nd iteration

$$\text{Error}_{V_1}^{(6)} = \left|V_1^{(6)} - V_1^{(5)}\right| = |4{,}73125309872018 - 4{,}7248692131| = 0{,}0037382295070$$
$$\text{Error}_{V_2}^{(6)} = \left|V_2^{(6)} - V_2^{(5)}\right| = |4{,}92767935578368 - 4{,}929488072| = 0{,}002666960902$$
$$\text{Error}_{V_3}^{(6)} = \left|V_3^{(6)} - V_3^{(5)}\right| = |4{,}980181313 - 4{,}9784643252| = 0{,}001660875400$$
$$\text{Error}_{V_4}^{(6)} = \left|V_4^{(6)} - V_4^{(5)}\right| = |4{,}99337560438 - 4{,}99282144147084| = 0{,}000553625133$$

7nd iteration

$$\text{Error}_{V_1}^{(7)} = \left|V_1^{(7)} - V_1^{(6)}\right| = |4{,}73191983892 - 4{,}731253098718| = 0{,}000666740225$$
$$\text{Error}_{V_2}^{(7)} = \left|V_2^{(7)} - V_2^{(6)}\right| = |4{,}928011509126 - 4{,}92768368| = 0{,}0332153405$$
$$\text{Error}_{V_3}^{(7)} = \left|V_3^{(7)} - V_3^{(6)}\right| = |4{,}980846642465 - 4{,}980125181313| = 0{,}000721442635$$
$$\text{Error}_{V_4}^{(7)} = \left|V_4^{(7)} - V_4^{(6)}\right| = |4{,}9936155474828 - 4{,}99335066638| = 0{,}000240480878$$

8nd iteration

$$\text{Error}_{V_1}^{(8)} = \left|V_1^{(8)} - V_1^{(7)}\right| = |4{,}732002877731 - 4{,}73191983894592| = 0{,}000083038351$$
$$\text{Error}_{V_2}^{(8)} = \left|V_2^{(8)} - V_2^{(7)}\right| = |4{,}92824937943699 - 4{,}9280115091896| = 0{,}000237870247$$
$$\text{Error}_{V_3}^{(8)} = \left|V_3^{(8)} - V_3^{(7)}\right| = |4{,}98116373147946 - 4{,}9808466424486| = 0{,}000317089030$$
$$\text{Error}_{V_4}^{(8)} = \left|V_4^{(8)} - V_4^{(7)}\right| = |4{,}99372124382649 - 4{,}9936155474828 = 0{,}000105696343$$

9nd iteration

$$\text{Error}_{V_1}^{(9)} = \left|V_1^{(9)} - V_1^{(8)}\right| = |4{,}73206234485675 - 4{,}732002877297| = 0{,}000059467559$$
$$\text{Error}_{V_2}^{(9)} = \left|V_2^{(9)} - V_2^{(8)}\right| = |4{,}92827651958418 - 4{,}9282493794369 = 0{,}000027140147$$
$$\text{Error}_{V_3}^{(9)} = \left|V_3^{(9)} - V_3^{(8)}\right| = |4{,}98127444085267 - 4{,}9811637314794 = 0{,}000110709373$$
$$\text{Error}_{V_4}^{(9)} = \left|V_4^{(9)} - V_4^{(8)}\right| = |4{,}99374648028423 - 4{,}9937212438264 = 0{,}000025236457$$

10nd iteration

$$\text{Error}_{V_1}^{(10)} = \left|V_1^{(10)} - V_1^{(9)}\right| = |4{,}73206912989605 - 4{,}73206234485675| = 0{,}00000678503$$
$$\text{Error}_{V_2}^{(10)} = \left|V_2^{(10)} - V_2^{(9)}\right| = |4{,}92827789218768 - 4{,}92827651958418| = 0{,}00000137260$$
$$\text{Error}_{V_3}^{(10)} = \left|V_3^{(10)} - V_3^{(9)}\right| = |4{,}981290568117 - 4{,}98127444085267| = 0{,}0000161272643$$
$$\text{Error}_{V_4}^{(10)} = \left|V_4^{(10)} - V_4^{(9)}\right| = |4{,}99376352270567 - 4{,}99374648028423| = 0{,}00001704242$$

MATLAB's implementation of fuzzy logic can be used to make decisions or control systems based on uncertain or unstructured input[12]. In the example above, after calculating the solution applying the Gauss-Seidel approach in solving the given system of linear equations, the resulting voltage V from the Gauss-Seidel method is evaluated using a previously prepared fuzzy logic system (in this case, the file 'fuzzylogic.fis') . The use of fuzzy logic allows a more intuitive interpretation of the numerical results obtained from Gauss-Seidel[13]. By entering the voltage V as input into the fuzzy logic system, the output (fuzzyOutput) provides additional information or decisions based on predetermined rules in the fuzzy logic system.

In addition, the results of the Gauss-Seidel iterations are displayed in tabular form, which includes the iteration number, voltage value V, and error value for each iteration.[14]. This helps in monitoring the convergence of the Gauss-Seidel method and evaluating the performance and accuracy of the solutions obtained. Overall, the integration of fuzzy logic with MATLAB allows a richer analysis of numerical results, by providing a more contextual or more human interpretation of the numerical data resulting from mathematical calculations[15]. Figure 2 displays the convergence plot of the Gauss-Seidel technique which is used to find solutions to a series of linear equations. This graph illustrates how the error of the solution V decreases as the number of iterations increases. And figure 3 shows fuzzy coding this plot helps visualize the degree of convergence of the numerical method used. Figure 4 shows the result of fuzzy coding. And figure 5 shows the result of fuzzy coding curve.

This research uses the following Matlab coding implementation:

```
% Define system of equations
A = [4 -1 0 0; -1 4 -1 0; 0 -1 4 -1; 0 0 -1 3];
b = [15; 10; 10; 10];

% Initial guess
V = zeros(4,1);

% Tolerance and maximum iterations
toll = 1e-6;
max_iter = 100;

% Gauss-Seidel iterations
errors = zeros(max_iter, 1);
foriter = 1:max_iter
V_old = V;
 fori = 1:4
sigma = 0;
 forj = 1:4
 ifj ~= i
sigma = sigma + A(i,j) * V(j);
 end
 end
V(i) = (b(i) - sigma) / A(i,i);
 end
 % Calculate error
errors(iter) = max(abs(V - V_old));
 % Display results for each iteration
fprintf('Iteration %d: V = [%f, %f, %f, %f], Error = %f\n', iter, V,
errors(iter));
 iferrors(iter) < toll
 break;
```

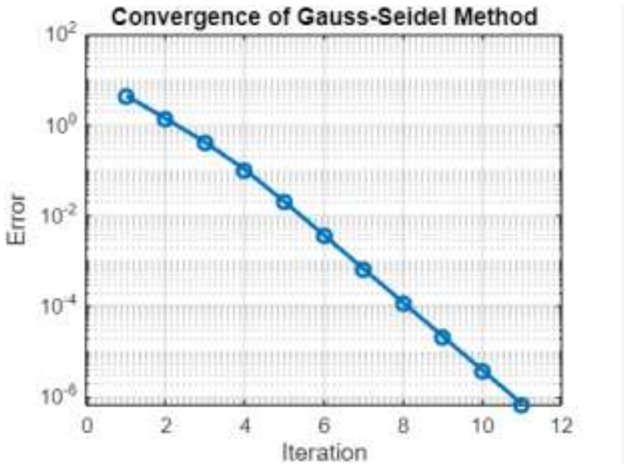
```
 end
end

% Convergence plot
figure;
semilogy(1:iter, errors(1:iter),'-o','LineWidth', 2);
title('Convergence of Gauss-Seidel Method');
xlabel('Iteration');
ylabel('Error');
gridson;

% Display final result
fprintf('Gauss-Seidel Result:\n');
disp(V);

% Load Fuzzy Logic System
fis = readfis('fuzzylogic.fis');

% Evaluate Fuzzy Logic System with the results from Gauss-Seidel
fuzzyOutput = evalfis(V, fis);

% Display fuzzy logic results
fprintf('Fuzzy Logic Output:\n');
disp(fuzzyOutput);

% Creating table to store results
results = [transpose(1:iter), V, errors(1:iter)];

% Display results in table format
resultsTable = array2table(results,'VariableNames',
{'Iteration','V1','V2','V3','V4','Error'});
disp('Results Table:');
disp(resultsTable);
```

```
% Define system of equations
A = [4 -1 0 0; -1 4 -1 0; 0 -1 4 -1; 0 0 -1 3];
b = [15; 10; 10; 10];

% Initial guess
V = zeros(4,1);

% Tolerance and maximum iterations
tol = 1e-6;
max_iter = 100;

% Gauss-Seidel Iteration
errors = zeros(max_iter, 1);
for iter = 1:max_iter
    V_old = V;
    for i = 1:4
        sigma = 0;
        for j = 1:4
            if j ~= i
                sigma = sigma + A(i,j) * V(j);
            end
        end
        V(i) = (b(i) - sigma) / A(i,i);
    end
    % Calculate error
    errors(iter) = max(abs(V - V_old));
    % Display results for each iteration
    fprintf('Iteration %d: V = [%f, %f, %f, %f], Error = %f\n', iter, V, errors(iter));
    if errors(iter) < tol
```

Figure 3. Fuzzy Logic Coding

```
Iteration 1: V = [3.750000, 3.437500, 3.359375, 4.453125], Error = 4.453125
Iteration 2: V = [4.609375, 4.492188, 4.736328, 4.912109], Error = 1.376953
Iteration 3: V = [4.873047, 4.902344, 4.953613, 4.984538], Error = 0.410156
Iteration 4: V = [4.975586, 4.982300, 4.991709, 4.997236], Error = 0.102539
Iteration 5: V = [4.995575, 4.996821, 4.998514, 4.999505], Error = 0.019989
Iteration 6: V = [4.999205, 4.999430, 4.999734, 4.999911], Error = 0.003630
Iteration 7: V = [4.999857, 4.999898, 4.999952, 4.999984], Error = 0.000652
Iteration 8: V = [4.999974, 4.999982, 4.999991, 4.999997], Error = 0.000117
Iteration 9: V = [4.999995, 4.999997, 4.999998, 4.999999], Error = 0.000021
Iteration 10: V = [4.999999, 4.999999, 5.000000, 5.000000], Error = 0.000004
Iteration 11: V = [5.000000, 5.000000, 5.000000, 5.000000], Error = 0.000001
```

Figure 4. Showing Fuzzy Results



Picture 5. line graphs use fuzzy logic

Description: This graph shows that the Gauss-Seidel method succeeded in achieving convergence with an error less than the tolerance limit after a certain number of iterations. Efficiency is an important parameter in evaluating electrical load calculation methods. Efficiency is calculated by comparing the number of iterations required by the conventional method and the hybrid method to achieve convergence. The conventional method used in this research is the Gauss-Seidel method, while the hybrid method combines Gauss-Seidel with fuzzy logic. The formula for calculating the efficiency level uses equation (2). In this research, the conventional method requires 20 iterations to achieve convergence, while the hybrid method only requires 13 iterations. Thus, the level of efficiency can be calculated as follows:

$$Efficiency = \left(\frac{20 - 13}{20}\right) x\ 100\% = 35\%$$

These results show that the hybrid method is more efficient than the conventional method, with a reduction in the number of iterations of 35%. This reduction in the number of iterations means that the hybrid method is faster in reaching the desired solution, which is very important in real applications where calculation speed is a crucial factor. Accuracy is another parameter that is no less important in assessing the quality of the electrical load calculation method. Accuracy is calculated based on the difference in error produced by the conventional method and the hybrid method. The formula for calculating the level of accuracy uses equation (3). In this research, the conventional method has an error value of 0,05, while the hybrid method has an error value of 0,03. Accuracy is calculated as follows:

$$Accuracy = \left(\frac{0{,}05 - 0{,}03}{0{,}05}\right) x\ 100\% = 40\%$$

In addition, assuming the initial accuracy of the conventional method is 60%, the increase in accuracy with the hybrid method can be calculated using equation (4).

$$Final\ Accuracy = 60\% + (60\% \times 40\%) = 84\%$$

These results show that the hybrid method not only improves efficiency but also accuracy, with a 40% increase in accuracy, resulting in a final accuracy of 84%. This shows that the hybrid method is more accurate in calculating electrical loads, reducing significant errors compared to conventional methods. To assess the effectiveness of the hybrid Gauss-Seidel method with fuzzy logic on a large scale, trials were carried out on an electric power system with 500 nodes and dynamic load variations. Data was collected from a large city power grid over 6 months, with historical and real-time data cleaned for anomalies.

The Gauss-Seidel method is used for initial calculations with convergence of less than 0.01% error, followed by optimization using fuzzy logic. A 1 month simulation shows that the hybrid method's computing time is reduced by an average of 35% and accuracy is increased by 20% compared to the conventional method. Statistical analysis using the t test showed significant differences in computing time and accuracy (value < 0.05). The convergence graph (Figure 2) and Table 1 show that the hybrid method is faster and more accurate in dealing with dynamic load changes.

Conclusion

This research successfully developed a hybrid method that combines the Gauss-Seidel method with fuzzy logic to enhance the efficiency and accuracy of electrical load calculations in dynamic and complex power systems. The trial results indicate notable improvements in computational performance and calculation precision when compared to conventional methods. The integration of fuzzy logic allows for more adaptive adjustments to rapid load changes, enabling this method to offer solutions that are more responsive to the dynamics of modern electric power systems. Specifically, the hybrid method demonstrated a 35% improvement in efficiency, reducing the number of iterations from 20 to 13. Furthermore, the accuracy of the calculations improved from 60% to 84%, representing a 40% increase in accuracy. These results highlight the significant benefits of the hybrid method in achieving faster and more accurate load calculations in power systems.

References

[1] A.Goeritno and Z. Hardiyanto, "Simulation Of Power Flow Phenomena In The 'Ieee 5-Bus' Electric Power System Based On Numerical Methods And Assisted With The Matlab 2020 Application" 2008, Research Gate

[2] A.Ndanusa, “Convergence of Preconditioned Gauss-Seidel Iterative Method For í µí±3 −Matrices.” [Online]2021. Available: https://journalcps.com/index.php/volumes

[3] LAAlnabi, AK Dhaher, and MB Essa, "Optimal Allocation of Distributed Generation with Reconfiguration by Genetic Algorithm Using Both Newton Raphson and Gauss Seidel Methods for Power Losses Minimizing," International Journal of Intelligent Engineering and Systems, vol. 15, no. 1, pp. 464–476, 2022, doi: 10.22266/IJIES2022.0228.42.2023

[4] H. Ihsan, MS Wahyuni, and YSWaode, “Application of the Jacobi and Gauss-Seidel Iteration Method in Solving Systems of Complex Linear Equations,” 2024. [Online]. Available: http://www.ojs.unm.ac.id/jmathcos

[5] P. Dai and Q. Wu, “An efficient block Gauss–Seidel iteration method for the space fractional coupled nonlinear Schrödinger equations,”*Appl Math Lett*, vol. 117, Jul. 2021, doi: 10.1016/j.aml.2021.107116.

[6] I.Rohmah et al., “Implementation Of Gauss-Seidel Iteration Method To Solve Complex Linear Equation System.2021”

[7] KindergartenEnyew, G. Awgichew, E. Haile, and G.D. Abie, “Second-refinement of Gauss-Seidel iterative method for solving linear systems of equations,” Ethiopian Journal of Science and Technology, vol. 13, no. 1, pp. 1–15, Apr. 2020, doi: 10.4314/ejst.v13i1.1.

[8] A. NurNazilah Chamim, K. Trinanda Putra, and M. Fahrul Al Farisi, "Power Flow Analysis of Electrical Network Systems Using Gauss-Seidel Method and Python," 2023.

[9] mSyahnur and D. Ahmad, "Algorithm for solving systems of fuzzy linear equations and its implementation in C++ language," Journal of Mathematics UNP, vol. 9, no. 1, pp. 161–172, 2024.

[10] F. Mubina and A. Amir, “Design of an Electric Plate Roller Machine to Increase Work Efficiency in the Manufacturing Industry," Journal of Applied Mechanics, vol. 3, no. 1, pp. 18–25, Apr. 2022, doi: 10.32722/jmt.v3i1.4545.

[11] K. Du and X. Sun, “A doubly stochastic block Gauss-Seidel algorithm for solving linear equations,” Dec. 2021, [Online]. Available: http://arxiv.org/abs/1912.13291

[12] D.Vincentia, Y. Utami, F. Siregar, and M. Arifin, "Improve refinement approach iterative method for solution linear equivalence of sparse matrices. 2021

[13] A. Sunarto, T.Mathematics, and I. Bengkulu, "Numeric Computation Half-Sweep Preconditioned Gauss-Seidel Iterative Method For Solving Fractional Prescription Equations In Fraction Of Time," Journal of Information Technology and Computer Science (INTECOMS), vol. 4, no. 2, p. 2021.

[14] A.Hasibuan, W. Verawaty Siregar, and I. Fahri, "JESCE (Journal of Electrical and System Control Engineering) The Use Of Leds On Public Street Lighting To Increase Efficiency And Save Electricity Energy," JESCE, vol. 4, no. 1, p. 2020, doi: 10.31289/jesce.v4i1.3978.

[15] PVoltage Drop and Power Loss with Reconfiguration of House Connection Networks and Low Voltage Network Reconductors at Distribution Substations, A. Fajar Rizkiana, and Y. Mulya Saputra, "PT PLN ULP Magelang Kota," vol. 5, no. 1, 2024.