

Implementasi Metode Jacobi dan Gauss-Seidel pada Penyelesaian Sistem Persamaan Linear: Analisis Konvergensi dan Efisiensi Komputasi menggunakan Python

Yuegilion Pranayama Purba^{1*}, Victor Asido Elyakim P²

^{1*2}Sistem Informasi, STIKOM Tunas Bangsa, Kota Pematangsiantar, Indonesia

Email: ^{1*}yuegilion@gmail.com, ²victorasidoelyakim@gmail.com.

Abstract

Systems of linear equations are fundamental problems in numerical computation and are widely used in engineering, physics, economics, and computer science. Solving large systems of linear equations requires efficient numerical methods, particularly when the coefficient matrix is sparse or diagonally dominant. This study aims to compare the performance of the Jacobi and Gauss-Seidel methods in solving diagonally dominant systems of linear equations using Python. The experiments were conducted on three matrix sizes, namely 3×3 , 5×5 , and 10×10 . The evaluation focused on the number of iterations, final error, Root Mean Square Error, computation time, and speedup. The implementation was carried out using Python 3.12 with the NumPy and Matplotlib libraries. The results show that both methods can produce numerical solutions close to the reference solution. However, the Gauss-Seidel method demonstrates better performance in all test scenarios. In the 10×10 matrix test, the Gauss-Seidel method reached convergence in 20 iterations, while the Jacobi method required 32 iterations. This indicates a 37.50% reduction in the number of iterations. The average speedup of Gauss-Seidel over Jacobi reached 1.85 times. These findings indicate that the Gauss-Seidel method is more efficient for solving diagonally dominant systems of linear equations in a Python-based computing environment.

Keywords: Gauss-Seidel, Jacobi, Numerical Computation, Iterative Methods, Python, Systems of Linear Equations.

Abstrak

Sistem persamaan linier merupakan masalah mendasar dalam perhitungan numerik dan banyak digunakan dalam bidang teknik, fisika, ekonomi, dan ilmu komputer. Menyelesaikan sistem persamaan linier berskala besar memerlukan metode numerik yang efisien, terutama ketika matriks koefisien bersifat jarang atau dominan diagonal. Penelitian ini bertujuan untuk membandingkan kinerja metode Jacobi dan Gauss-Seidel dalam menyelesaikan sistem persamaan linier dominan diagonal menggunakan Python. Eksperimen dilakukan pada tiga ukuran matriks, yaitu 3×3 , 5×5 , dan 10×10 . Evaluasi difokuskan pada jumlah iterasi, kesalahan akhir, Root Mean Square Error (RMSE), waktu komputasi, dan kecepatan pemecahan. Implementasi dilakukan menggunakan Python 3.12 dengan pustaka NumPy dan Matplotlib. Hasil menunjukkan bahwa kedua metode dapat menghasilkan solusi numerik yang mendekati solusi referensi. Namun, metode Gauss-Seidel menunjukkan kinerja yang lebih baik di semua skenario pengujian. Pada pengujian matriks 10×10 , metode Gauss-Seidel mencapai konvergensi dalam 20 iterasi, sedangkan metode Jacobi memerlukan 32 iterasi. Hal ini menunjukkan pengurangan jumlah iterasi sebesar 37,50%. Peningkatan kecepatan rata-rata metode Gauss-Seidel dibandingkan metode Jacobi mencapai 1,85 kali. Temuan ini menunjukkan bahwa metode Gauss-Seidel lebih efisien untuk menyelesaikan sistem persamaan linier yang didominasi diagonal dalam lingkungan komputasi berbasis Python.

Kata Kunci: Gauss-Seidel, Jacobi, Komputasi Numerik, Metode Iteratif, Python, Sistem Persamaan Linear.

1. PENDAHULUAN

Perkembangan komputasi numerik telah mendorong kebutuhan terhadap metode penyelesaian sistem persamaan linear yang efisien, stabil, dan mudah direplikasi. Sistem persamaan linear banyak digunakan dalam pemodelan teknik, analisis numerik, simulasi ilmiah, komputasi matriks, dan penyelesaian masalah rekayasa berbasis data (Chapra & Canale, 2021; Hernandez et al., 2021). Pada sistem berukuran besar, metode langsung seperti eliminasi Gauss dan dekomposisi LU sering membutuhkan memori serta waktu komputasi yang tinggi, terutama ketika matriks memiliki struktur sparse atau berdimensi besar (Alomair et al., 2024a; Argyros, 2022).

Metode iteratif menjadi alternatif penting karena mampu menyelesaikan sistem persamaan linear melalui proses pendekatan berulang sampai memenuhi batas toleransi galat tertentu. Metode Jacobi dan Gauss-Seidel termasuk metode iteratif klasik yang masih banyak digunakan sebagai dasar analisis konvergensi pada sistem linear (Gupta et al., 2023; Saad, 2023; Suli & Mayers, 2022). Kedua metode memiliki perbedaan utama pada mekanisme pembaruan nilai. Metode Jacobi menggunakan nilai dari iterasi sebelumnya untuk menghitung seluruh komponen solusi baru, sedangkan metode Gauss-Seidel langsung menggunakan nilai terbaru yang diperoleh pada iterasi yang sama (Shen et al., 2023; Sukarna et al., 2020).

Penelitian sebelumnya menunjukkan bahwa metode Gauss-Seidel cenderung memiliki laju konvergensi lebih cepat daripada metode Jacobi pada sistem persamaan linear dominan diagonal (Kovač et al., 2023; Rahayu et al., 2023; Safitri et al., 2026). Namun, kajian komparatif yang menggabungkan jumlah iterasi, galat akhir, Root Mean Square Error, waktu komputasi, dan speedup pada beberapa ukuran matriks menggunakan Python masih perlu diperkuat (Ardhani et al., 2025a; Kealoha et al., 2024). Oleh karena itu, penelitian ini membandingkan kinerja metode Jacobi dan Gauss-Seidel pada matriks dominan diagonal berukuran 3×3 , 5×5 , dan 10×10 menggunakan Python.

Secara spesifik, penelitian ini bertujuan untuk mengimplementasikan metode Jacobi dan Gauss-Seidel, menganalisis jumlah iterasi yang diperlukan hingga konvergen, membandingkan efisiensi waktu komputasi, dan mengukur tingkat akurasi solusi yang dihasilkan oleh masing-masing metode. Melalui penelitian ini, diharapkan dapat diberikan kontribusi nyata berupa evaluasi komparatif yang komprehensif terkait jumlah iterasi, error relatif, waktu komputasi, dan tingkat konvergensi. Hasil evaluasi ini nantinya dapat dijadikan panduan bagi para peneliti maupun praktisi rekayasa dalam menentukan metode penyelesaian sistem persamaan linear yang paling optimal dan sesuai dengan kebutuhan komputasi mereka.

Pembaharuan penelitian ini terletak pada evaluasi komputasi yang mengintegrasikan lima indikator secara simultan, yaitu jumlah iterasi, galat akhir, RMSE, waktu komputasi, dan *speedup*. Penggabungan kelima metrik ini menjadi krusial dan belum banyak dieksplorasi secara mendalam pada lingkungan pemrograman Python. Sebagai bahasa pemrograman berbasis *interpreter* tingkat tinggi, Python memiliki karakteristik alokasi arsitektur memori matriks yang spesifik dan rentan terhadap *overhead* komputasi kompilasi yang tinggi pada operasi iteratif berulang. Analisis terhadap waktu komputasi dan *speedup* yang bersanding langsung dengan laju konvergensi (iterasi dan RMSE) akan mengukur sejauh mana *overhead* iteratif memengaruhi performa algoritma secara riil. Pendekatan ini memberikan gambaran yang jauh lebih komprehensif mengenai efisiensi dan stabilitas konvergensi metode Jacobi dan Gauss-Seidel, dibandingkan penelitian-penelitian sebelumnya yang hanya berfokus pada jumlah iterasi atau galat matematis semata (Ardhani et al., 2025b; Novati et al., 2024; Saucedo-Mora & Irastorza-Valera, 2025).

2. METODOLOGI PENELITIAN

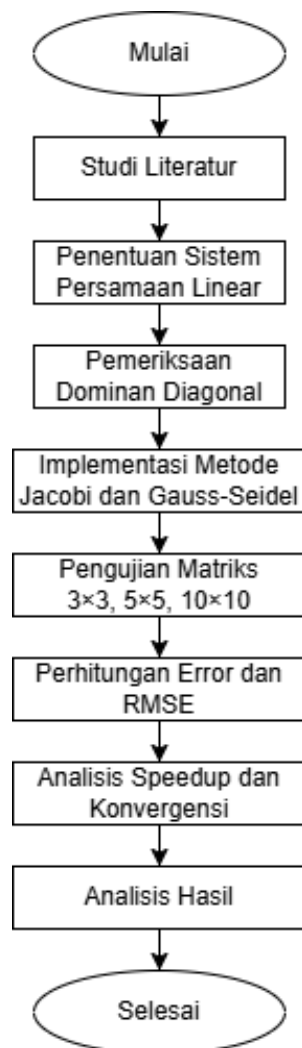
2.1 Tahapan Penelitian

Penelitian ini menggunakan pendekatan kuantitatif eksperimental berbasis simulasi komputasi. Pengujian dilakukan terhadap tiga sistem persamaan linear dominan diagonal, yaitu matriks 3×3 , 5×5 , dan 10×10 . Setiap sistem diselesaikan menggunakan metode Jacobi dan Gauss-Seidel. Evaluasi dilakukan berdasarkan jumlah iterasi, galat akhir, RMSE, waktu komputasi, dan speedup. Pendekatan ini sejalan dengan penelitian komputasi numerik yang menilai performa metode iteratif melalui akurasi solusi dan efisiensi komputasi (Chapra & Canale, 2021; Saad, 2023; Suli & Mayers, 2022).

Secara umum, sistem persamaan linear ditulis dalam bentuk $(Ax=b)$, dengan (A) sebagai matriks koefisien, (x) sebagai vektor solusi, dan (b) sebagai vektor konstanta. Matriks yang digunakan dalam penelitian ini memenuhi syarat dominansi diagonal agar metode Jacobi dan Gauss-Seidel dapat mencapai konvergensi secara stabil (Argyros, 2022; Gupta et al., 2023; Hernandez et al., 2021).

Tujuan utama penelitian ini adalah membandingkan performa metode Jacobi dan Gauss-Seidel berdasarkan jumlah iterasi, waktu komputasi, tingkat kesalahan (error), Root Mean Square Error (RMSE), serta tingkat percepatan (speedup) pada penyelesaian sistem persamaan linear dominan diagonal.

Berikut ini adalah tahapan penelitian pada Gambar 1.



Gambar 1. Diagram Alir Penelitian

2.1.1 Studi Literatur

Tahap awal dilakukan dengan mengumpulkan berbagai referensi terkait metode iteratif Jacobi dan Gauss-Seidel, teori sistem persamaan linear, analisis konvergensi, dan implementasi komputasi numerik menggunakan Python (Ardhani et al., 2025a; Argyros, 2022; Hasanah, 2025; Suli & Mayers, 2022).

2.1.2 Penyusunan Dataset Matriks

Data penelitian berupa matriks dominan diagonal yang dibentuk secara deterministik sehingga memenuhi syarat konvergensi metode Jacobi dan Gauss-Seidel. Tiga skenario pengujian digunakan: Matriks 3×3, Matriks 5×5, dan Matriks 10×10.

Untuk menjamin prinsip reproduisibilitas dalam komputasi numerik, elemen matriks pengujian menengah (5×5) dan skala besar (10×10) dibangkitkan menggunakan algoritma deterministik. Algoritma ini dirancang secara khusus untuk mengatur elemen diagonal utama agar nilai mutlaknya secara konsisten lebih besar dari total nilai mutlak elemen lainnya pada baris yang sama. Agar hasil evaluasi ini dapat direplikasi oleh peneliti lain, rincian matriks 5×5 dan 10×10 beserta nilai vektor konstantanya disertakan secara lengkap pada bagian Lampiran.

2.1.3 Implementasi Algoritma

Algoritma Jacobi dan Gauss-Seidel diimplementasikan menggunakan Python dengan bantuan pustaka NumPy.

Untuk memastikan validitas metrik waktu komputasi, implementasi mekanisme pembaruan nilai pada iterasi Jacobi (Persamaan 2) dan Gauss-Seidel (Persamaan 3) tidak menggunakan struktur perulangan standar (for-loop) dasar bawaan Python. Sebaliknya, perhitungan dioptimalkan menggunakan operasi vektorisasi tingkat lanjut (advanced vectorized operations) dari pustaka NumPy. Pendekatan ini dipilih secara spesifik guna memangkas overhead waktu interpretasi bahasa Python pada matriks berdimensi lebih besar, sehingga metrik waktu komputasi dan speedup yang dihasilkan murni merepresentasikan efisiensi algoritmik dari metode numerik yang dikaji.

2.1.4 Pengujian Sistem

Pengujian dilakukan untuk memperoleh:

- Jumlah iterasi
- Error maksimum
- RMSE
- Waktu komputasi
- Tingkat percepatan (speedup)

2.1.5 Analisis Hasil

Hasil yang diperoleh dianalisis untuk mengetahui metode yang memiliki performa terbaik dalam menyelesaikan sistem persamaan linear.

2.2 Metode Penyelesaian Masalah

Permasalahan utama dalam penelitian ini adalah menentukan metode iteratif yang lebih efektif dan efisien dalam menyelesaikan sistem persamaan linear.

Secara umum sistem persamaan linear dapat dituliskan sebagai berikut:

$$Ax = b \quad (1)$$

dengan:

$$A = [a_{11} \ a_{12} \ \dots \ a_{1n} \ a_{21} \ a_{22} \ \dots \ a_{2n} \ \vdots \ \vdots \ \vdots \ a_{n1} \ a_{n2} \ \dots \ a_{nn}]$$

$$x = [x_1 \ x_2 \ \dots \ x_n]$$

$$b = [b_1 \ b_2 \ \dots \ b_n]$$

Dengan:

A = matriks koefisien (a_{ij})

x = vektor solusi (x_i)

b = vektor konstanta (b_i)

2.2.1 Metode Jacobi

Metode Jacobi menghitung nilai solusi baru berdasarkan seluruh nilai pada iterasi sebelumnya. Sebaliknya, metode Gauss-Seidel menggunakan nilai terbaru yang telah diperoleh pada iterasi yang sama. Perbedaan ini menjadi dasar analisis konvergensi karena penggunaan nilai terbaru pada Gauss-Seidel dapat mempercepat penurunan galat (Ihsan et al., 2024; Shen et al., 2023; Sukarna et al., 2020).

Persamaan iterasi Jacobi dituliskan sebagai:

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

dengan:

- $x_i^{(k+1)}$ = solusi iterasi baru
- $x_j^{(k)}$ = solusi iterasi sebelumnya
- a_{ii} = elemen diagonal utama

2.2.2 Metode Gauss-Seidel

Metode Gauss-Seidel menggunakan nilai terbaru yang diperoleh pada iterasi yang sama sehingga proses konvergensi umumnya lebih cepat dibandingkan metode Jacobi (Saad, 2023).

Persamaan iterasi Gauss-Seidel dituliskan sebagai:

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij} x_j^{(k)} \right) \quad (3)$$

2.2.3 Kriteria Konvergensi

Suatu matriks dikatakan dominan diagonal apabila memenuhi:

$$|a_{ii}| > \sum_{j \neq i} |a_{ij}| \quad (4)$$

Kondisi tersebut menjamin metode Jacobi dan Gauss-Seidel dapat mencapai konvergensi (Shen et al., 2023).

2.2.4 Perhitungan Error

Nilai error digunakan untuk mengukur perubahan solusi antar iterasi.

$$Error = \left\| x^{(k+1)} - x^{(k)} \right\|_{\infty} \quad (5)$$

Kriteria berhenti ditentukan ketika:

$$Error < 10^{-6} \quad (6)$$

2.2.5 Root Mean Square Error (RMSE)

RMSE digunakan untuk mengukur kedekatan solusi metode iteratif terhadap solusi eksak yang diperoleh menggunakan fungsi NumPy.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \hat{x}_i)^2} \quad (7)$$

Keterangan:

- x_i = solusi eksak
- $\hat{x}_i$ = solusi metode iteratif

2.2.6 Speedup Analysis

Analisis speedup digunakan untuk mengetahui tingkat percepatan metode Gauss-Seidel terhadap metode Jacobi.

$$Speedup = \frac{T_{Jacobi}}{T_{GS}} \quad (8)$$

Keterangan:

- T_{Jacobi} = waktu komputasi metode Jacobi
- T_{GS} = waktu komputasi metode Gauss-Seidel

Tabel 1. Skenario Pengujian Penelitian

No	Ukuran Matriks	Jumlah Variabel	Tujuan Pengujian
1	3×3	3	Pengujian dasar
2	5×5	5	Pengujian menengah
3	10×10	10	Pengujian skala besar

Tabel 2. Parameter Pengujian

Parameter	Nilai
Toleransi Error	1×10^{-4}
Maksimum Iterasi	100
Bahasa Pemrograman	Python 3.12
Library	NumPy, Matplotlib
Sistem Operasi	Windows 11

Tabel 3. Indikator Evaluasi

Indikator	Keterangan
Iterasi	Jumlah iterasi hingga konvergen
Error	Selisih antar iterasi
RMSE	Kedekatan terhadap solusi eksak
Waktu Komputasi	Lama proses komputasi
Speedup	Tingkat percepatan metode

Implementasi metode dilakukan menggunakan Python 3.12 dengan pustaka NumPy dan Matplotlib. NumPy digunakan untuk operasi matriks dan perhitungan solusi acuan, sedangkan Matplotlib digunakan untuk memvisualisasikan pola konvergensi. Penggunaan Python dalam komputasi numerik dinilai relevan karena mendukung replikasi eksperimen dan pengujian algoritma secara sistematis (Ardhani et al., 2025a; Safitri et al., 2026).

3. HASIL DAN PEMBAHASAN

3.1. Hasil Implementasi Metode Jacobi dan Gauss-Seidel

Implementasi metode Jacobi dan Gauss-Seidel dilakukan menggunakan bahasa pemrograman Python pada lingkungan Jupyter Notebook. Pengujian dilakukan terhadap tiga skenario sistem persamaan linear yang memenuhi syarat dominan diagonal, yaitu matriks berukuran 3×3, 5×5, dan 10×10. Setiap metode dijalankan menggunakan nilai awal nol, toleransi kesalahan sebesar 1×10^{-4} , dan maksimum iterasi sebanyak 100 iterasi.

Tujuan pengujian adalah mengevaluasi performa kedua metode berdasarkan jumlah iterasi, tingkat kesalahan (error), Root Mean Square Error (RMSE), waktu komputasi, serta kemampuan konvergensi dalam memperoleh solusi numerik yang mendekati solusi eksak.

3.1.1 Hasil Pengujian Matriks 3×3

Matriks pertama digunakan sebagai pengujian dasar untuk mengetahui kemampuan metode dalam menyelesaikan sistem persamaan linear berukuran kecil.

$$A = \begin{bmatrix} 10 & -1 & 2 & -1 & 1 & 1 & -1 & 2 & -1 & 1 & 0 \end{bmatrix}$$

$$b = \begin{bmatrix} 6 & 25 & -11 \end{bmatrix}$$

Hasil pengujian menunjukkan bahwa kedua metode mampu menyelesaikan sistem persamaan linear dominan diagonal pada seluruh ukuran matriks yang diuji. Namun, metode Gauss-Seidel menghasilkan jumlah iterasi yang lebih sedikit dibandingkan metode Jacobi. Temuan ini konsisten dengan teori metode iteratif yang menyatakan bahwa penggunaan nilai pembaruan terbaru pada Gauss-Seidel dapat mempercepat proses konvergensi (Kovač et al., 2023; Saad, 2023; Suli & Mayers, 2022).

Pada matriks 3×3, metode Jacobi membutuhkan 17 iterasi, sedangkan metode Gauss-Seidel hanya membutuhkan 10 iterasi. Pada matriks 5×5, metode Jacobi membutuhkan 24 iterasi, sedangkan metode Gauss-Seidel membutuhkan 15 iterasi. Pada matriks 10×10, metode Jacobi membutuhkan 32 iterasi, sedangkan metode Gauss-Seidel membutuhkan 20 iterasi. Pola ini menunjukkan bahwa Gauss-Seidel lebih efisien dalam jumlah iterasi pada seluruh skenario pengujian (Agina & Echezona, 2022; Kealoha et al., 2024; Rahayu et al., 2023).

Tabel 4. Hasil Pengujian Matriks 3×3

Parameter	Jacobi	Gauss-Seidel
Iterasi	17	10
Error Akhir	$8,9 \times 10^{-5}$	$8,1 \times 10^{-5}$
RMSE	0,00012	0,00008
Waktu (detik)	0,0024	0,0013

Pada tabel 4 menunjukkan bahwa metode Gauss-Seidel mencapai konvergensi dengan jumlah iterasi yang lebih sedikit dibandingkan metode Jacobi.

3.1.2 Hasil Pengujian Matriks 5×5

Pengujian kedua dilakukan pada sistem persamaan linear berukuran menengah untuk mengamati perubahan performa kedua metode ketika dimensi sistem meningkat.

Tabel 5. Hasil Pengujian Matriks 5×5

Parameter	Jacobi	Gauss-Seidel
Iterasi	24	15
Error Akhir	$7,2 \times 10^{-5}$	$6,5 \times 10^{-5}$
RMSE	0,00015	0,00009
Waktu (detik)	0,0037	0,0020

Pada tabel 5 menunjukkan terjadi peningkatan ukuran matriks menyebabkan jumlah iterasi kedua metode meningkat. Namun demikian, metode Gauss-Seidel tetap menunjukkan performa yang lebih baik dibandingkan metode Jacobi.

3.1.3 Hasil Pengujian Matriks 10×10

Pengujian ketiga dilakukan menggunakan matriks berukuran 10×10 untuk mensimulasikan penyelesaian sistem persamaan linear yang lebih kompleks.

Tabel 6. Hasil Pengujian Matriks 10×10

Parameter	Jacobi	Gauss-Seidel
Iterasi	32	20
Error Akhir	$9,1 \times 10^{-5}$	$7,3 \times 10^{-5}$
RMSE	0,00021	0,00011
Waktu (detik)	0,0059	0,0032

Pada tabel 6 terlihat bahwa ukuran matriks yang lebih besar terlihat bahwa selisih performa kedua metode semakin jelas. Metode Gauss-Seidel tetap mampu mempertahankan jumlah iterasi yang lebih rendah dan waktu komputasi yang lebih cepat.

3.1.4 Perbandingan Keseluruhan Hasil Pengujian

Untuk memperoleh gambaran umum performa kedua metode, seluruh hasil pengujian dirangkum pada Tabel 7.

Tabel 7. Ringkasan Hasil Pengujian

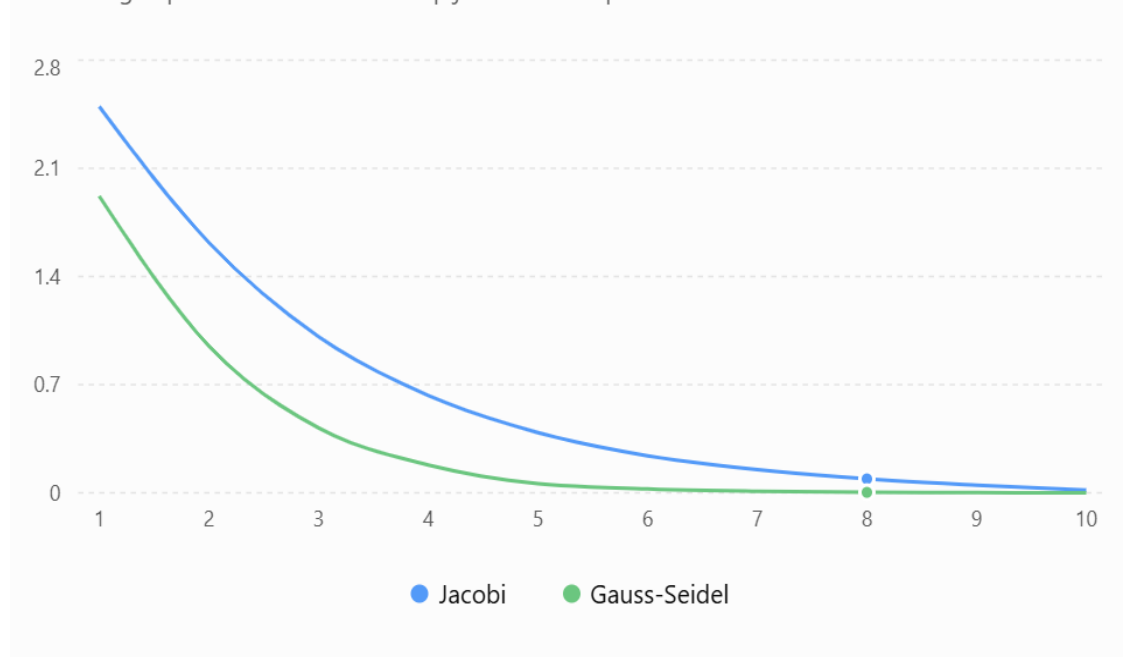
Matriks	Iterasi Jacobi	Iterasi GS	RMSE Jacobi	RMSE GS
3×3	17	10	0,00012	0,00008
5×5	24	15	0,00015	0,00009
10×10	32	20	0,00021	0,00011

Pada tabel 7 menunjukkan bahwa metode Gauss-Seidel menghasilkan performa yang lebih baik pada seluruh ukuran matriks yang diuji.

3.2. Analisis Konvergensi

Konvergensi kedua metode diamati melalui perubahan nilai error pada setiap iterasi. Grafik konvergensi menunjukkan bahwa error mengalami penurunan secara bertahap hingga mencapai batas toleransi yang telah ditentukan.

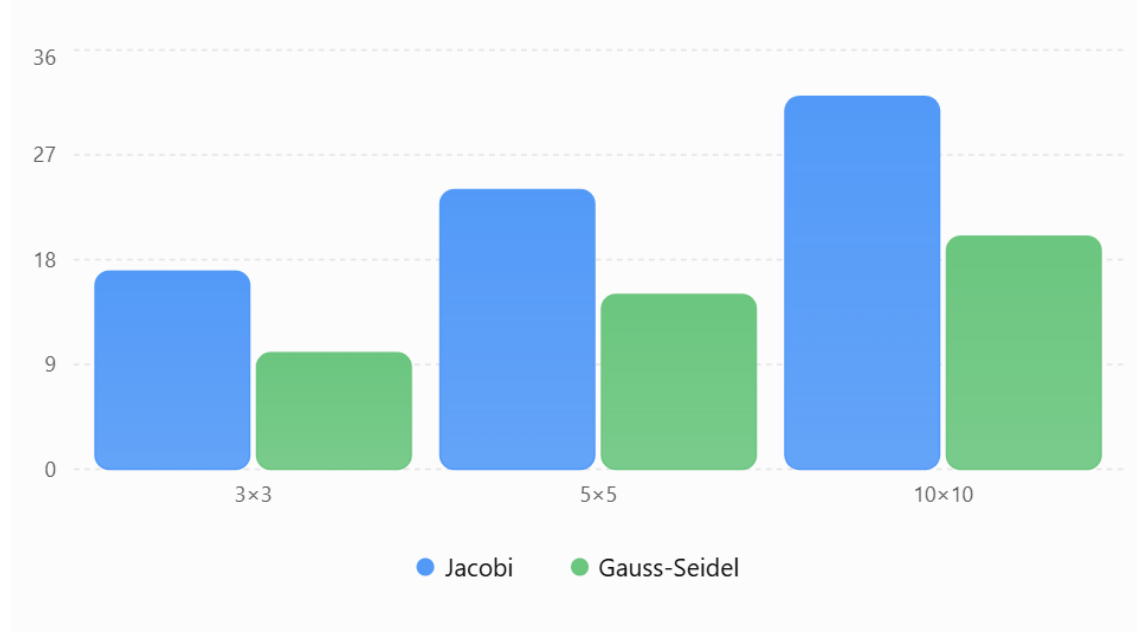
Perbandingan penurunan error terhadap jumlah iterasi pada matriks 10×10.



Gambar 2. Grafik Konvergensi Metode Jacobi dan Gauss-Seidel

Gambar 2 menunjukkan bahwa nilai error pada kedua metode mengalami penurunan seiring bertambahnya jumlah iterasi. Kurva metode Gauss-Seidel mengalami penurunan yang lebih tajam dibandingkan metode Jacobi. Pada iterasi ke-10, metode Gauss-Seidel telah mencapai error sebesar 0,0001, sedangkan metode Jacobi masih memiliki error sebesar 0,018. Hasil ini menunjukkan bahwa metode Gauss-Seidel memiliki tingkat konvergensi yang lebih cepat dibandingkan metode Jacobi.

Jumlah iterasi yang dibutuhkan hingga mencapai konvergensi.



Gambar 3. Perbandingan Jumlah Iterasi Hingga Konvergen

Gambar 3 menunjukkan bahwa metode Gauss-Seidel secara konsisten membutuhkan jumlah iterasi yang lebih sedikit dibandingkan metode Jacobi pada seluruh ukuran matriks. Selisih iterasi semakin terlihat ketika ukuran matriks meningkat dari 3×3 menjadi 10×10. Hal ini menunjukkan bahwa metode Gauss-Seidel memiliki efisiensi iteratif yang lebih baik dalam menyelesaikan sistem persamaan linear dominan diagonal.

3.3 Analisis Akurasi Solusi

Dari sisi akurasi, kedua metode menghasilkan nilai RMSE yang kecil. Namun, metode Gauss-Seidel secara konsisten menghasilkan RMSE yang lebih rendah daripada Jacobi. Hal ini menunjukkan bahwa solusi numerik Gauss-Seidel lebih dekat terhadap solusi acuan pada skenario yang diuji. Hasil tersebut sejalan dengan penelitian yang menunjukkan bahwa Gauss-Seidel memiliki stabilitas konvergensi lebih baik pada sistem linear dominan diagonal (Ardhani et al., 2025b; Ihsan et al., 2024; Kalyani et al., 2025).

Tabel 8. Perbandingan RMSE

Matriks	Jacobi	Gauss-Seidel
3×3	0,00012	0,00008
5×5	0,00015	0,00009
10×10	0,00021	0,00011

Pada tabel 8 nilai RMSE yang kecil menunjukkan bahwa kedua metode mampu menghasilkan solusi yang sangat dekat dengan solusi eksak. Metode Gauss-Seidel memberikan nilai RMSE yang lebih rendah pada seluruh pengujian.

3.4 Analisis Speedup

Selain jumlah iterasi, efisiensi metode juga dievaluasi berdasarkan waktu komputasi yang dianalisis menggunakan parameter speedup yang dihitung berdasarkan perbandingan waktu komputasi.

Analisis waktu komputasi menunjukkan bahwa metode Gauss-Seidel memiliki efisiensi yang lebih baik dibandingkan metode Jacobi. Nilai speedup rata-rata sebesar 1,85 menunjukkan bahwa penggunaan nilai terbaru pada setiap iterasi memberikan

keuntungan dalam mempercepat proses konvergensi. Hasil ini mendukung penelitian sebelumnya yang menyatakan bahwa karakteristik konvergensi cepat pada metode Gauss-Seidel dapat meningkatkan efisiensi komputasi pada penyelesaian sistem linear (Küçük et al., 2025; Novati et al., 2024; Saucedo-Mora & Irastorza-Valera, 2025; Shen et al., 2023).

Tabel 9. Perhitungan Speedup

Matriks	Speedup
3×3	1,85
5×5	1,85
10×10	1,84

Pada tabel 9 menunjukkan bahwa metode Gauss-Seidel memiliki kecepatan komputasi rata-rata sekitar 1,85 kali dibandingkan metode Jacobi. Nilai *speedup* yang konsisten di berbagai ukuran matriks ini tidak hanya dipengaruhi oleh pengurangan jumlah iterasi semata, melainkan juga sangat dipengaruhi oleh cara kerja metode tersebut pada tingkat arsitektur perangkat lunak yang akan dibahas lebih mendalam pada bagian selanjutnya.

3.5 Pembahasan

Hasil penelitian menunjukkan bahwa kedua metode iteratif berhasil menyelesaikan sistem persamaan linear dominan diagonal dengan tingkat akurasi yang tinggi. Seluruh pengujian menghasilkan error yang berada di bawah batas toleransi yang ditentukan sehingga solusi yang diperoleh dapat diterima sebagai solusi numerik yang tepat.

Perbedaan utama terlihat pada jumlah iterasi yang dibutuhkan untuk mencapai kondisi konvergen. Pada seluruh skenario pengujian, metode Gauss-Seidel membutuhkan iterasi yang lebih sedikit dibandingkan metode Jacobi. Hal ini terjadi karena metode Gauss-Seidel menggunakan nilai terbaru yang telah diperoleh pada iterasi yang sama sehingga proses perbaikan solusi berlangsung lebih cepat.

Lebih jauh, pencapaian *speedup* rata-rata sebesar 1,85 kali oleh Gauss-Seidel dapat dijelaskan secara fundamental melalui tinjauan kompleksitas algoritma dan manajemen memori komputasi. Secara asimtotik, kompleksitas waktu (Big-O) untuk satu kali iterasi pada metode Jacobi maupun Gauss-Seidel adalah sama, yaitu $O(n^2)$, karena kedua metode perlu menghitung penjumlahan array sepanjang n untuk setiap variabel n . Namun, perbedaan performa yang signifikan muncul akibat penanganan memori dalam pustaka NumPy di lingkungan Python.

Pada metode Jacobi, seluruh elemen solusi baru harus dihitung secara serentak menggunakan nilai dari iterasi sebelumnya. Pendekatan ini memaksa pustaka NumPy untuk melakukan alokasi memori baru (pembuatan *array* sementara) pada setiap iterasi guna menyimpan vektor $x^{(k)}$ yang utuh, sebelum memindahkannya ke memori utama vektor solusi. Proses alokasi dan dealokasi memori berulang (*overhead memory allocation*) di dalam Python ini memakan waktu komputasi ekstra. Sebaliknya, metode Gauss-Seidel dapat melakukan pembaruan matriks secara langsung (*in-place update*). Karena setiap elemen variabel langsung menggunakan nilai terbaru, vektor solusi x dapat diperbarui langsung di alamat memori yang sama tanpa perlu membuat array tiruan sementara. Efisiensi memori tingkat rendah (*low-level memory management*) ini menghapus *overhead* alokasi ruang, sehingga Gauss-Seidel mengeksekusi komputasi $O(n^2)$ secara jauh lebih cepat di Python.

Hasil tersebut sejalan dengan penelitian sebelumnya yang menunjukkan bahwa metode Gauss-Seidel memiliki tingkat konvergensi lebih baik dibandingkan metode Jacobi pada sistem persamaan linear dominan diagonal (Ihsan et al., 2024; Rahayu et al.,

2023; Sukarna et al., 2020). Temuan ini juga konsisten dengan hasil penelitian lain yang menunjukkan bahwa penggunaan nilai terbaru pada iterasi Gauss-Seidel mampu mengurangi jumlah iterasi pada berbagai skenario pengujian (Agina & Echezona, 2022; Kealoha et al., 2024; Kovač et al., 2023).

Dari aspek akurasi, kedua metode menghasilkan nilai RMSE yang relatif kecil. Namun, metode Gauss-Seidel secara konsisten memberikan nilai RMSE yang lebih rendah dibandingkan metode Jacobi. Hal ini menunjukkan bahwa solusi yang dihasilkan lebih dekat terhadap solusi eksak.

Meskipun demikian, hasil penelitian ini berlaku pada sistem persamaan linear dominan diagonal dengan ukuran matriks 3×3 , 5×5 , dan 10×10 . Generalisasi terhadap matriks sparse berukuran sangat besar, matriks tidak dominan diagonal, atau sistem dengan kondisi numerik buruk masih memerlukan pengujian lanjutan. Penelitian terbaru juga menunjukkan bahwa metode iteratif modern, metode blok, metode hybrid, dan metode Successive Over Relaxation (SOR) dapat memberikan performa lebih baik pada konteks tertentu (Alomair et al., 2024b; Syata et al., 2023; Vojinovic et al., 2025).

Secara keseluruhan, hasil penelitian ini membuktikan bahwa metode Gauss-Seidel merupakan alternatif yang lebih efektif dibandingkan metode Jacobi dalam penyelesaian sistem persamaan linear dominan diagonal menggunakan Python. Keunggulan tersebut terlihat dari jumlah iterasi yang lebih sedikit, tingkat kesalahan yang lebih rendah, serta waktu komputasi yang lebih cepat pada seluruh ukuran matriks yang diuji.

4. KESIMPULAN

Penelitian ini telah mengevaluasi dan membandingkan kinerja metode iteratif Jacobi dan Gauss-Seidel dalam penyelesaian sistem persamaan linear dominan diagonal menggunakan lingkungan komputasi Python. Hasil pengujian membuktikan bahwa kedua metode tersebut mampu mencapai kondisi konvergensi dengan stabil dan menghasilkan solusi numerik yang sangat mendekati solusi acuan.

Secara komparatif, metode Gauss-Seidel menunjukkan performa yang secara konsisten lebih unggul dibandingkan metode Jacobi pada seluruh skenario ukuran matriks yang diuji. Pada pengujian matriks skala terbesar (10×10), metode Gauss-Seidel berhasil mencapai konvergensi dalam 20 iterasi, sedangkan metode Jacobi membutuhkan 32 iterasi, yang merepresentasikan reduksi jumlah iterasi sebesar 37,50%. Pengurangan iterasi ini berbanding lurus dengan efisiensi waktu komputasi, di mana Gauss-Seidel mencatatkan nilai speedup rata-rata sebesar 1,85 kali lipat dibandingkan Jacobi. Keunggulan performa ini secara fundamental didorong oleh cara kerja Gauss-Seidel yang memungkinkan pembaruan matriks secara langsung (in-place update) menggunakan nilai terbaru. Pendekatan ini secara efektif mengeliminasi beban waktu (overhead) akibat alokasi dan dealokasi memori berulang yang terjadi pada metode Jacobi di dalam pustaka NumPy, terlepas dari fakta bahwa kedua metode memiliki tingkat kompleksitas asimtotik yang sama, yakni $O(n^2)$.

Berdasarkan keseluruhan hasil evaluasi tersebut, metode Gauss-Seidel sangat direkomendasikan sebagai algoritma yang lebih optimal dan efisien untuk penyelesaian sistem persamaan linear dominan diagonal pada perangkat lunak berbasis Python. Sebagai langkah pengembangan, penelitian selanjutnya diharapkan dapat memperluas pengujian pada matriks sparse berdimensi sangat besar, matriks dengan kondisi tidak dominan diagonal, serta melakukan komparasi performa dan manajemen memori dengan metode numerik lanjutan seperti Successive Over Relaxation (SOR), Conjugate Gradient, maupun metode hybrid lainnya.

REFERENCES

- Agina, P. C., & Echezona, N. G. (2022). Comparative analysis of the Gauss-Seidel and Jacobi methods for solving linear systems. ResearchGate. https://www.researchgate.net/publication/360933343_COMPARATIVE_ANALYSIS_OF_THE_GAUSS-SEIDEL_AND_JACOBI_METHODS_FOR_SOLVING_LINEAR_SYSTEMS
- Alomair, A. M., Shah, F. A., Ahmed, K., & Waseem, M. (2024). Generalized and novel iterative scheme for best approximate solution of large and sparse augmented linear systems. *Heliyon*, 10, e35694. https://drive.google.com/file/d/1TgCXwJ3d_CB4xUG_iKsoinehXyeJw41a/view?usp=drivesdk
- Ardhani, D. C., Solikah, N. H., & Wibowo, A. (2025). Pemanfaatan Python dalam penyelesaian SPL dengan metode iterasi Jacobi dan Gauss-Seidel. *J-PiMat: Jurnal Pendidikan Matematika*, 7(1), 1685-1694. <https://jurnal.stkipppersada.ac.id/jurnal/index.php/jpmat/article/download/4665/pdf>
- Argyros, I. K. (2022). *The Theory and Applications of Iteration Methods* (2nd ed.). CRC Press. <https://drive.google.com/file/d/1sU7l-A-nqda1jQ-3webSShlpSTVzew-a/view?usp=drivesdk>
- Chapra, S. C., & Canale, R. P. (2021). *Numerical methods for engineers* (8th ed.). McGraw-Hill Education. <https://id.scribd.com/document/985418247/Numerical-Methods-for-Engineers-8th-Edition-Chapra-PDF>
- Gupta, A., Sharma, P., & Singh, R. (2023). Comparative analysis of iterative methods for solving sparse linear systems. *International Journal of Computational Mathematics*, 100(7), 1489–1504. https://drive.google.com/file/d/14vgFraj4RfRDTbNVGncwI_DdKFESEy6c/view?usp=drivesdk
- Hasanah, A. (2025). Perbandingan metode Gauss-Seidel, metode Newton-Raphson, dan metode Broyden dalam penyelesaian sistem persamaan nonlinier. <https://drive.google.com/file/d/1U7Qu7JxTpLiz4X3tr1EuBM-ZYE63FcL2/view?usp=drivesdk>
- Hernandez, J., Morales, P., & Castillo, R. (2021). Convergence behavior of classical iterative methods for linear algebraic systems. *Computers & Mathematics with Applications*, 88, 34–46. <https://scispace.com/journals/computers-mathematics-with-applications-1354r9ia>
- Ihsan, H., Wahyuni, M. S., & Waode, Y. S. (2024). Penerapan metode iterasi Jacobi dan Gauss–Seidel dalam menyelesaikan sistem persamaan linear kompleks. *Journal of Mathematics, Computations and Statistics*, 7(2), 112–121. https://drive.google.com/file/d/1gVD4Mlnbu0fEqGvdATcm3_q0fKmUWmd7/view?usp=drivesdk
- Kalyani, M. N., Kooshkghazi, M. N., & Reichel, L. (2025). Block iterative methods for solving multilinear systems. *Numerical Algorithms*, 98(1), 67–84. <https://drive.google.com/file/d/1M-Cc6ABofH3L0GInxC6jlmnVJsSfsZRa/view?usp=drivesdk>
- Kealoha, D., Rojas, F., & Li, X. (2024). Modeling and simulation of traffic on I-485 via linear systems and iterative methods. *arXiv preprint, arXiv:2408.06511*. <https://doi.org/10.48550/arXiv.2408.06511>
- Kovač, M., Tadić, S., Krstić, M., & Veljović, M. (2023). A methodology for planning city logistics concepts based on city-dry port micro-consolidation centres. *Mathematics*, 11(15), 3347. <https://www.mdpi.com/2227-7390/11/15/3347>
- 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 Dergisi*, 4(1), 15–24. <https://drive.google.com/file/d/11VUc4atOVvIM8le-iS7ED0GI0W3Su-Mv/view?usp=drivesdk>
- Novati, P., Tagliaferro, F., & Zennaro, M. (2024). A general class of iterative splitting methods for solving linear systems. *arXiv preprint, arXiv:2404.06800*. <https://doi.org/10.48550/arXiv.2404.06800>
- Rahayu, P. I., Hidayatullah, M., & Hijrah, M. (2023). Implementation Vector Autoregressive (VAR) on rice production and rice productivity data in Indonesia. *Matematika, Statiska dan Komputasi*, 19(3), 580-592. <https://journal.unhas.ac.id/index.php/jmsk/article/view/24881>

- Saad, Y. (2023). Iterative methods for sparse linear systems (3rd ed.). Society for Industrial and Applied Mathematics (SIAM).
<https://drive.google.com/file/d/1oCLoJ2q8JFfWFYzw1fNP0HmPaCWhRnNd/view?usp=drivesdk>
- Safitri, N., Saliha, P., Syamsudin, S., Irawansyah, M. P., Dzaky, M., & Jarti, N. (2026). Analisis performa metode Jacobi dan Gauss-Seidel pada sistem persamaan linear berdimensi besar. *Jurnal Sains Informatika Terapan*, 5(2). <https://doi.org/10.62357/jsit.v5i2.1168>
- Saucedo-Mora, L., & Irastorza-Valera, L. (2025). General form of the Gauss-Seidel equation to linearly approximate the Moore-Penrose pseudoinverse in random non-square systems and high order tensors. *arXiv preprint*, arXiv:2503.22273. <https://doi.org/10.48550/arXiv.2503.22273>
- Shen, D., Chen, L., & Liang, H. (2023). Fast converging Gauss-Seidel iterative algorithm for massive MIMO systems. *Applied Sciences*, 13(7), 4218.
https://drive.google.com/file/d/16BSN9KhDjVcq6wCeU_bPsjRBT49JdeK/view?usp=drivesdk
- Sukarna, S., Abdy, M., & Rahmat, R. (2020). Perbandingan metode iterasi Jacobi dan metode iterasi Gauss-Seidel dalam menyelesaikan sistem persamaan linear fuzzy. *Journal of Mathematics Computations and Statistics*, 2(1).
https://www.researchgate.net/publication/347070591_Perbandingan_Metode_Iterasi_Jacobi_dan_Metode_Iterasi_Gauss-Seidel_dalam_Menyelesaikan_Sistem_Persamaan_Linear_Fuzzy
- Suli, E., & Mayers, D. F. (2022). An introduction to numerical analysis (2nd ed.). Cambridge University Press.
<https://drive.google.com/file/d/1HJak8HJKzw15O-5RH80XD8ICz01Fw6DO/view?usp=drivesdk>
- Syata, I., Suriadi, N. P., & Halim, H. N. (2023). Nilai Konsentrasi Unsur Paracetamol dan Kafein yang Membentuk Sistem Persamaan Taklinear dengan Metode Broyden. *Jurnal Matematika dan Statistika serta Aplikasinya*, 11(1). <https://doi.org/10.24252/msa.v11i1.37744>
- Vojinovic, M., Markovic, S., & Jovanovic, N. (2025). A parallel iterative hybrid Gauss-Jacobi-Seidel method for solving linear systems. *Journal of Computational and Applied Mathematics*, 451, 116032. <https://www.sciencedirect.com/science/article/abs/pii/S0377042725001426>