TRIGONOMETRI
ISSN 3030-8496

Jurnal Matematika dan Ilmu Pengetahuan Alam
Vol 5 No 2 Tahun 2024.
Prefix DOI : 10.8734/trigo.v1i2.365

PENYELESAIAN PERSAMAAN LINEAR MENGGUNAKAN METODE JACOBI DAN GAUSS SEIDEL DENGAN APLIKASI MATLABA

Seila Amalia[1*], Sandy Dwi Payana[2], Alya Nabilla Putri[3], Enjelita Simangunsong[4]
Prodi Statistika, Fakultas FMIPA, Universitas Negeri Medan
Email : seilaamaliaa21@gmail.com[1], payanasandidwi@gmail.com[2], nabillaputrialya@gmail.com[3], enjelitasimangunsong@gmail.com[4]

ABSTRAK

Penyelesaian sistem persamaan linear merupakan salah satu topik fundamental dalam matematika numerik, yang memiliki berbagai aplikasi dalam ilmu teknik dan ilmu komputer. Dua metode iteratif yang sering digunakan untuk menyelesaikan sistem persamaan linear adalah metode Jacobi dan Gauss-Seidel. Artikel ini bertujuan untuk membandingkan kedua metode tersebut dalam menyelesaikan sistem persamaan linear dan mengimplementasikannya menggunakan perangkat lunak matlab. Metode Jacobi mengandalkan penggunaan nilai lama dari setiap variabel dalam iterasi, sedangkan metode Gauss-Seidel menggunakan nilai yang telah diperbarui dalam iterasi sebelumnya. Keduanya diterapkan pada beberapa contoh sistem persamaan linear, dan hasilnya dibandingkan berdasarkan konvergensi serta akurasi solusi. Dalam artikel ini, dibahas juga cara implementasi algoritma-metode tersebut dalam matlab dan analisis kecepatan konvergensi yang tercapai. Hasil eksperimen menunjukkan bahwa metode Gauss-Seidel lebih cepat konvergen dibandingkan metode Jacobi, meskipun kedua metode ini memiliki kelebihan dan keterbatasan tergantung pada sifat sistem persamaan yang diselesaikan.

ABSTRACT

The solution of linear systems of equations is a fundamental topic in numerical mathematics, with wide applications in engineering and computer science. Two iterative methods commonly used to solve linear systems are the Jacobi method and the Gauss-Seidel method. This paper aims to compare these two methods in solving linear systems and to implement them using MATLAB software. The Jacobi method relies on the use of the previous values of each variable during iteration, while the Gauss-Seidel method uses the updated values from the previous iteration. Both methods are applied to several example linear systems, and the results are compared based on convergence and solution accuracy. Additionally, the implementation of these algorithms in MATLAB and the analysis of their convergence speed are discussed. Experimental results show that the Gauss-Seidel method converges faster than the

Article History
Received: Desember 2024
Reviewed: Desember 2024
Published: Desember 2024

Plagirism Checker No 223
DOI : 10.8734/Trigo.v1i2.365
Copyright : Author
Publish by : Trigonometri

TRIGONOMETRI
ISSN 3030-8496

Jurnal Matematika dan Ilmu Pengetahuan Alam
Vol 5 No 2 Tahun 2024.
Prefix DOI : 10.8734/trigo.v1i2.365

Jacobi method, although both methods have their strengths and limitations depending on the characteristics of the system being solved.
Keywords: *Acceptance Status; Binary Logistic Regression; Job Applicants*

PENDAHULUAN

Menyelesaikan suatu sistem persamaan linier dibutuhkan metode yang digunakan untuk mendapatkan suatu solusi. Ada dua metode yang digunakan dalam menyelesaikan persamaan linier, yaitu metode langsung dan tidak langsung Metode langsung merupakan metode di mana prosesnya melalui langkah-langkah berhingga untuk mendapatkan solusi yang memenuhi suatu persamaan linier. Metode Eliminasi Gauss, Metode Eliminasi Gauss-Jordan, Metode Matriks Invers, aturan Cramer, dan Metode Dekomposisi Lower-Upper (LU) merupakan contoh metode langsung. Metode tidak langsung dikenal sebagai metode iteratif dan membutuhkan nilai awal (solusi). Metode Iterasi Jacobi, Metode Gauss-Seidel, dan Metode Successive Over Relaxation (SOR) merupakan beberapa contoh metode tidak langsung[1].

Metode Gauss-Seidel adalah salah satu metode tidak langsung yang digunakan dalam menyelesaikan sistem persamaan linier melalui proses berulang atau iterasi untuk mendapatkan nilai solusi. Pada Metode Gauss-Seidel ini menggunakan nilai awal kemudian nilai yang diperoleh dari iterasi sebelumnya digunakan untuk iterasi berikutnya.Metode gaus-Seidel dapat diterapkan pada matriks koefisien dengan syarat bahwa matriks tersebut bersifat "strictly diagonally dominant" di mana elemen diagonalnya memiliki nilai yang lebih besar dibandingkan dengan jumlah nilai dari non-diagonal pada setiap baris. Kelebihan Metode Gauss-Seidel adalah proses iteratifnya lebih cepat untuk mencapai solusi konvergen [2].

Metode Gauss-Seidel tidak hanya digunakan dalam menyelesaikan system persamaan linier dengan variabel dan konstanta bilangan riil tetapi dapat digunakan untuk menyelesaikan sistem persamaan linier dengan variabel dan konstanta berupa bilangan fuzzy Bilangan fuzzy merupakan teori fundamental yang ada pada logika fuzzy. Logika fuzzy adalah evolusi dari logika Boolean yang hanya menerima nilai salah (0) atau benar (1). Pada logika fuzzy memungkinkan adanya nilai antara salah (0) dan benar (1).

Nilai fungsi keanggotaan yang berada di antara 0 sampai 1 disebut sebagai bilangan fuzzy. Bilangan fuzzy didefinisikan sebagai perluasan dari bilangan tegas (crisp), sebab dalam bilangan tegas (crisp) fungsi keanggotaannya bernilai 0 atau 1[3].

Untuk menyelesaikan sistem persamaan linear, ada berbagai metode yang dapat digunakan. Metode langsung seperti eliminasi Gauss biasanya digunakan untuk system persamaan berukuran kecil hingga sedang, karena metode ini memberikan solusi yang tepat dalam beberapa langkah operasi. Namun, untuk sistem dengan ukuran besar atau sistem yang memiliki matriks jarang terisi penuh (sparse matrix), metode iterasi sering lebih disukai. Metode iterasi memberikan solusi mendekati dengan melakukan perhitungan berulang kali hingga hasil yang diinginkan diperoleh. Salah satu metode iterasi yang paling umum digunakan adalah metode iterasi Jacobi, yang sederhana namun efisien untuk sistem tertentu. Metode Jacobi adalah salah satu metode iterasi yang digunakan untuk menyelesaikan sistem persamaan linear.

Prinsip dasar dari metode ini adalah memisahkan variabel yang dicari di setiap persamaan, sehingga setiap elemen solusi pada iterasi berikutnya hanya bergantung pada nilai iterasi sebelumnya[4].

TRIGONOMETRI
ISSN 3030-8496
Jurnal Matematika dan Ilmu Pengetahuan Alam
Vol 5 No 2 Tahun 2024.
Prefix DOI : 10.8734/trigo.v1i2.365

TINJAUAN PUSTAKA

Sistem persamaan linear merupakan kumpulan persamaan linear yang saling berhubungan untuk mencari nilai variabel yang memenuhi semua persamaan linear tersebut. Sistem persamaan linier kadang muncul secara langsung dari masalah-masalah yang nyata sehingga membutuhkan proses penyelesaian. Menyelesaikan suatu persamaan linear adalah mencari nilai-nilai variable yang memenuhi semua persamaan linear yang diberikan. Sistem persamaan linear biasanya terdiri atas m persamaan dan n variabel. Sistem persamaan linear dapat ditulis dalam bentuk persamaan matriks Ax = b dengan semua entri-entri di dalam A dan b adalah bilangan riil. Secara umum sistem persamaan linear dapat diselesaikan dengan dua metode yaitu metode langsung dan metode tidak langsung. Metode langsung biasanya disebut metode eksak, diantaranya metode eliminasi, subtitusi, dekomposisi LU, dekomposisi Cholesky, dan dekomposisi Crout. Metode tidak langsung biasanya disebut iterasi , diantaranya metode iterasi Jacobi, metode SOR, metode Gauss-Seidel. Metode iterasi Jacobi merupakan salah satu metode tak langsung, yang bermula dari suatu hampiran penyelesaian awal dan kemudian berusaha memperbaiki hampiran dalam tak berhingga namun langkah konvergen sedangkan Metode Gauss Seidel merupakan metode yang menggunakan proses iterasi hingga diperoleh nilai yang sesungguhnya. Metode ini menggunakan nilai awal dan pada proses selanjutnya menggunakan nilai yang sudah diketahui sebelumnya.

Pemanfaatan Matlab yang identik dengan matriks tentu erat kaitannya dengan bidang matematika dan komputasi. Berbagai permasalah matematika dapat dengan mudah dicari penyelesaiannya dengan Maltab, begitu pun dengan bidang komputasi. Matlab merupakan bahasa pemrograman level tinggi yang dikhususkan untuk kebutuhan komputasi teknis, visualisasi dan pemrograman seperti komputasi matematik, analisis data, pengembangan algoritma, simulasi dan pemodelan dan grafik-grafik perhitungan(Firmansyah, A. 2007). Masih dalam tulisan yang sama, ia mengungkapkan bahwa tugastugas matematika yang membutuhkan analisis ataupun perhitungan yang kompleks dan rumit dapat kita implementasikan dengan mudah dengan memanfaatkan fasilitas yang tedapat di Matlab. Pada bidang komputasi, , Matlab dapat digunakan untuk akuisisi citra digital, termasuk juga dalam hal pengembangan dan algoritma. Permodelan, simulasi sampai pembuatan prototype juga berkaitan erat dengan komputasi.

Oleh karena itu, pada artikel ini membahas perbandingan dua metode yaitu metode Iterasi Jacobi dan Iterasi Gauss Seidel dalam menyelesaikan sistem persamaan linear dengan memperhatikan banyaknya iterasi dan nilai error pada suatu hasil iterasi tersebut menggunakan aplikasi MATLAB.

METODE ITERASI JACOBI

Metode iterasi Jacobi ini digunakan untuk menyelesaikan persamaan linier yang proporsi koefisien nol nya besar. Metode ini ditemukan olek matematikawan yang berasal dari Jerman, Carl Gustav Jakob Jacobi. Penemuan ini diperkirakan pada tahun 1800-an. Iterasi dapat diartikan sebagai suatu proses atau metode yang digunakan secara berulang-ulang (pengulangan) dalam menyelesaikan suatu permasalahan matematika. Adapun metode iterasi Jacobi yaitu:

$$\boldsymbol{x_i^{(k)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j \neq i}^{n} a_{ij}\, x_j^{(k-1)} \right), i = 1, 2, \dots, n; k = 1, 2, 3, \dots, n} \qquad (1)$$

TRIGONOMETRI
ISSN 3030-8496

Jurnal Matematika dan Ilmu Pengetahuan Alam
Vol 5 No 2 Tahun 2024.
Prefix DOI : 10.8734/trigo.v1i2.365

METODE ITERASI GAUSS-SEIDEL

Metode Gauss-Seidel digunakan untuk menyelesaikan sistem persamaan linier (SPL) berukuran besar dan proporsi koefisien nolnya besar, seperti sistem-sistem yang banyak ditemukan dalam sistem persamaan diferensial. Teknik iterasi jarang digunakan untuk menyelesaikan SPL berukuran kecil karena metode-metode langsung seperti metode eliminasi Gauss lebih efisien daripada metode iteratif. Akan tetapi, untuk SPL berukuran besar dengan persentase elemen nol pada matriks koefisien besar, teknik iterasi lebih efisien daripada metode langsung dalam hal penggunaan memori komputer maupun waktu komputasi. Metode iterasi Gauss-Seidel dapat dinyatakan sebagai berikut:

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

GALAT

Misalkan $\bar{x}$ suatu nilai hampiran numerik untuk nilai numerik eksak x, yang tidak diketahui. Nilai $e_{\bar{x}} = x - \bar{x}$

METODOLOGI

Metodologi yang diterapkan dalam penelitian ini melibatkan beberapa langkah terstruktur untuk menganalisis dan membandingkan efektivitas metode Jacobi dan Gauss-Seidel dalam menyelesaikan sistem persamaan linear.

Proses dimulai dengan melakukan kajian literatur untuk mendapatkan pemahaman yang mendalam tentang teori dasar kedua metode, termasuk kelebihan dan kekurangan masing-masing. Setelah itu, algoritma untuk kedua metode dikembangkan dan diimplementasikan dalam perangkat lunak MATLAB, yang mencakup penentuan parameter awal, pembaruan nilai variabel, dan perhitungan error di setiap iterasi. Simulasi dilakukan dengan menggunakan contoh sistem persamaan linear yang telah ditentukan, di mana setiap metode diuji untuk mengamati proses iterasi dan konvergensi. Hasil dari simulasi kemudian dianalisis dengan menekankan jumlah iterasi yang diperlukan untuk mencapai toleransi error yang ditetapkan, serta membandingkan kecepatan konvergensi dan efisiensi waktu komputasi antara kedua metode. Untuk memperjelas hasil, data dari iterasi dan error divisualisasikan dalam bentuk grafik, sehingga memudahkan pemahaman dan perbandingan performa kedua metode.

Dengan pendekatan ini, penelitian bertujuan untuk memberikan wawasan yang jelas mengenai efektivitas metode Jacobi dan Gauss-Seidel dalam menyelesaikan sistem persamaan linear.

PEMBAHASAN

Metode Jacobi

1) Definisi Parameter

```
/MATLAB Drive/untitled123.m
% Definisi parameter
A = [4 -1 1; -2 6 -1; 1 3 4]; % Matriks koefisien
b = [5; -9; 8]; % Vektor konstanta
x = zeros(3, 1); % Nilai awal x
tol = 1e-6; % Toleransi error
max_iter = 100; % Jumlah iterasi maksimum
error = inf; % Awal error
iter = 0; % Inisialisasi iterasi
errors = []; % Untuk menyimpan error tiap iterasi
```

- **Fungsi**: Mendefinisikan parameter awal untuk metode Jacobi.
- A adalah matriks koefisien dari sistem persamaan.

TRIGONOMETRI
ISSN 3030-8496
Jurnal Matematika dan Ilmu Pengetahuan Alam
Vol 5 No 2 Tahun 2024.
Prefix DOI : 10.8734/trigo.v1i2.365

- b adalah vektor konstanta yang berkorespondensi dengan A.
- x = zeros(3, 1) adalah nilai awal vektor x yang diasumsikan sebagai nol.
- tol adalah nilai toleransi untuk menentukan kapan iterasi berhenti.
- max_iter adalah batas maksimum iterasi.
- error dan iter diinisialisasi untuk memulai proses iterasi.
- errors akan menyimpan nilai error pada setiap iterasi.

2) Tabel Header

```
% Tabel header
fprintf('Iterasi\tx1\t\tx2\t\tx3\t\tError\n');
fprintf('-------------------------------------------------------\n');
```

- **Fungsi**: Menampilkan header untuk tabel iterasi yang akan diikuti oleh hasil perhitungan setiap iterasi.
- fprintf digunakan untuk menampilkan teks format dalam bentuk tabel agar mudah dipahami.

3) Metode Jacobi (Loop While)

```
% Metode Jacobi
while error > tol && iter < max_iter
    x_new = x; % Menyimpan nilai baru x
    x_new(1) = (b(1) - (-A(1,2)*x(2) - A(1,3)*x(3))) / A(1,1);
    x_new(2) = (b(2) - (-A(2,1)*x(1) - A(2,3)*x(3))) / A(2,2);
    x_new(3) = (b(3) - (A(3,1)*x(1) - A(3,2)*x(2))) / A(3,3);
```

- **Fungsi**: Menghitung nilai x_new pada setiap iterasi dengan metode Jacobi.
- x_new adalah vektor yang menyimpan nilai baru x berdasarkan persamaan iterasi Jacobi.
- Pada setiap elemen x_new, nilai diperbarui menggunakan formula Jacobi, di mana nilai saat ini tidak dipengaruhi oleh nilai yang baru saja diperbarui dalam iterasi yang sama.

4) Menghitung Error dan Update Nilai x

```
% Menghitung error
error = norm(x_new - x);
errors = [errors; error]; % Menyimpan error di setiap iterasi

% Update nilai x
x = x_new;
iter = iter + 1;
```

- Fungsi: Menghitung error sebagai norma (jarak Euclidean) antara x_new dan x.
- errors = [errors; error] menyimpan error dalam vektor errors agar bisa diplot atau dianalisis setelah iterasi selesai.
- x = x_new memperbarui nilai x untuk iterasi selanjutnya.
- iter = iter + 1 meningkatkan jumlah iterasi.

5) Menampilkan Tabel Iterasi

```
    % Tampilkan tabel iterasi
    fprintf('%d\t\t%.6f\t%.6f\t%.6f\t%.6f\n', iter, x(1), x(2), x(3), error);
end
```

- **Fungsi**: Menampilkan hasil setiap iterasi dalam bentuk tabel.
- fprintf digunakan untuk menampilkan nilai iter, x(1), x(2), x(3), dan error dengan format desimal 6 angka di belakang koma.

TRIGONOMETRI
ISSN 3030-8496
Jurnal Matematika dan Ilmu Pengetahuan Alam
Vol 5 No 2 Tahun 2024.
Prefix DOI : 10.8734/trigo.v1i2.365

6) Menampilkan Hasil Akhir

```
ive/untitled123.m
% Menampilkan hasil akhir
disp('Solusi x:');
disp(x);
disp(['Jumlah iterasi: ', num2str(iter)]);
disp(['Error akhir: ', num2str(error)]);
```

- **Fungsi**: Menampilkan solusi akhir yang diperoleh setelah proses iterasi selesai.
- disp('Solusi x:') menampilkan teks untuk menjelaskan bahwa x adalah solusi akhir.
- disp(x) menampilkan nilai vektor x, yang merupakan solusi dari sistem persamaan.
- disp(['Jumlah iterasi: ', num2str(iter)]) menampilkan jumlah iterasi yang dilakukan hingga konvergensi atau mencapai batas iterasi maksimum.
- disp(['Error akhir: ', num2str(error)]) menampilkan nilai error terakhir yang menunjukkan seberapa dekat solusi terhadap hasil yang benar.

7) Plot Grafik Error

```
% Plot grafik error
figure;
plot(1:iter, errors, '-o');
xlabel('Iterasi');
ylabel('Error');
title('Konvergensi Metode Jacobi');
grid on;
```

- **Fungsi**: Membuat grafik untuk memvisualisasikan konvergensi metode Jacobi berdasarkan error pada setiap iterasi.
- figure membuka jendela baru untuk grafik.
- plot(1:iter, errors, '-o') membuat grafik menggunakan vektor errors (yang menyimpan error tiap iterasi) sebagai sumbu y dan nomor iterasi 1:iter sebagai sumbu x. Tanda -o digunakan untuk menggambar garis dengan titik-titik pada setiap iterasi.
- xlabel('Iterasi') dan ylabel('Error') memberi label pada sumbu x dan y agar grafik lebih informatif.
- title('Konvergensi Metode Jacobi') menambahkan judul yang menjelaskan bahwa grafik tersebut menunjukkan konvergensi metode Jacobi.
- grid on menambahkan garis grid pada grafik untuk membantu visualisasi data.

Loop While dan Error Calculation: Kode ini digunakan untuk memastikan metode iterasi Jacobi berjalan sampai hasilnya berada dalam toleransi yang ditentukan (tol) atau mencapai batas iterasi (max_iter). Error dihitung setiap kali, dan nilai-nilai disimpan untuk memantau konvergensi.

Menampilkan Hasil dan Visualisasi Error: Bagian ini berguna untuk melihat hasil akhir dari solusi dan memahami seberapa cepat metode Jacobi mencapai konvergensi. Grafik konvergensi menunjukkan apakah error berkurang di setiap iterasi, yang menandakan efektivitas metode dalam mencapai solusi.

TRIGONOMETRI
ISSN 3030-8496

Jurnal Matematika dan Ilmu Pengetahuan Alam
Vol 5 No 2 Tahun 2024.
Prefix DOI : 10.8734/trigo.v1i2.365

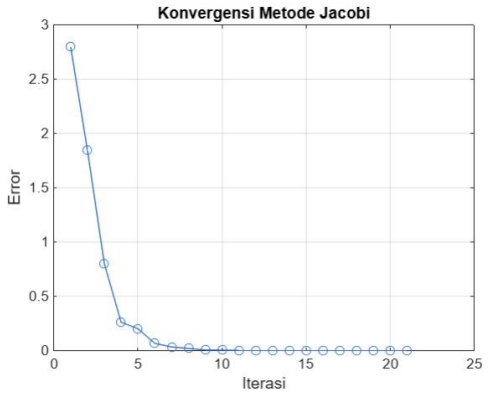
Tabel Hasil Iterasi :

Iterasi	x1	x2	x3	Error
1	1.250000	-1.500000	2.000000	2.795085
2	2.125000	-2.250000	0.562500	1.842425
3	1.953125	-2.302083	-0.218750	0.801627
4	1.770833	-2.114583	-0.214844	0.261537
5	1.724935	-2.054470	-0.028646	0.200972
6	1.756456	-2.070204	0.027913	0.066634
7	1.774529	-2.090138	0.008233	0.033336
8	1.774593	-2.092882	-0.011236	0.019661
9	1.770412	-2.089658	-0.013310	0.005672
10	1.769087	-2.087919	-0.009847	0.004095
11	1.769518	-2.088055	-0.008211	0.001697
12	1.769961	-2.088471	-0.008420	0.000643
13	1.770013	-2.088584	-0.008843	0.000441
14	1.769935	-2.088530	-0.008941	0.000135
15	1.769897	-2.088488	-0.008881	0.000082
16	1.769902	-2.088486	-0.008841	0.000041
17	1.769911	-2.088494	-0.008840	0.000013
18	1.769914	-2.088497	-0.008848	0.000010
19	1.769912	-2.088496	-0.008851	0.000003
20	1.769911	-2.088496	-0.008850	0.000002
21	1.769911	-2.088495	-0.008849	0.000001

```
Solusi x:
    1.7699
   -2.0885
   -0.0088

Jumlah iterasi: 21
Error akhir: 9.6453e-07
>>
```

Visualisasi Hasil Iterasi :



Metode Gauss Seidel

1) Inisialisasi Parameter

```
rive/gauss.m
% Definisi parameter
A = [4 -1 1; -2 6 -1; 1 3 4]; % Matriks koefisien
b = [5; -9; 8]; % Vektor konstanta
x = zeros(3, 1); % Nilai awal x
tol = 1e-6; % Toleransi error
max_iter = 100; % Jumlah iterasi maksimum
error = inf; % Awal error
iter = 0; % Inisialisasi iterasi
errors = []; % Untuk menyimpan error tiap iterasi
```

- A: Matriks koefisien dari sistem persamaan linear yang ingin diselesaikan. Matriks ini berisi nilai koefisien untuk setiap variabel dalam sistem persamaan.
- b: Vektor konstanta yang berisi nilai-nilai yang berada di sisi kanan dari persamaan linear.
- x: Vektor solusi yang diinisialisasi dengan nilai awal nol. Vektor ini akan diperbarui di setiap iterasi hingga mencapai solusi yang mendekati.
- tol: Toleransi error yang menentukan kapan iterasi dihentikan. Jika error (selisih antara iterasi sebelumnya dan saat ini) lebih kecil dari tol, maka iterasi akan dihentikan.

TRIGONOMETRI
ISSN 3030-8496
Jurnal Matematika dan Ilmu Pengetahuan Alam
Vol 5 No 2 Tahun 2024.
Prefix DOI : 10.8734/trigo.v1i2.365

- max_iter: Batas maksimum iterasi. Jika jumlah iterasi mencapai max_iter sebelum error mencapai tol, iterasi akan dihentikan untuk menghindari looping tak terbatas.
- error: Diinisialisasi dengan nilai tak terhingga (inf) sebagai tanda awal agar iterasi pertama dijalankan. Nantinya, error akan dihitung berdasarkan selisih antara nilai x di iterasi sebelumnya dan iterasi saat ini.
- iter: Variabel penghitung jumlah iterasi yang diinisialisasi dengan nilai nol.
- errors: Array kosong untuk menyimpan nilai error pada setiap iterasi, yang nantinya bisa digunakan untuk melihat perkembangan konvergensi.

2) Tabel Header

```
% Tabel header
fprintf('Iterasi\tx1\t\tx2\t\tx3\t\tError\n');
fprintf('-------------------------------------------------------\n');
```

- **Fungsi**: fprintf digunakan untuk mencetak header tabel yang menunjukkan kolom-kolom hasil iterasi. Kolom ini mencakup Iterasi, x1, x2, x3, dan Error agar hasil dari setiap iterasi mudah dibaca dan dianalisis.
- Tabel ini berfungsi sebagai dokumentasi visual dari setiap iterasi sehingga kita bisa melacak perubahan nilai solusi x dan error dari satu iterasi ke iterasi berikutnya.

3) **Metode Gauss-Seidel (Perulangan Utama)**

```
% Metode Gauss-Seidel
while error > tol && iter < max_iter
    x_old = x; % Menyimpan nilai x sebelumnya
```

- **Fungsi**: Perulangan while ini menjalankan proses iterasi sampai dua kondisi terpenuhi: (1) error lebih kecil dari tol, dan (2) jumlah iterasi iter kurang dari max_iter.
- _old = x;: Menyimpan nilai x dari iterasi sebelumnya ke dalam variabel x_old. Nilai ini digunakan untuk menghitung error pada akhir iterasi.
- **Metode Gauss-Seidel**: Metode ini menggunakan nilai terbaru dari setiap variabel x(i) segera setelah diperbarui dalam iterasi yang sama. Hal ini mempercepat konvergensi dibandingkan metode Jacobi.

4) Update Nilai x1, x2, x3 Menggunakan Nilai Baru

```
% Update x1, x2, x3 menggunakan nilai yang paling baru di setiap iterasi
x(1) = (b(1) - (-A(1,2)*x(2) - A(1,3)*x(3))) / A(1,1);
x(2) = (b(2) - (-A(2,1)*x(1) - A(2,3)*x(3))) / A(2,2);
x(3) = (b(3) - (A(3,1)*x(1) - A(3,2)*x(2))) / A(3,3);
```

- **Fungsi** : Setiap baris ini memperbarui nilai x(i) untuk setiap variabel dalam sistem persamaan.
- x(1) diperbarui dengan menggunakan nilai terbaru x(2) dan x(3) dari iterasi sebelumnya.
- x(2) diperbarui menggunakan nilai terbaru dari x(1) dan nilai sebelumnya dari x(3).
- x(3) diperbarui menggunakan nilai terbaru dari x(1) dan x(2).
- **Tujuan**: Dengan langsung memperbarui dan menggunakan nilai terbaru dari setiap variabel, metode Gauss-Seidel biasanya memiliki konvergensi yang lebih cepat dibandingkan metode Jacobi, yang hanya menggunakan nilai dari iterasi sebelumnya untuk setiap variabel.

TRIGONOMETRI
ISSN 3030-8496

Jurnal Matematika dan Ilmu Pengetahuan Alam
Vol 5 No 2 Tahun 2024.
Prefix DOI : 10.8734/trigo.v1i2.365

5) Menghitung Error

```
% Menghitung error
error = norm(x - x_old);
errors = [errors; error]; % Menyimpan error di setiap iterasi
```

- Fungsi error = norm(x - x_old);: Menghitung error sebagai norma dari selisih antara x di iterasi saat ini dengan x_old dari iterasi sebelumnya. Ini memberi tahu seberapa jauh perubahan dalam solusi setelah satu iterasi.
- errors = [errors; error];: Menyimpan nilai error dalam array errors pada setiap iterasi. Dengan menyimpan error pada setiap iterasi, kita bisa melihat bagaimana error menurun seiring iterasi.

6) Menampilkan Tabel Iterasi

```
    % Tampilkan tabel iterasi
    fprintf('%d\t\t%.6f\t%.6f\t%.6f\t%.6f\n', iter+1, x(1), x(2), x(3), error);

    iter = iter + 1;
end
```

- fprintf: Menampilkan hasil setiap iterasi ke layar dengan format Iterasi, x1, x2, x3, dan Error. Setiap hasil dicetak dengan presisi hingga 6 desimal untuk menunjukkan perubahan kecil di setiap iterasi.
- iter = iter + 1;: Menambah nilai iter untuk melanjutkan ke iterasi berikutnya.

7) Menampilkan Hasil Akhir

```
% Menampilkan hasil akhir
disp('Solusi x:');
disp(x);
disp(['Jumlah iterasi: ', num2str(iter)]);
disp(['Error akhir: ', num2str(error)]);
```

- **Fungsi**: Menampilkan solusi akhir vektor x setelah metode Gauss-Seidel menyelesaikan iterasi.
- disp(x) menampilkan nilai vektor x sebagai solusi dari sistem persamaan.
- disp(['Jumlah iterasi: ', num2str(iter)]) menampilkan jumlah iterasi yang diperlukan untuk mencapai toleransi error yang diinginkan.
- disp(['Error akhir: ', num2str(error)]) menampilkan error terakhir untuk melihat seberapa dekat solusi yang diperoleh.

8) Plot Grafik Error

```
% Plot grafik error
figure;
plot(1:iter, errors, '-o');
xlabel('Iterasi');
ylabel('Error');
title('Konvergensi Metode Gauss-Seidel');
grid on;
```

- **Fungsi**: Membuat grafik yang menunjukkan konvergensi dari error pada setiap iterasi.
- plot(1:iter, errors, '-o') membuat grafik dengan titik-titik setiap iterasi dari errors, yang merupakan vektor penyimpan nilai error pada tiap iterasi.
- xlabel('Iterasi') dan ylabel('Error') memberi label pada sumbu x dan y.
- title('Konvergensi Metode Gauss-Seidel') menambahkan judul pada grafik untuk menjelaskan bahwa ini adalah konvergensi metode Gauss-Seidel.

TRIGONOMETRI
ISSN 3030-8496
Jurnal Matematika dan Ilmu Pengetahuan Alam
Vol 5 No 2 Tahun 2024.
Prefix DOI : 10.8734/trigo.v1i2.365

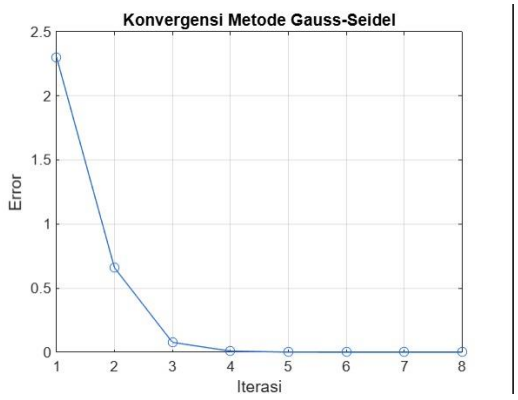
- grid on menambahkan garis grid pada grafik agar lebih mudah melihat titik konvergensi.

Tabel Hasil Iterasi :

```
>> gauss
Iterasi x1            x2            x3            Error
-----------------------------------------------------
1             1.250000      -1.916667       0.250000       2.301871
2             1.791667      -2.138889      -0.052083       0.658817
3             1.771701      -2.081887      -0.004340       0.076989
4             1.769387      -2.089072      -0.009151       0.008952
5             1.769980      -2.088468      -0.008846       0.000900
6             1.769905      -2.088494      -0.008847       0.000079
7             1.769912      -2.088496      -0.008850       0.000007
8             1.769912      -2.088496      -0.008850       0.000001
Solusi x:
    1.7699
   -2.0885
   -0.0088

Jumlah iterasi: 8
Error akhir: 8.3714e-07
>>
```

Visualisasi Hasil Iterasi :



KESIMPULAN

Pembahasan dari penelitian ini menunjukkan bahwa metode Jacobi dan Gauss-Seidel adalah dua teknik iteratif yang digunakan untuk menyelesaikan sistem persamaan linear. Metode Jacobi memungkinkan komputasi paralel karena setiap variabel diperbarui secara independen, meskipun memiliki kecepatan konvergensi yang lebih lambat dan memerlukan lebih banyak iterasi untuk mencapai toleransi error yang sama. Sebaliknya, metode Gauss-Seidel lebih efisien dalam hal waktu komputasi dan jumlah iterasi, karena setiap variabel yang diperbarui langsung digunakan dalam perhitungan variabel lainnya. Namun, metode ini tidak dapat diimplementasikan secara paralel. Pilihan antara kedua metode ini tergantung pada kebutuhan spesifik aplikasi, apakah lebih mengutamakan efisiensi iterasi atau kemampuan komputasi paralel.

DAFTAR PUSTAKA

[1] Sa'adah, M. S. and Alisah, E., "Interpretasi Metode Gauss-Seidel pada Sistem Persamaan Linier Fuzzy dengan Bilangan Fuzzy Sigmoid," *Jurnal Riset Matematika dan Matematika*, vol. 3, no. 5, pp. 223-240, 2024.

[2] Widyastuti, E., Siagian, N. S. B., Manurung, R. R., Ginting, R. R. B., Situmorang, S. A., Pane, S. R. N., and Siringoringo, T. R., "Penerapan Metode Iterasi Jacobi dengan Excel untuk Menyelesaikan Matriks Linear," *Jurnal Pendidikan Tambusai*, vol. 8, no. 3, pp. 41861-41867, 2024.

[3] Sukarna, Abdy, and Rahmat. "Perbandingan Metode Iterasi Jacobi dan Metode Iterasi Gauss-Seidel dalam Menyelesaikan Sistem Persamaan Linear Fuzzy." *Journal of Mathematics,*

Computations, and Statistics, vol. 2, no. 1, pp. 1-18, May 2020.

[4] A. Atina, "Segmentasi Citra Paru Menggunakan Metode k-Means Clustering," *Jurnal Penelitian Fisika dan Terapannya*, vol. 1, no. 1, pp. 28-34, 2019.

[5] Amelia, B. Sistem Persamaan Linear dengan Metode Gauss Seidel. *Jurnal Pustaka Cendekia Pendidikan*, *2*(2), 132-136, 2024.

[6] Chamim, A. N. N., Putra, K. T., & Al Farisi, M. F. Power Flow Analysis of Electrical Network Systems Using Gauss-Seidel Method and Python. *Emerging Information Science and Technology*, *4*(1), 28-36, 2023.