DOI: http://dx.doi.org/10.22441/fifo.2024.v16i1.002

Pengembangan Aplikasi untuk Solusi Persamaan Linear Menggunakan Metode Eliminasi Gauss dengan Substitusi Terbalik

Mohamad Yusuf*[1], Agil Dwiki Yudistira[2], Sebastianus Lukito[3], Alhamdi Rifai[4], Ridho Pangestu[5]

[1,2,3,4,5]Fakultas Ilmu Komputer, Universitas Mercu Buana, Jakarta, Indonesia
[*1]mhd.yusuf@mercubuana.ac.id, [2]41522110068@student.mercubuana.ac.id
[3]41522110051@student.mercubuana.ac.id, [4]41522110035@student.mercubuana.ac.id,
[5]41522110041@student.mercubuana.ac.id

*) Corresponding author

(***received***: 03-04-24, ***revised***: 03-05-24, ***accepted***: 06-05-24)

Abstract

Mathematics plays a crucial role in various aspects of life, particularly in understanding and modeling phenomena using linear equations. The Gaussian Elimination with Back Substitution method is a key technique for solving systems of linear equations, allowing for matrix transformation to simplify equations into a more orderly form. This study focuses on developing a web application that implements this method, aiming to present linear equation solutions in a more interactive manner and facilitate understanding. The application is designed using TypeScript and Next.js, offering a responsive user interface and enabling step-by-step visualization of the solution process. Testing results demonstrate the application's effectiveness in providing accurate solutions, as well as its high performance and usability. This study paves the way for more interactive mathematical learning approaches and enhances the understanding of linear equation concepts and their solution methods.

Keyword: *Linear Equations, Gaussian Elimination, Web Application*

Abstrak

Matematika berperan penting dalam berbagai aspek kehidupan, terutama dalam pemahaman dan pemodelan fenomena menggunakan persamaan linear. Metode Eliminasi Gauss dengan Substitusi Terbalik merupakan teknik kunci dalam menyelesaikan sistem persamaan linear, yang memungkinkan transformasi matriks untuk menyederhanakan persamaan menjadi bentuk yang lebih teratur. Penelitian ini berfokus pada pengembangan aplikasi web yang mengimplementasikan metode ini, bertujuan untuk menyajikan solusi persamaan linear dengan cara yang lebih interaktif dan memudahkan pemahaman. Aplikasi dirancang menggunakan *TypeScript* dan *Next.js*, menawarkan antarmuka pengguna yang responsif dan memfasilitasi visualisasi langkah demi langkah dari proses penyelesaian. Hasil pengujian menunjukkan efektivitas aplikasi dalam memberikan solusi yang akurat, serta kinerja dan kegunaan yang tinggi. Penelitian ini membuka jalan bagi pendekatan pembelajaran matematika yang lebih interaktif dan meningkatkan pemahaman konsep persamaan linear dan metode penyelesaiannya.

Kata Kunci: Persamaan linear, Eliminasi Gauss, Aplikasi Web

I. Pendahuluan

Matematika sebagai fondasi ilmu pengetahuan dan teknologi memiliki peran krusial dalam memahami, menganalisis, dan memodelkan fenomena di berbagai bidang kehidupan. Salah satu konsep matematika yang mendalam dan memiliki dampak besar adalah persamaan linear. Persamaan linear muncul di berbagai konteks, seperti dalam pemodelan perubahan kuantitas terhadap variabel tertentu, pembentukan matriks dalam ilmu komputer, hingga analisis data dalam bidang statistik. Penyelesaian persamaan linear menjadi langkah awal dalam menyusun model matematika yang dapat diaplikasikan dalam kehidupan nyata. Dalam konteks ini, Metode Eliminasi Gauss dengan Substitusi Terbalik telah menjadi salah satu alat utama yang digunakan untuk menangani sistem persamaan linear. Dengan melakukan transformasi matriks, metode ini mampu menyederhanakan sistem persamaan linear menjadi bentuk yang lebih teratur, memudahkan perhitungan dan analisis lebih lanjut [1].

Metode Eliminasi Gauss dengan Substitusi Terbalik adalah salah satu alat yang kuat dalam menangani sistem persamaan linear. Metode ini mengacu pada serangkaian langkah-langkah sistematis untuk mengubah matriks koefisien persamaan linear menjadi bentuk segitiga atas, yang kemudian memungkinkan penentuan solusi dengan melakukan substitusi terbalik [2]. Oleh karena itu, Metode Eliminasi Gauss dengan Substitusi Terbalik bukan hanya menjadi pendekatan teoritis, tetapi juga solusi efisien dalam menangani sistem persamaan linear yang kompleks [3]. Sebuah persamaan linear dapat didefinisikan sebagai persamaan polinomial berderajat satu, di mana variabel-variabelnya muncul dalam bentuk linear, yaitu tidak memiliki pangkat lebih tinggi dari satu [4]. Bentuk umum dari persamaan linear dengan dua variabel x dan y adalah ax + by = c, dan jika ditambahkan variabel z, bentuk umumnya menjadi ax + by + cz = d [5]. Dalam konteks sistem persamaan linear, seringkali digunakan notasi matriks untuk merepresentasikan koefisien-koefisien variabel.

Metode Eliminasi Gauss dengan Substitusi Terbalik merupakan pendekatan sistematis dalam menyelesaikan sistem persamaan linear. Proses ini terdiri dari dua tahap utama: Eliminasi dan substitusi terbalik [6]. Pada tahap eliminasi, matriks koefisien sistem persamaan linear diubah menjadi bentuk segitiga atas [7]. Ini dicapai dengan menerapkan operasi baris elementer untuk menghilangkan koefisien di bawah diagonal utama [8]. Setiap langkah melibatkan pemilihan pivot (elemen bukan nol di kolom saat ini) dan menggunakannya untuk membuat semua elemen di bawah pivot di kolom yang sama menjadi nol [9]. Proses ini terus dilakukan hingga matriks mencapai bentuk segitiga atas. Pada gambar 1 merupakan bentuk umum matriks segitiga atas setelah tahap eliminasi adalah:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ 0 & a_{22} & a_{23} & \cdots & a_{2n} \\ 0 & 0 & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_{nn} \end{bmatrix}$$

Gambar 1. Bentuk Umum Matriks Segitiga Atas Setelah Tahap Eliminasi

Pada tahap substitusi terbalik, nilai variabel yang tidak diketahui dapat ditentukan dengan menggantikan nilai-nilai yang sudah diketahui ke dalam persamaan. Proses ini dimulai dari persamaan terakhir, di mana nilai variabel terakhir dapat dihitung, dan nilai-nilai tersebut kemudian digunakan untuk menghitung variabel-variabel sebelumnya. Langkah ini dilanjutkan hingga semua variabel memiliki nilai yang ditentukan [10].

Dalam perkembangan masyarakat berbasis teknologi informasi, pendidikan telah melihat pergeseran menuju penggunaan teknologi sebagai sarana pembelajaran. Oleh karena itu, diperlukan alat edukasi yang adaptif, interaktif, dan dapat diakses secara mudah untuk membantu mahasiswa dan pelajar dalam memahami konsep matematika yang kompleks seperti persamaan linear. Meskipun Metode Eliminasi Gauss telah memberikan kontribusi besar dalam menyelesaikan persamaan linear, terdapat beberapa kendala dan ketidaklengkapan. Salah satu aspeknya adalah ketidakmampuan dalam menangani sistem persamaan linear yang memiliki jumlah variabel yang berbeda, yang pada kenyataannya seringkali muncul dalam konteks aplikatif. Selain itu, penggunaan metode ini seringkali dihadapkan pada tingkat kesulitan dalam memahami dan menginternalisasi setiap langkah-langkah penyelesaian. Oleh karena itu, diperlukan pendekatan yang lebih visual dan interaktif untuk membantu pengguna memahami konsep tersebut dengan lebih baik. Berdasarkan latar belakang tersebut, penelitian ini bertujuan untuk mengembangkan sebuah aplikasi berbasis web yang mengimplementasikan Metode Eliminasi Gauss dengan Substitusi Terbalik. Aplikasi ini dirancang tidak hanya untuk memberikan solusi numerik dari persamaan linear, tetapi juga untuk menyajikan setiap langkah penyelesaian dengan cara yang lebih interaktif, memudahkan pemahaman dan pembelajaran.

Penelitian ini bertujuan untuk mengembangkan sebuah aplikasi berbasis web yang mengimplementasikan Metode Eliminasi Gauss dengan Substitusi Terbalik, dengan fokus pada peningkatan interaktivitas dan kemudahan pemahaman konsep melalui visualisasi proses penyelesaian persamaan linear. Aplikasi ini dirancang untuk membantu pengguna dalam memahami dan menginternalisasi setiap langkah penyelesaian persamaan linear, sekaligus memberikan solusi numerik yang akurat dan cepat. Melalui penggunaan teknologi web terkini, aplikasi ini diharapkan dapat menjadi alat pembelajaran yang efektif, meningkatkan keterlibatan pengguna, dan memfasilitasi pendidikan matematika yang lebih mendalam dan interaktif.

State of The Art

Penggunaan komputer dalam memecahkan persamaan linear telah berkembang signifikan sepanjang tahun, khususnya melalui aplikasi berbasis web yang memudahkan pembelajaran dan perhitungan. Metode Eliminasi Gauss, sebagai elemen penting dalam matematika komputasi, telah mendapat perhatian yang besar dalam penelitian untuk meningkatkan efisiensi dan efektivitasnya.

Penelitian oleh Sitorus et al (2023) menyoroti penggunaan urutan aritmetika yang lebih sederhana untuk mempercepat proses penyelesaian persamaan. Metode ini menjanjikan efisiensi yang lebih tinggi, terutama dalam aplikasi yang membutuhkan respon cepat .

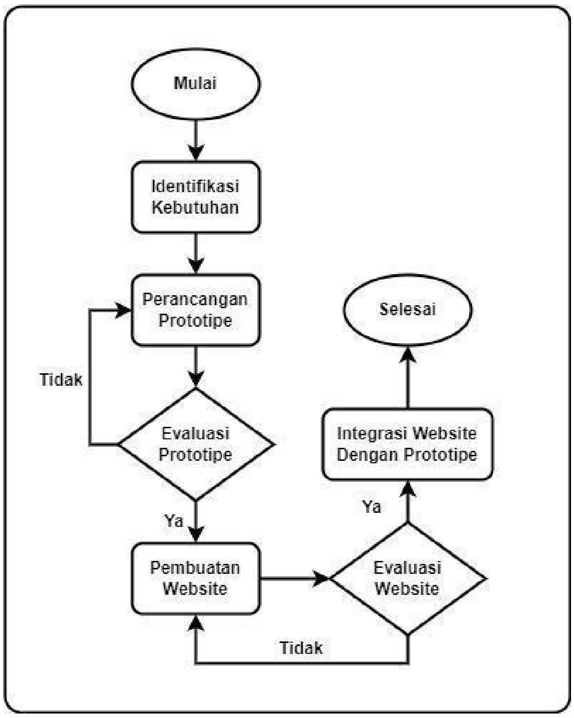
Nur-E-Arefin (2021) menemukan cara baru yang tidak perlu menghitung balik matriks, yang biasanya dilakukan dalam Eliminasi Gauss, sehingga bisa mengurangi kerumitan hitungan dan membuat hasilnya lebih akurat. Ini menunjukkan bahwa aplikasi di internet bisa lebih efisien dengan menggunakan cara ini.

Tewari et al (2021) juga menunjukkan bahwa metode Gauss bisa digunakan dalam banyak hal, dari teknik sampai menyelesaikan masalah sehari-hari. Mereka membandingkan berbagai cara menggunakan Gauss untuk membantu aplikasi di internet memilih yang terbaik sesuai dengan kebutuhan.

Perkembangan terkini dalam penyelesaian persamaan linear menunjukkan peningkatan yang signifikan dalam minat belajar dan memperluas cakupan aplikasi. Dengan metode-metode baru yang lebih efisien dan mudah diintegrasikan, terbuka peluang besar untuk pengembangan aplikasi web yang canggih yang memanfaatkan metode Eliminasi Gauss dan alternatifnya untuk meningkatkan pengalaman belajar..

II. Metodologi Penelitian

Dalam penelitian ini, penulis menggunakan metode *prototype* untuk merancang sistem aplikasi. Prosesnya dilakukan secara berfase, dimulai dengan mengidentifikasi kebutuhan, merancang *prototype*, mengevaluasi prototype, melakukan pengkodean aplikasi, menguji aplikasi, mengevaluasi aplikasi, dan akhirnya implementasi. Tahapan-tahapan ini dapat dilihat dalam diagram alir yang tergambar pada gambar 2 dibawah ini :



Gambar 2. Diagram Alur Penelitian

Pemilihan Teknologi

TypeScript dipilih karena fitur type *safety*-nya yang memastikan keandalan kode saat *runtime*. *Next.js*, di sisi lain, dipilih karena kemampuannya dalam *rendering server-side*, yang meningkatkan performa dan *SEO website*.

Desain Sistem

Aplikasi dirancang dengan arsitektur yang memungkinkan pengguna memasukkan sistem persamaan linier yang diinginkan. Sistem ini terdiri dari antarmuka pengguna yang memungkinkan input persamaan linier, mesin pengolahan yang menerapkan algoritma Eliminasi Gauss dengan substitusi terbalik, dan modul output yang menampilkan langkah-langkah penyelesaian dan solusi akhir. Desain ini memastikan bahwa pengguna dapat dengan mudah memasukkan data, memahami proses penyelesaian, dan menafsirkan hasil**.**

Algoritma Eliminasi Gauss dengan Substitusi Terbalik

Eliminasi Gauss dengan substitusi terbalik merupakan metode sistematis untuk menyelesaikan sistem persamaan linear. Terdapat dua tahap utama dalam metode ini: eliminasi dan substitusi balik.

Pada tahap eliminasi, matriks asli sistem diubah menjadi bentuk segitiga atas. Hal ini dilakukan dengan menerapkan operasi baris elementer untuk menghilangkan koefisien di bawah diagonal utama. Setiap langkah melibatkan pemilihan *pivot* (elemen bukan nol di kolom saat ini) dan menggunakannya untuk membuat semua elemen di bawah *pivot* di kolom yang sama menjadi nol. Proses ini terus dilakukan hingga matriks mencapai bentuk segitiga atas. Bentuk umum matriks segitiga atas adalah

$$\left[a_{11}\ a_{12}\ a_{13} \cdots a_{1n}\ 0\ a_{22}\ a_{23} \cdots a_{2n}\ 0\ 0\ a_{32} \cdots a_{3n}\ \vdots\ \vdots\ \ddots\ \ddots\ \vdots\ 0\ 0\ 0 \cdots a_{nn} \right] \quad (1)$$

Setelah matriks dalam bentuk segitiga atas, nilai variabel yang tidak diketahui dapat ditentukan dengan tahap substitusi balik. Dimulai dari persamaan terakhir, nilai variabel yang ditemukan disubstitusikan ke persamaan sebelumnya. Langkah ini melibatkan penyelesaian untuk variabel terakhir terlebih dahulu dan kemudian menggunakan nilainya untuk menemukan variabel sebelumnya, bergerak ke atas hingga semua variabel ditentukan.

Pengujian

Pengujian aplikasi dilakukan untuk memastikan bahwa semua fitur berfungsi seperti yang diharapkan dan algoritma memberikan solusi yang akurat. Pengujian unit digunakan untuk memverifikasi setiap komponen secara terpisah, sementara pengujian integrasi memastikan bahwa semua komponen bekerja bersama dengan baik. Pengujian pengguna akhir juga dilakukan untuk mengumpulkan umpan balik tentang kegunaan dan memperbaiki masalah yang berkaitan dengan pengalaman pengguna.

III. Hasil dan Pembahasan

Teknik Eliminasi Gauss

Dalam matematika, eliminasi Gauss adalah algoritma yang digunakan untuk menyelesaikan sistem persamaan linear. Metode ini melibatkan serangkaian operasi pada matriks koefisien dari sistem tersebut, yang meskipun mengubah bentuk matriks, tidak mengubah solusi dari sistem persamaan. Operasi ini memungkinkan matriks koefisien diubah menjadi matriks segitiga atas, sehingga solusi dari sistem persamaan dapat ditentukan dengan melakukan eliminasi variabel secara berulang. Eliminasi Gauss juga digunakan untuk menghitung peringkat matriks, determinan dari matriks persegi, dan invers dari matriks nonsingular. Metode ini dinamai dari matematikawan Carl Friedrich Gauss (1777–1855), meskipun beberapa aspek dari metode ini telah diketahui oleh matematikawan Tionghoa sejak tahun 179 M, tetapi tanpa pembuktian formal.

Matriks yang dihasilkan dari algoritma ini akan memiliki bentuk eselon baris (row echelon form, REF). Jika setiap koefisien utama (nilai bukan nol pertama di sebuah baris) matriks adalah 1, dan kolom-kolom yang mengandung koefisien utama memiliki bentuk yang mirip dengan kolom pada matriks identitas, matriks tersebut dikatakan memiliki bentuk eselon baris tereduksi (reduced row echelon form, RREF). Lanjutan dari eliminasi Gauss ke RREF terkadang disebut sebagai eliminasi Gauss-Jordan. Untuk alasan komputasi, operasi baris terkadang dihentikan sebelum matriks mencapai bentuk tereduksinya saat menyelesaikan sistem persamaan.

Kompleksitas komputasi eliminasi Gauss untuk matriks berukuran $n \times n$ adalah n^3.

Dalam bentuk yang paling sederhana, algoritma ini rentan terhadap kesalahan pembulatan, namun ini dapat diatasi dengan menggunakan metode pivot. Ini menjadikannya metode standar untuk menemukan solusi sistem persamaan linear dan menjadi bagian penting dari perpustakaan program aljabar linear seperti NAG, IMSL, dan LAPACK.Dalam matematika, eliminasi Gauss adalah algoritma yang digunakan untuk menyelesaikan sistem persamaan linear. Metode ini melibatkan serangkaian operasi pada matriks koefisien dari sistem tersebut, yang meskipun mengubah bentuk matriks, tidak mengubah solusi dari sistem persamaan. Operasi ini memungkinkan matriks koefisien diubah menjadi matriks segitiga atas, sehingga solusi dari sistem persamaan dapat ditentukan dengan melakukan eliminasi variabel secara berulang. Eliminasi Gauss juga digunakan untuk menghitung peringkat matriks, determinan dari matriks persegi, dan invers dari matriks nonsingular. Metode ini dinamai dari matematikawan Carl Friedrich Gauss (1777–1855), meskipun beberapa aspek dari metode ini telah diketahui oleh matematikawan Tionghoa sejak tahun 179 M, tetapi tanpa pembuktian formal.

Matriks yang dihasilkan dari algoritma ini akan memiliki bentuk eselon baris (row echelon form, REF). Jika setiap koefisien utama (nilai bukan nol pertama di sebuah baris) matriks adalah 1, dan kolom-kolom yang mengandung koefisien utama memiliki bentuk yang mirip dengan kolom pada matriks identitas, matriks tersebut dikatakan memiliki bentuk eselon baris tereduksi (reduced row echelon form, RREF). Lanjutan dari eliminasi Gauss ke RREF terkadang disebut sebagai eliminasi Gauss-Jordan. Untuk alasan komputasi, operasi baris terkadang dihentikan sebelum matriks mencapai bentuk tereduksinya saat menyelesaikan sistem persamaan. Setelah mengatur sistem persamaan linear dalam bentuk matriks koefisien, eliminasi Gauss melakukan serangkaian operasi baris dasar untuk "menyederhanakan" baris-baris matriks. Eliminasi ini dapat dibagi menjadi dua bagian. Bagian pertama, sering disebut eliminasi maju, mereduksi sistem yang diberikan ke dalam bentuk REF. Hasil dari bagian ini dapat memberitahu apakah sistem persamaan linear tidak memiliki solusi, memiliki solusi unik, atau memiliki solusi tak hingga. Bagian kedua, terkadang disebut substitusi mundur, melanjutkan penggunaan operasi baris dasar sampai matriks berada dalam bentuk RREF, dengan kata lain, hingga solusi ditemukan.

Ada tiga jenis operasi baris dasar yang dapat dilakukan pada baris matriks:

1. Menukar posisi dua baris.
2. Mengalikan suatu baris dengan skalar bukan nol.
3. Menambahkan suatu baris dengan suatu kelipatan dari baris lain.

Operasi-operasi ini tidak mengubah kumpulan solusi. Oleh karena itu, jika tujuan seseorang adalah untuk menyelesaikan sistem persamaan linear, penggunaan operasi baris ini dapat membuat masalah menjadi lebih mudah. Operasi baris dasar juga dapat dianggap sebagai proses menghasilkan dekomposisi matriks dari matriks asli, yang ternyata sangat berguna dalam menganalisis algoritma. Operasi baris dasar dapat dilihat sebagai sebuah perkalian matriks dasar (di sebelah kiri) dengan matriks asli. Hal ini mengartikan bagian pertama dari eliminasi Gauss sebenarnya mencari dekomposisi LU; sedangkan bagian kedua menyatakan matriks asli sebagai hasil perkalian dari suatu matriks terbalikkan yang unik dengan suatu matriks eselon baris tereduksi yang unik.

Operasi dasar yang lain adalah menukar posisi dua kolom. Hal ini tidak diharuskan dalam menjalankan algoritma, namun terkadang digunakan dalam program komputer karena alasan stabilitas. Analoginya, ketika menghitung dalam kepala, kita terkadang menghindari mengalikan baris dengan bentuk pecahan yang rumit. Kekurangan dari operasi ini adalah menghasilkan komputasi tambahan dan juga mengubah nilai determinan dari matriks koefisien.

Untuk setiap baris dalam matriks yang tidak sepenuhnya terdiri dari nol, entri pertama yang bukan nol (dari kiri ke kanan) disebut sebagai koefisien utama atau pivot dari baris tersebut. Apabila terdapat dua koefisien utama yang berada di kolom yang sama, maka dapat dilakukan operasi baris tipe ketiga untuk mengubah salah satu dari koefisien tersebut menjadi nol. Operasi baris tipe pertama digunakan untuk mengatur ulang urutan baris-baris tersebut. Baris yang memiliki koefisien utama lebih dulu (yang posisinya lebih ke kiri) akan ditempatkan lebih tinggi dalam matriks koefisien daripada baris lainnya. Sementara itu, baris yang tidak memiliki koefisien utama (yaitu semua entri bernilai nol) akan ditempatkan pada posisi paling bawah dari matriks. Setelah semua operasi ini selesai, matriks tersebut dapat dikatakan telah berada dalam bentuk eselon baris. Istilah "eselon" digunakan karena secara visual bisa dibayangkan bahwa setiap baris diberi peringkat berdasarkan ukurannya, dimulai dari yang terbesar di bagian atas hingga yang terkecil di bagian bawah.

Sebagai contoh, matriks dalam bentuk eselon baris menunjukkan koefisien utamanya yang di-*highlight* dengan warna merah:

DOI: http://dx.doi.org/10.22441/fifo.2024.v16i1.002

$$\begin{bmatrix} 0 & \mathbf{2} & 1 & -1 \\ 0 & 0 & \mathbf{3} & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Terlihat baris pertama memiliki koefisien utama memiliki posisi lebih kiri (berada di kolom kedua) ketimbang baris kedua, yang koefisien utamanya berada di kolom ketiga. Selain itu, baris berisi nol terletak pada baris paling bawah.

Suatu matriks dikatakan dalam bentuk eselon baris tereduksi jika selanjutnya semua koefisien utama sama dengan 1 (yang dapat dicapai dengan menggunakan operasi baris dasar jenis ke-2), dan di setiap kolom yang berisi koefisien utama, semua entri lain di kolom itu adalah nol (yang dapat dicapai dengan menggunakan operasi baris dasar jenis ke-3). Bentuk eselon baris tereduksi dari contoh di atas adalah:

$$\begin{bmatrix} 0 & \mathbf{1} & 0 & -\frac{2}{3} \\ 0 & 0 & \mathbf{1} & \frac{1}{3} \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Misalnya kita ingin menemukan himpunan solusi dari persamaan linear berikut:

$$\begin{alignedat}{2} 2x + y - z &= 8 &\qquad (L_1) \\ -3x - y + 2z &= -11 &\qquad (L_2) \\ -2x + y + 2z &= -3 &\qquad (L_3) \end{alignedat}$$

Ketiga persamaan Tersebut dapat dinyatakan ke dalam bentuk matriks koefisien gabungan:

$$\left[\begin{array}{rrr|r} 2 & 1 & -1 & 8 \\ -3 & -1 & 2 & -11 \\ -2 & 1 & 2 & -3 \end{array}\right]$$

Prosedur operasi baris dasarnya adalah sebagai berikut:

1. Hilangkan variabel x dari semua persamaan di bawah L_1, lalu dihilangkan variabel y dari semua persamaan di bawah ini L_2.
2. Lalu langkah selanjutnya, variabel x dihilangkan dari L_2 dengan menambahkan $\frac{3}{2} L_1 + L_2$. Sedangkan x dihilangkan dari L_3 dengan menambahkan L_1 ke L_3. Langkah ini akan menghasilkan matriks koefisien dan persamaan berikut:

Matriks koefisien gabungan	Sistem persamaan
$\left[\begin{array}{rrr\|r} 2 & 1 & -1 & 8 \\ 0 & \frac{1}{2} & \frac{1}{2} & 1 \\ 0 & 2 & 1 & 5 \end{array}\right]$	$\begin{aligned} 2x + y - z &= 8 \\ \tfrac{1}{2}y + \tfrac{1}{2}z &= 1 \\ 2y + z &= 5 \end{aligned}$

3. Langkah berikutnya adalah menghilangkan variabel y dengan melakukan operasi $L_3 - 4L_2 \rightarrow L_3$. Langkah ini akan menghasilkan matriks berikut:

Matriks koefisien gabungan	Sistem persamaan
$\left[\begin{array}{ccc\|c} 2 & 1 & -1 & 8 \\ 0 & \frac{1}{2} & \frac{1}{2} & 1 \\ 0 & 0 & -1 & 1 \end{array}\right]$	$\begin{aligned} 2x + y - z &= 8 \\ \tfrac{1}{2}y + \tfrac{1}{2}z &= 1 \\ -z &= 1 \end{aligned}$

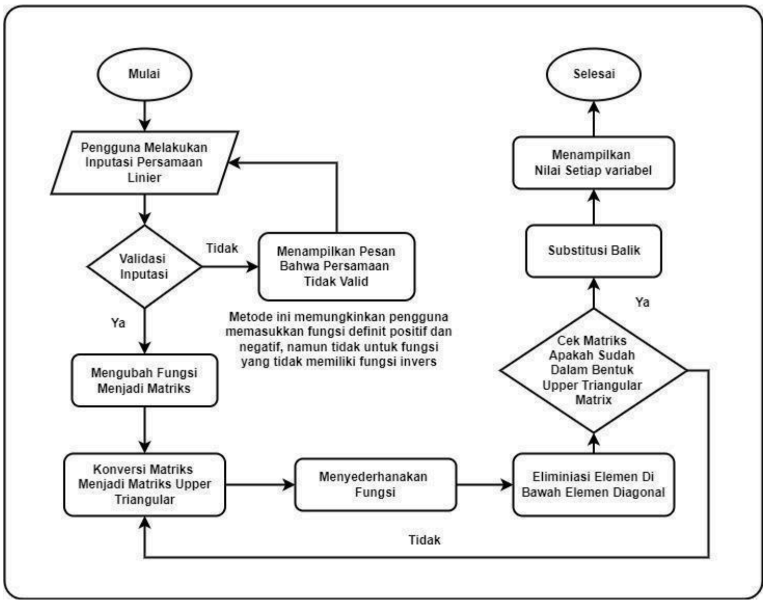
4. Selanjutnya adalah menemukan solusi dari variabel dalam urutan terbalik, atau substitusi balik. Kita sudah mempunyai $z = -1$. Lalu dengan melakukan substitusi nilai z ke persamaan $\frac{1}{2}y + \frac{1}{2}z = 1$ akan didapatkan $y = 3$. Lalu dengan dilakukan substitusi kembali ke persamaan $2x + y - z = 8$, maka didapatkan $x = 2$.

Implementasi Algoritma

Pada tahap pembangunan aplikasi untuk penyelesaian persamaan linier, kami mengimplementasikan algoritma Eliminasi Gauss dengan Substitusi Terbalik sebagai inti dari perhitungan matematis. Implementasi ini dilakukan dengan menggunakan bahasa pemrograman *TypeScript*, yang dipilih karena kemampuannya dalam menyediakan pengetikan statis dan fitur-fitur pemrograman berorientasi objek, serta *Next.js* sebagai *framework* untuk membangun antarmuka pengguna dan *server-side logic*.

1. Rancangan Algoritma

   Algoritma eliminasi gauss dengan substitusi terbalik dirancang untuk mengkonversi matriks persamaan linier menjadi bentuk eselon baris, yang selanjutnya memudahkan proses penemuan solusi dari setiap variabel. Gambar 3 menampilkan flowchart yang mengilustrasikan langkah-langkah yang diambil dalam implementasi algoritma ini:



Gambar 3. Diagram Alur Proses Penghitungan

2. Implementasi dalam *TypeScript* dan *Next.js*

   Implementasi algoritma dalam *TypeScript* memanfaatkan fitur-fitur seperti tipe data dan *interface* untuk memastikan integritas data dan memudahkan pembacaan kode. *Next.js* digunakan untuk membangun *user interface* yang responsif dan interaktif, dimana pengguna dapat memasukkan persamaan linier dan menerima solusi secara *real-time*.

Struktur Data

a. Mendeskripsikan Tipe Data Matriks

```
export type Matrix = number[][];
```

Matriks numerik yang digunakan untuk representasi sistem persamaan linear

b. Mendeskripsikan Tipe Data Solusi

```
export interface Solution {
  value: number;
  variable: string;
  steps: string[];
}
```

Menyimpan nilai solusi, variabel terkait, dan langkah-langkah yang diambil untuk mencapai solusi tersebut.

c. Mendeskripsikan Tipe Data Step dan Matriks

```
export interface Step {
  matrix: Matrix;
  step: string;
}
```

d. Gaussian Elimination Result

```
export interface GaussianEliminationResult {
  eliminationSteps: Step[];
  solution: Solution[];
  equations: string[];
  inputMatrix: Matrix;
}
```

Fungsi Utama

a. Mendefinisikan Fungsi gaussianElimination(input: string)

```
function gaussianElimination(input: string):
GaussianEliminationResult {
  const { matrix, equations } = parseInput(input);
  const inputMatrix = JSON.parse(JSON.stringify(matrix));
  const steps: Step[] = [];
  const n = matrix.length;

  // Mengonversi matriks menjadi bentuk segitiga atas
  for (let i = 0; i < n; i++) {
    ...
  }

  const solution = backSubstitution(matrix);

  return {
    eliminationSteps: steps,
```

DOI: http://dx.doi.org/10.22441/fifo.2024.v16i1.002

```
    solution: solution,
    equations,
    inputMatrix,
  };
}
```

Fungsi ini mengolah input berupa string persamaan, mengkonversinya menjadi matriks, dan menjalankan proses eliminasi Gauss serta substitusi terbalik. Langkah-langkah proses disimpan dalam *array steps*.

b. Proses Eliminasi

Kode ini mengiterasi setiap baris dari matriks untuk mengkonversi matriks ke bentuk segitiga atas:

```
for (let i = 0; i < n; i++) {
  const factor = matrix[i][i];
  for (let k = i; k < n + 1; k++) {
    matrix[i][k] /= factor;
  }
  steps.push({
    matrix: JSON.parse(JSON.stringify(matrix)),
    step: `Sederhanakan baris ${i + 1} dengan pembagi ${factor}`,
  });
  ...
}
```

c. Proses Substitusi Balik

Setelah matriks diubah menjadi bentuk segitiga atas, fungsi backSubstitution(matrix: Matrix) dijalankan:

```
export function backSubstitution(matrix: Matrix): Solution[] {
  const n = matrix.length;
  const solution: Solution[] = new Array(n).fill(0);

  for (let i = n - 1; i >= 0; i--) {
    ...
  }

  return solution;
}
```

d. Proses Parsing Input

Fungsi parseInput(input: string) bertanggung jawab untuk mengubah string input yang berisi persamaan linear menjadi matriks yang bisa diolah oleh algoritma eliminasi Gauss:

```
export function parseInput(input: string): {
  matrix: Matrix;
  equations: string[];
} {
  const equations = input.split(";").map((equation) =>
equation.trim());
  const matrix: Matrix = [];
```

DOI: http://dx.doi.org/10.22441/fifo.2024.v16i1.002

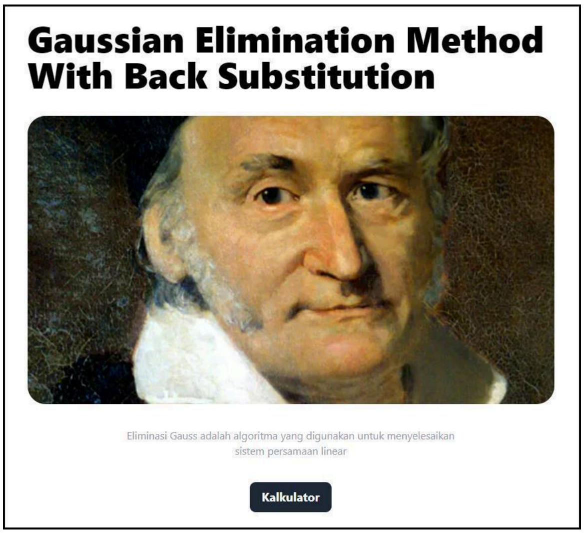
```
    for (const equation of equations) {
      ...
    }

    return {
      matrix,
      equations,
    };
```

Tampilan Aplikasi

1. Tampilan Utama Aplikasi

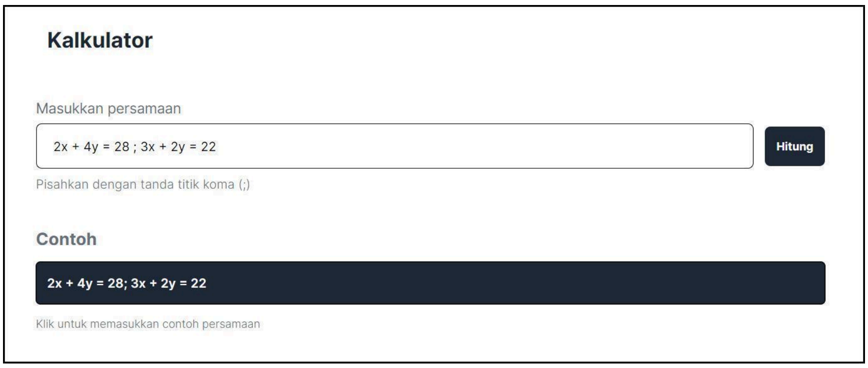
   Pada tampilan utama, pengguna disajikan dengan antarmuka, dimana terdapat penjelasan singkat mengenai prinsip dasar dan kegunaan Eliminasi Gauss. Untuk mengakses kalkulator, pengguna dapat menggulir ke bawah halaman atau mengklik tombol "Kalkulator" yang tersedia. Hal ini akan membawa mereka ke *form input* di mana persamaan linear maksimal 2 variabel dapat dimasukkan untuk dihitung.



Gambar 4. Tampilan Utama *Website*

2. Tampilan Kalkulator

   Lalu pada laman ini, pengguna hanya perlu memasukkan persamaan linear maksimal 2 variabel ke dalam kolom input yang disediakan. Format inputasi lebih dari 1 persamaan linear dipisahkan dengan tanda titik koma (;). Gambar 5 memperlihatkan tampilan kalkulator yang dirancang untuk kemudahan penggunaan, dengan petunjuk dan contoh input yang jelas untuk memandu pengguna dalam prosesnya.



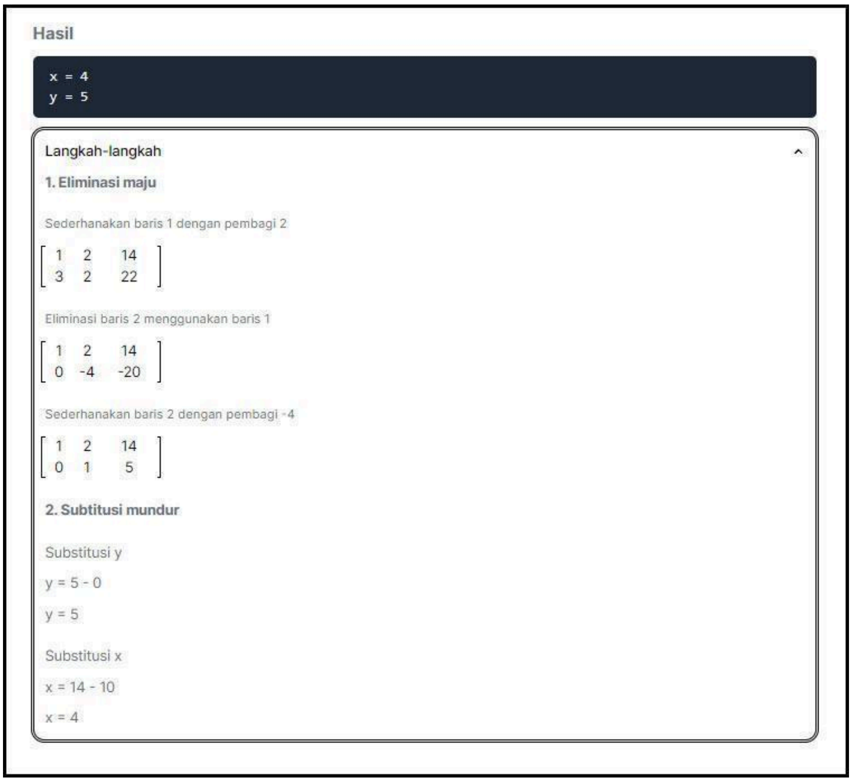
Gambar 5. Tampilan Kalkulator

DOI: http://dx.doi.org/10.22441/fifo.2024.v16i1.002

3. Tampilan Hasil Perhitungan

Setelah persamaan dimasukkan dan tombol "hitung" di-klik, aplikasi langsung memproses input dan menampilkan serangkaian langkah penyelesaian persamaan linear tersebut. Mulai dari penerapan metode eliminasi hingga substitusi terbalik, setiap tahap dipaparkan, sehingga pengguna dapat memahami proses penyelesaiannya. Gambar 6 memperlihatkan hasil akhir disajikan dengan rinci, memberikan solusi persamaan serta penjelasan langkah demi langkah yang telah dilalui.

Gambar 6. Tampilan Hasil Perhitungan



Gambar 7. Tampilan Hasil Perhitungan Lanjutan

Pengujian Unit

Pengujian unit dilakukan untuk memvalidasi setiap komponen logika penghitungan persamaan linier secara terpisah, memastikan bahwa setiap fungsi bekerja sesuai dengan harapan. Penulis menggunakan *Jest* untuk melakukan pengujian. *Jest* merupakan *framework* pengujian *open-source* yang dibuat dengan *JavaScript*. *Framework* ini dirancang khusus untuk aplikasi web yang menggunakan *React* dan *React Native*. *Jest* digunakan sebagai *framework* pengujian karena kemudahannya dalam menulis dan menjalankan tes, serta dukungannya terhadap pengujian asinkron dan *mock functions*.

1. Konfigurasi Pengujian

   Jest dikonfigurasi untuk bekerja dengan proyek TypeScript melalui Babel untuk transpalasi kode. Konfigurasi ini memungkinkan penggunaan semua fitur ES6+ dalam tes, serta integrasi mulus dengan sistem modul TypeScript.

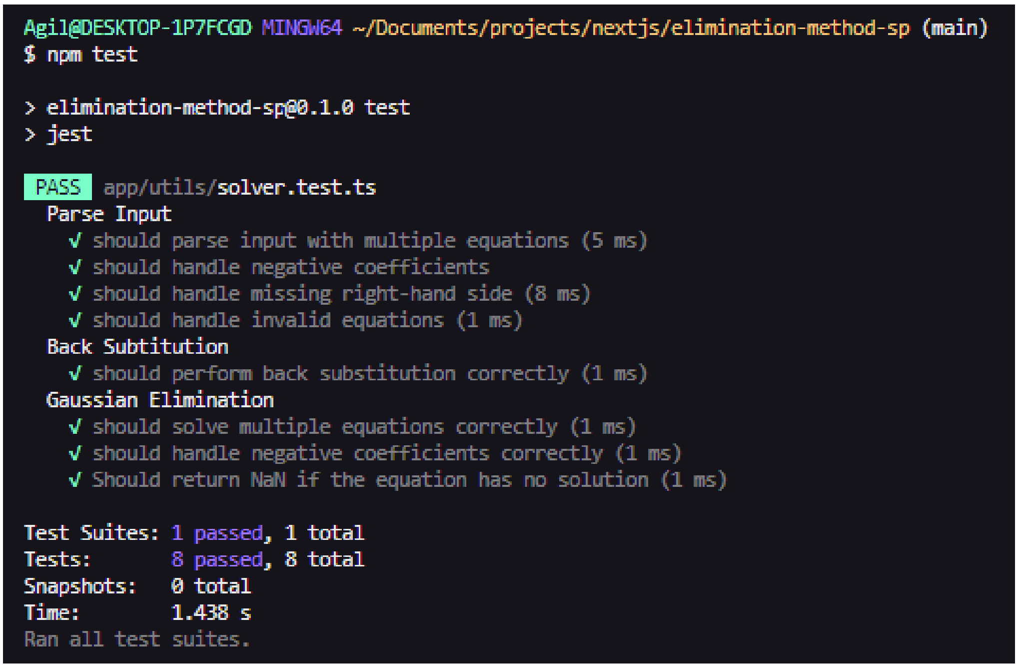
2. Struktur Pengujian

   Pengujian unit dibagi berdasarkan fungsi-fungsi utama dalam logika penghitungan persamaan linier, meliputi
   a. Pengujian fungsi eliminasi Gauss
   Memverifikasi bahwa fungsi ini menghasilkan matriks eselon yang benar dari set persamaan linier input.
   b. Pengujian fungsi substitusi terbalik
   Memastikan bahwa fungsi ini dapat menyelesaikan matriks eselon untuk menemukan nilai variabel yang tidak diketahui.
   c. Pengujian kasus *edge*
   Meliputi pengujian pada persamaan tanpa solusi atau dengan solusi tak hingga dan pada persamaan yang tidak sesuai, untuk memastikan bahwa sistem memberikan respons yang sesuai.

3. Hasil Pengujian

   Pengujian unit mengungkapkan bahwa:

   a. Fungsi eliminasi Gauss berhasil mengubah setiap set persamaan linier ke dalam bentuk matriks eselon dalam semua kasus uji.
   b. Fungsi substitusi terbalik secara akurat menemukan solusi untuk variabel yang tidak diketahui dalam semua kasus persamaan linier yang diuji.
   c. Pengujian kasus *edge* berhasil diidentifikasi dan ditangani dengan tepat, dengan sistem yang memberikan pesan kesalahan yang sesuai.

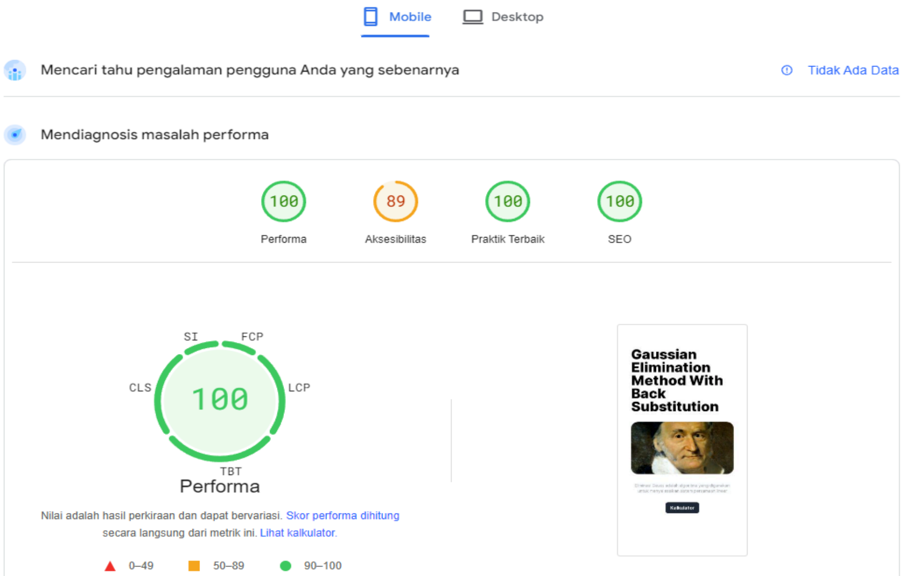


Gambar 8. Hasil Pengujian Unit

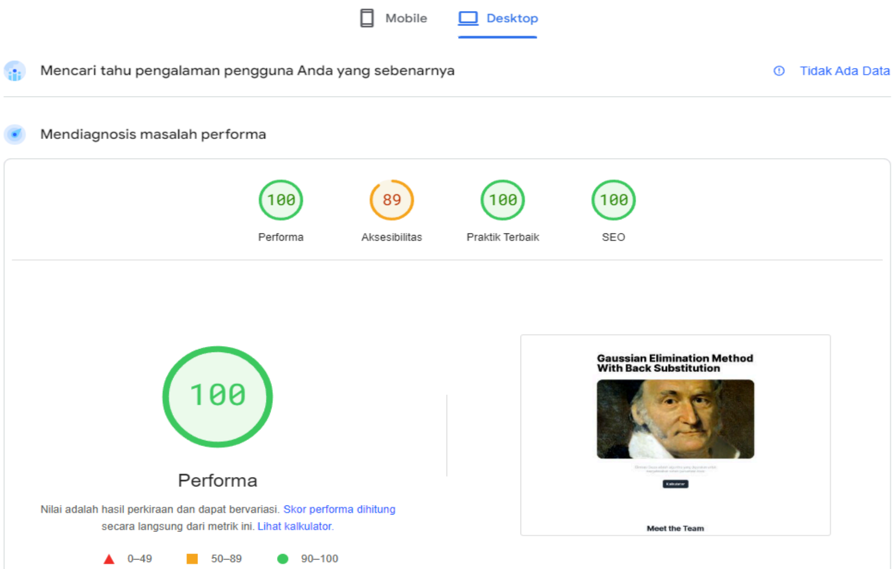
Performa

Performa aplikasi diukur menggunakan *Google PageSpeed Insights*, yang menyediakan skor berdasarkan berbagai metrik yang berkaitan dengan kecepatan pemuatan halaman dan pengalaman pengguna (*UX*). Alat ini menguji website dalam dua kondisi berbeda: untuk *desktop* dan perangkat *mobile*, mengingat perbedaan kapasitas pemrosesan dan ukuran layar.

Berdasarkan pengujian *Google PageSpeed Insights* yang ditampilkan pada gambar 9 dan gambar 10, aplikasi ini berhasil meraih skor sempurna, yaitu 100/100 untuk versi *desktop* dan 100/100 untuk versi *mobile*. Pencapaian ini menunjukkan performa luar biasa dalam hal kecepatan pemuatan dan interaktivitas, baik pada desktop maupun perangkat *mobile*.



Gambar 9. Skor performa pada versi *mobile*.



Gambar 10. Skor performa pada versi *desktop*

IV. Kesimpulan

Penelitian ini menunjukkan bahwa pengembangan aplikasi web yang mengimplementasikan Metode Eliminasi Gauss dengan Substitusi Terbalik berhasil memenuhi kebutuhan pengguna dalam memahami dan menyelesaikan persamaan linear secara interaktif dan efektif. Aplikasi ini, yang dirancang menggunakan TypeScript dan Next.js, tidak hanya meningkatkan efisiensi dalam perhitungan matematis, tetapi juga menyediakan antarmuka yang responsif dan interaktif, memudahkan pengguna dalam memasukkan persamaan dan memvisualisasikan setiap langkah penyelesaian. Dengan demikian, aplikasi ini berhasil memperkuat pemahaman konseptual pengguna tentang persamaan linear dan metode penyelesaiannya, serta memperlihatkan potensi sebagai alat pembelajaran matematika yang efektif dan inovatif.

Daftar Pustaka

[1] M. Tang, S. Cai, and V. K. N. Lau, “Remote State Estimation With Asynchronous Mission-Critical IoT Sensors,” *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 3, pp. 835–850, Mar. 2021, doi: 10.1109/JSAC.2020.3018800.
[2] F. Wang, C.-M. Fan, Q. Hua, and Y. Gu, “Localized MFS for the inverse Cauchy problems of two-dimensional Laplace and biharmonic equations,” *Appl Math Comput*, vol. 364, p. 124658, Jan. 2020, doi: 10.1016/j.amc.2019.124658.
[3] X. Qin, L. Zhang, L. Yang, and S. Cao, “Heuristics to sift extraneous factors in Dixon resultants,” *J Symb Comput*, vol. 112, pp. 105–121, Sep. 2022, doi: 10.1016/j.jsc.2022.01.003.
[4] A. Ramamoorthy, A. B. Das, and L. Tang, “Straggler-Resistant Distributed Matrix Computation via Coding Theory: Removing a Bottleneck in Large-Scale Data Processing,” *IEEE Signal Process Mag*, vol. 37, no. 3, pp. 136–145, May 2020, doi: 10.1109/MSP.2020.2974149.
[5] H. Wang, Z. Kou, J. Guo, and Z. Chen, “A semi-analytical model for the transient pressure behaviors of a multiple fractured well in a coal seam gas reservoir,” *J Pet Sci Eng*, vol. 198, p. 108159, Mar. 2021, doi: 10.1016/J.PETROL.2020.108159.
[6] S. R. Zhu, L. Z. Wu, and X. L. Song, “An improved matrix split-iteration method for analyzing underground water flow,” *Eng Comput*, Jan. 2022, doi: 10.1007/s00366-021-01551-z.
[7] N. Dhiman and M. Tamsir, “Re-modified quintic B-spline collocation method for the solution of Kuramoto–Sivashinsky type equations,” *Multidiscipline Modeling in Materials and Structures*, vol. 18, no. 3, pp. 518–533, Jun. 2022, doi: 10.1108/MMMS-06-2018-0111.
[8] R. Amin, K. Shah, M. Asif, I. Khan, and F. Ullah, “An efficient algorithm for numerical solution of fractional integro-differential equations via Haar wavelet,” *J Comput Appl Math*, vol. 381, p. 113028, Jan. 2021, doi: 10.1016/J.CAM.2020.113028.
[9] C. Baier, C. Hensel, L. Hutschenreiter, S. Junges, J. P. Katoen, and J. Klein, “Parametric Markov chains: PCTL complexity and fraction-free Gaussian elimination,” *Inf Comput*, vol. 272, p. 104504, Jun. 2020, doi: 10.1016/J.IC.2019.104504.
[10] R. Amin, K. Shah, M. Asif, and I. Khan, “A computational algorithm for the numerical solution of fractional order delay differential equations,” *Appl Math Comput*, vol. 402, p. 125863, Aug. 2021, doi: 10.1016/J.AMC.2020.125863.
[11] Zunaida Sitorus, Miftahul Jannah, Nurliana Nurliana, and Septi Nur Selase, “Comparison of Speed Completion System Linear Equations Using the Elimination Method and Using Arithmetic Sequences,” *Journal Of Education And Teaching Learning (JETL)*, vol. 5, no. 3, pp. 35–43, Sep. 2023, doi: https://doi.org/10.51178/jetl.v5i3.1527.
[12] Y. Tewari, A. Hadap, P. Soni, M. Sharma, D. Shukla, and S. Malanker, “An Overview of Applications of Gaussian Numerical Methods,” *2021 2nd International Conference on Smart Electronics and Communication (ICOSEC)*, Oct. 2021, doi: https://doi.org/10.1109/icosec51865.2021.9591706.
[13] M. Nur-E-Arefin, “A Unique approach to Solve the System of Linear Equations,” *International Journal of Innovative Technology and Interdisciplinary Sciences*, vol. 4, no. 1, pp. 623–633, Mar. 2021, doi: https://doi.org/10.15157/IJITIS.2021.4.1.623-633.