BAB IV
SISTEM PERSAMAAN
ALJABAR LINIER

4 Sistem Persamaan Aljabar Linier (SPAL)

A. Model Aljabar

Penyelesaian model matematik/persamaan pemerintah (MM/PP) dapat dilakukan secara analitis atau dengan metode numerik. Banyak permasalahan di lapangan yang tidak ada penyelesaian analitisnya karena kompleksnya permasalahan. Apabila sulit untuk diselesaikan dengan metoda analitik maka diselesaikan secara numerik. Metoda numerik merupakan suatu cabang ilmu matematika yang menggunakan sistem bilangan untuk menyelesaikan model matematik yang dapat berbentuk persamaan aljabar, persamaan diferensial biasa, persamaan diferensial parsial, persamaan integral atau persamaan integrodiferensial. Jika model matematik berbentuk persamaan diferensial atau persamaan integral, maka solusinya adalah dengan mengubah model asli menjadi model aljabar, kemudian model aljabar diubah lagi menjadi model komputer di memori komputer. Bahasa pemrograman, seperti Fortran, C++ dan Visual Basic digunakan untuk mengubah model aljabar menjadi model komputer.

Sistem persamaan aljabar linier (SPAL) atau dikenal juga sebagai “Persamaan aljabar linier serempak” merupakan salah satu model aljabar yang banyak dijumpai dalam perhitungan-perhitungan yang melibatkan solusi numerik. SPAL terdiri dari sejumlah persamaan berhingga dan sejumlah peubah yang juga berhingga. Mencari solusi suatu SPAL adalah mencari nilai-nilai peubah-peubah tersebut, sehingga memenuhi semua sistem persamaan yang ada. Terdapat tiga metode untuk mencari solusi SPAL yaitu metode langsung, metode faktorisasi dan metode iterasi.

Dalam tulisan di bab ini akan dibahas empat metode langsung yaitu:

1. Metode eliminasi Gauss
2. Metode eliminasi Gauss-Jordan
3. Metode dekomposisi LU
4. Metode matriks inversi

Kemudian dibahas tiga metode iterasi yaitu:

1. Metode Jacobi
2. Metode Gauss-Seidel
3. Metode SOR *(Successive Over Relaxation)*.

A.1. Sistem Persamaan Linier

Suatu persamaan dalam matematika merupakan sebuah ungkapan kesamaan dengan tanda "=" yang melibatkan konstanta, peubah dan operasi aritmatika (+, -, x, ÷). Dalam sebuah persamaan, komponen-komponen yang dijumlahkan atau dikurangkan disebut suku. Ungkapan di sebelah kiri tanda "=" disebut ruas kiri dan ungkapan di sebelah kanan tanda "=" disebut ruas kanan. SPAL didefinisikan sebagai suatu himpunan persamaan-persamaan aljabar yang peubah-peubahnya berpangkat tunggal (linier). Sebuah persamaan dikelompokkan menjadi persamaan linier jika suku-sukunya tidak mengandung polinom derajat tinggi atau fungsi trigonometri. Secara umum, persamaan linier dengan *N* peubah, $x_1, x_2, \ldots, x_N$ dapat ditulis dalam bentuk:

$$a_N x_N + a_{N-1} x_{N-1} + a_{N-2} x_{N-2} + \ldots + a_1 x_1 + a_0 x_0 = 0 = \sum_{i=0}^{N} a_i x_i \ . \qquad (4.1)$$

dimana $a_N, a_{N-1}, a_{N-2}, \ldots, a_1, a_0$ adalah konstanta-konstanta yang diketahui.

Bentuk SPAL yang memiliki *M* buah persamaan dan *N* peubah ditulis dengan notasi berkut :

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \ldots + a_{1N}x_N &= b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \ldots + a_{2N}x_N &= b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \ldots + a_{3N}x_N &= b_3 \\ &\ldots \\ a_{m1}x_1 + a_{m2}x_2 + a_{m3}x_3 + \ldots + a_{MN}x_N &= b_{N1} \end{aligned} \qquad (4.2)$$

Sedemikian rupa sehingga persamaan di atas dapat ditulis dalam notasi matriks berikut:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \ldots & a_{1N} \\ a_{21} & a_{22} & a_{23} & \ldots & a_{2N} \\ a_{31} & a_{32} & a_{33} & \ldots & a_{3N} \\ & \ldots & \ldots & \ldots & \\ a_{N1} & a_{N2} & a_{N3} & \ldots & a_{NN} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \\ x_N \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \\ b_N \end{bmatrix} \qquad (4.3)$$

Dalam sistem persamaan aljabar linier, pada umumnya hanya digunakan matriks bujur-sangkar sehingga secara sederhana order matriks identik dengan jumlah persamaan. SPAL di atas memiliki *N* buah peubah atau bilangan anu $(x_j, j = 1, 2, \ldots, n)$ yang dicari nilainya dan identik dengan jumlah persamaan. Koefisien-koefisien a_{ij} $(a_{11} \ \ldots \ a_{NN})$ merupakan konstanta (diketahui), demikian juga b_i $(b_1 \ldots b_N)$.

Atau dapat ditulis dengan notasi matriks: **Ax = b**. Lambang matriks koefisien selalu ditulis dalam huruf besar dan dicetak tebal, lambang vektor ditulis dalam huruf kecil tetapi dicetak tebal, sedangkan elemen-elemennya ditulis dalam huruf kecil seperti dalam penulisan matriks berikut:

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \ldots & a_{1N} \\ a_{21} & a_{22} & a_{23} & \ldots & a_{2N} \\ a_{31} & a_{32} & a_{33} & \ldots & a_{3N} \\ & \ldots & \ldots & \ldots & \\ a_{N1} & a_{N2} & a_{N3} & \ldots & a_{NN} \end{bmatrix}, \ \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \\ x_N \end{bmatrix}, \ \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \\ b_N \end{bmatrix} \qquad (4.4)$$

dengan: **A** ≡ matriks koefisien berorde N x N , **x** ≡ vektor peubah, dan **b** ≡ vektor konstanta yang dikenal sebagai vektor ruas kanan (VRK). Vektor adalah suatu himpunan obyek berupa skalar (berdimensi satu) yang kepadanya dapat dilakukan operasi-operasi skalar yang spesifik berupa 'penambahan vektor' *(vector addition)* dan 'perkalian skalar' *(scalar multiplication)*.

A.2. Determinan dan Matriks *Singular*

Jika kita menghadapi dua persamaan linear dengan dua peubah x_1 dan x_2 , sistem persamaan linier dalam dimensi 2 secara geometrik adalah dua buah garis lurus di bidang x_1-x_2. Misalnya contoh 1, cari solusi SPAL dari persamaan di bawah ini:

$$2x_1 + x_2 = 4$$
$$x_1 - x_2 = -1$$

Jawab:

$$\left.\begin{aligned} 2x_1 + x_2 &= 4 \\ x_1 - x_2 &= -1 \end{aligned}\right\} \leftrightarrow \mathbf{Ax} = \mathbf{b}, \text{ di mana } \mathbf{A} = \begin{bmatrix} 2 & 1 \\ 1 & -1 \end{bmatrix}, \ \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \ \mathbf{b} = \begin{bmatrix} 4 \\ -1 \end{bmatrix},$$

atau ditulis dengan:

$$\begin{bmatrix} 2 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 4 \\ -1 \end{bmatrix}$$

Apabila persamaan ke dua dikalikan dengan 2 dan kemudian dijumlahkan dengan persamaan pertama, kita peroleh nilai $x_1 = 1$ dan bila disubstitusikan nilai x_1 ke dalam salah satu persamaan di atas, maka kita peroleh $x_2 = 2$. Jadi solusi sistem adalah koordinat titik potong kedua garis tersebut P(1,2) yaitu di x_1 = 1 dan x_2 = 2.

Contoh 2, cari solusi SPAL dari persamaan dibawah ini:

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

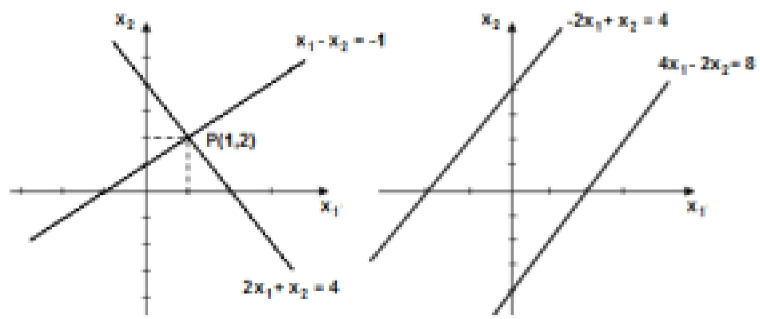
Jawab:

$$\left.\begin{aligned} -2x_1 + x_2 &= 4 \\ 4x_1 - 2x_2 &= 8 \end{aligned}\right\} \leftrightarrow \mathbf{Ax} = \mathbf{b}, \text{ di mana } \mathbf{A} = \begin{bmatrix} -2 & 1 \\ 4 & -2 \end{bmatrix}, \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} 4 \\ 8 \end{bmatrix},$$

atau ditulis dengan:

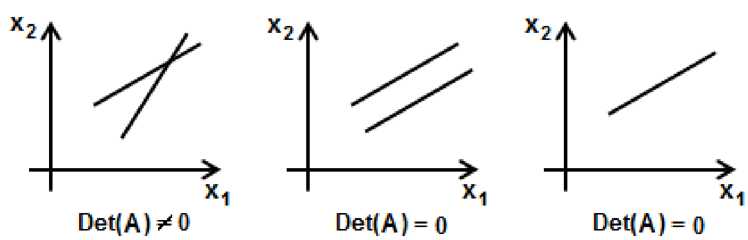
$$\begin{bmatrix} -2 & 1 \\ 4 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 4 \\ 8 \end{bmatrix}$$

Apabila persamaan pertama dikalikan dengan 2 dan kemudian dijumlahkan dengan persamaan kedua, diperoleh nilai 0 = 16. Ini berarti SPAL tersebut tidak mempunyai solusi dan bila kedua persamaan digambarkan maka merupakan dua garis sejajar yang tidak berpotongan di manapun.



Gambar 4.1 – Persamaan dua garis berpotongan dan sejajar

Menurut aljabar linear, kita perlu tahu nilai determinan dari matriks **A** yang dinyatakan oleh det(**A**). Pada contoh 1, det(**A**) = $a_{11}a_{22} - a_{12}a_{21} = -3$. Sedangkan pada contoh 2, det(**A**) = 4 – 4 = 0.



Gambar 4.2 – Solusi matriks *singular* dalam SPAL

Seperti diperlihatkan pada gambar 4.2, SPAL dimensi 2 mempunyai tiga alternatif penyelesaian:

1. Mempunyai solusi tunggal karena kedua garis lurus berpotongan di satu titik atau det(**A**) ≠ 0.
2. Tidak memiliki solusi karena kedua garis lurus sejajar atau det(**A**) = 0.
3. Memiliki banyak banyak solusi karena kedua garis lurus berimpit atau det(**A**) = 0.

Determinan A tidak boleh sama dengan nol karena jika det(**A**) = 0 maka disebut matriks *singular* dan akan mengakibatkan persamaan aljabar bersifat tidak memiliki solusi atau memiliki banyak solusi. Dua garis lurus sejajar jika mereka tidak konsisten alias bertentangan, misalnya dua sistem $2x + y = 4$ *dan* $2x + y = 5$. Dua garis akan berimpit jika kedua persamaan identik, misalnya $2x + y = 4$ dan $4x + 2y = 8$.

B. Metode Langsung

Metode langsung adalah menerapkan operasi baris elementer yaitu operasi pengubahan nilai elemen matriks berdasarkan barisnya, tanpa mengubah matriksnya. Pertama akan dibahas konsep perhitungan numeris yang berhubungan dengan metode eliminasi Gauss dan perbaikan metode Gauss dengan strategi berputar sebagian *(partial pivoting)* dan perbaikan iteratif. Kemudian dilanjutkan dengan metode eliminasi Gauss-Jordan, metode dekomposisi LU dan metode matriks inversi.

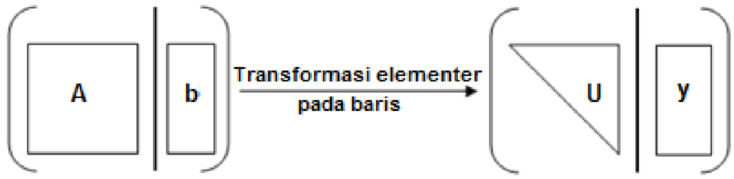
B.1. Metode Eliminasi Gauss

Metode eliminasi Gauss dilakukan dengan operasi baris elementer sehingga terbentuk **"matriks segitiga atas"**. Metode ini berangkat dari kenyataan bahwa bila matriks koefisien **A** berbentuk segitiga atas, maka solusinya dapat dihitung dengan teknik penyulihan mundur *(backward substitution)*. Meski sudah tua, tetapi metode ini dengan berbagai perbaikan masih cukup efektif dan masih digunakan sampai sekarang, khususnya untuk N relatif kecil, misalnya $N \leq 50$. Metode Gauss juga perlu untuk memahami metode lain yang lebih canggih.

Metode eliminasi Gauss pada prinsipnya bertujuan mentransformasi sistem **Ax = b** menjadi sistem **Ux = y**, dengan **U** adalah matriks segitiga atas. Dua sistem yaitu **Ax = b** dan **Ux = y** disebut ekuivalen jika solusi mereka sama. Proses eliminasi terdiri atas tiga operasi elementer:

1. Pertukaran : urutan dua persamaan dapat ditukar karena pertukaran tidak mempengaruhi solusi akhir.
2. Penskalaan : persamaan dapat dikali dengan konstanta bukan nol, karena perkalian tidak mempengaruhi solusi akhir.

3. Penggantian : persamaan dapat diganti dengan penjumlahan persamaan itu dengan penggandaan persamaan yang lain



Gambar 4.3 – Mentransformasi matriks menjadi bentuk segi-tiga atas

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} \rightarrow \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ 0 & u_{22} & u_{23} & u_{24} \\ 0 & 0 & u_{33} & u_{34} \\ 0 & 0 & 0 & u_{44} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix}.$$

Sehingga diperoleh matriks segitiga atas yang kemudian diselesaikan dengan proses substitusi mundur untuk menghitung solusi nilai x

Untuk jelasnya, perhatikan contoh sebuah SPAL dengan N = 4, sbb.:

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

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

$$-4x_1 + x_2 + 2x_3 + 2x_4 = 12$$

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

Tentukan x_1, x_2, x_3 dan x_4 dengan metode eliminasi Gauss.:
Jawab:
Buat notasi matriks dari persamaan di atas.

Ax = b

$$\begin{bmatrix} -2 & 2 & 1 & -2 \\ 2 & -3 & -2 & 3 \\ -4 & 1 & 2 & 2 \\ 3 & 2 & -2 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} -3 \\ 2 \\ 12 \\ -3 \end{bmatrix}$$

Langkah #1: Triangularisasi dimulai dengan menggunakan unsur diagonal untuk mengeliminasi semua unsur di bawahnya. Pilih nilai a_{11} sedemikian rupa yang tidak bernilai nol. Tentukan konstanta pengali baris sebagai berikut:

$$m_{i,1} = \frac{a_{i,1}^{(1)}}{a_{1,1}^{(1)}}; i = 2,3,4,...,N \tag{4.5}$$

Maka diperoleh konstanta pengali baris: $m_{21}=a_{21}/a_{11}=-1$, $m_{31}=a_{31}/a_{11}=2$, $m_{41}=a_{41}/a_{11}=-1,5$. Konstanta pengali baris digunakan untuk melakukan eliminasi suku-suku x_1 pada persamaan #2 sampai persamaan #4 sbb.:

$$\begin{aligned} a_{i,j}^{(2)} &= a_{i,j}^{(1)} - m_{i,1}.a_{1,j}^{(1)}; \quad i,j = 2,3,4,...,N \\ b_i^{(2)} &= b_i^{(1)} - m_{i,1}.b_1^{(1)}; \quad i = 2,3,4,...,N \end{aligned} \tag{4.6}$$

Maka: baris #2 baru = baris #2 lama - m_{21} x baris #1; baris #3 baru = baris #3 lama – m_{31} x baris #1; baris #4 baru = baris #4 lama – m_{41} x baris #1. Hasilnya sistem di atas berubah menjadi sistem yang ekuivalen sbb.:

$$\mathbf{A^{(1)}\,x = b^{(1)}}$$

$$\begin{bmatrix} -2 & 2 & 1 & -2 \\ 0 & -1 & -1 & 1 \\ 0 & -3 & 0 & 6 \\ 0 & 5 & -.5 & -4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} -3 \\ -1 \\ 18 \\ -7.5 \end{bmatrix} \tag{4.7}$$

Dua sistem: $\mathbf{Ax = b}$ dan $\mathbf{A^{(1)}\,x = b^{(1)}}$ disebut ekuivalen jika solusi mereka sama.

Langkah #2: Proses eliminasi dilanjutkan untuk kolom #2, kita gunakan unsur diagonal $a_{22} = -1$ dari $\mathbf{A^{(1)}}$ untuk membuat semua unsur di bawahnya = 0. Tentukan konstanta 'pengali baris' sebagai berikut:

$$m_{i,2} = \frac{a_{i,2}^{(2)}}{a_{2,2}^{(2)}}; i = 3,4,...,N \tag{4.8}$$

Kita peroleh: $m_{32} = a_{32}/a_{22} = 3$ dan $m_{42} = a_{42}/a_{22} = -5$. Konstanta pengali baris digunakan untuk melakukan eliminasi suku-suku x_2 pada persamaan #3 dan persamaan #4 sbb.:

$$\begin{aligned} a_{i,j}^{(3)} &= a_{i,j}^{(2)} - m_{i,2}.a_{1,j}^{(2)}; \quad i,j = 3,4,...,N \\ b_i^{(3)} &= b_i^{(2)} - m_{i,2}.b_1^{(2)}; \quad i = 3,4,...,N \end{aligned} \tag{4.9}$$

Maka: baris #3 baru = baris #3 lama – m_{32} x baris #2 dan baris #4 baru = baris #4 lama – m42 x baris #2. Hasilnya menjadi sistem yang ekuivalen sbb.:

$$\mathbf{A}^{(2)}\,\mathbf{x} = \mathbf{b}^{(2)}$$

$$\begin{bmatrix} -2 & 2 & 1 & -2 \\ 0 & -1 & -1 & 1 \\ 0 & 0 & 3 & 3 \\ 0 & 0 & -5.5 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} -3 \\ -1 \\ 21 \\ -12.5 \end{bmatrix}. \tag{4.10}$$

Langkah #3: Akhirnya kita menggunakan unsur diagonal $a_{33} = 3$ dari (4.10) untuk menghilangkan unsur di bawahnya. Kita hitung konstanta pengali $m_{43} = a_{43}/a_{33} = -11/6$, dan hitung: Baris #4 baru = baris #4 dari (4.10) - m_{43} x baris #3, hingga (4.10) berubah menjadi sistem yang ekuivalen sbb.:

$$\mathbf{A}^{(3)}\,\mathbf{x} = \mathbf{b}^{(3)}$$

$$\begin{bmatrix} -2 & 2 & 1 & -2 \\ 0 & -1 & -1 & 1 \\ 0 & 0 & 3 & 3 \\ 0 & 0 & 0 & 6.5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} -3 \\ -1 \\ 21 \\ 26 \end{bmatrix}. \tag{4.11}$$

Perhatikan, bahwa matriks **A** di atas sekarang telah berubah menjadi matriks segitiga atas (**U**). Inilah hasil akhir dari tahap pertama dari metode Gauss yang disebut proses eliminasi atau triangularisasi. Selanjutnya diikuti oleh tahap kedua, yaitu proses substitusi mundur, yaitu berturut-turut menghitung solusi nilai-nilai **x** harus dimulai dari indeks terbesar (x_4) sampai indeks terkecil (x_1) yaitu:

$$x_N = \frac{b_N^{(N)}}{a_{N,N}^{(N)}} \tag{4.12}$$

Kemudian diikuti dengan:

$$x_k = \frac{1}{a_{k,k}^{(k)}}\left(b_k^{(k)} - \sum_{j=k+1}^{N} a_{k,j}^{(k)}.x_j \right);\; k = N-1, N-1, \ldots, 1 \tag{4.13}$$

Maka diperoleh $x_4 = 26 / 6.5 = 4$. Substitusi hasil ini ke persamaan #3 akan memberi hasil: $x_3 = (21 - 12) / 3 = 3$, dan selanjutnya substitusi nilai x_4 dan x_3 persamaan #2 memberi hasil: $x_2 = (-1 + 3 - 4) / (-1) = 2$, dan akhirnya substitusi nilai x_2, x_3 dan x_4, maka diperoleh: $x_1 = 1$. Dengan demikian selesailah metode Gauss untuk sistem dengan N = 4, yang dapat diperluas untuk N > 4.

Karena **x** dalam proses eliminasi tidak berperan apapun, maka di tahap ini kita cukup memperhatikan gabungan matrix **A** dan vektor **b**, yaitu matrix $N \times (N+1)$:

$$\left[\begin{array}{cccc|c} -2 & 2 & 1 & -2 & -3 \\ 2 & -3 & -2 & 3 & 2 \\ -4 & 1 & 2 & 2 & 12 \\ 3 & 2 & -2 & -1 & -3 \end{array}\right]$$

dengan $N - 1$ (di sini $N - 1 = 4 - 1 = 3$) tahap-tahap triangularisasi

$$\left[\begin{array}{cccc|c} -2 & 2 & 1 & -2 & -3 \\ 0 & -1 & -1 & 1 & -1 \\ 0 & -3 & 0 & 6 & 18 \\ 0 & 5 & -5 & -4 & -7{,}5 \end{array}\right] \rightarrow \left[\begin{array}{cccc|c} -2 & 2 & 1 & -2 & -3 \\ 0 & -1 & -1 & 1 & -1 \\ 0 & 0 & 3 & 3 & 21 \\ 0 & 0 & -5{,}5 & 1 & -12{,}5 \end{array}\right] \rightarrow$$

$$\left[\begin{array}{cccc|c} -2 & 2 & 1 & -2 & -3 \\ 0 & -1 & -1 & 1 & -1 \\ 0 & 0 & 3 & 3 & 21 \\ 0 & 0 & 0 & 6{,}5 & 26 \end{array}\right]$$

Setelah proses triangularisasi dilanjutkan dengan substitusi mundur seperti uraian di atas sehingga didapatkan nilai-nilai variabel bebas x_4, x_3, x_2 dan x_1 yang dicari.

Berikut adalah contoh program Visual Basic untuk metode eliminasi Gauss.

```
Private Sub Eliminasi_Gauss_Click()
Dim a(n,n), x(n), b(n) As Double
Dim m As Double
Dim i, j, k As Integer
Const n = 3
    a(1,1) = 2: a(1,2) = 3: a(1,3) = -1: b(1) = 5
    a(2,1) = 4: a(2,2) = 4: a(2,3) = -3: b(2) = 3
    a(3,1) =-2: a(3,2) = 3: a(3,3) = -1: b(3) = 1
    For k = 1 To n-1              ‘mulai dari baris  1 sampai baris  n-1
          For i = k+1 To n        ‘eliminasi mulai dari baris k+1 sampai baris n
              m = a(i,k)/a(k,k)   ‘hitung faktor pengali
              For j = k To n      ‘eliminasi elemen dari kolom k sampai n
                  a(i,j) = a(i,j) – m*a(k,j)
              Next j
          b(i) = b(i) – m*b(k)    ‘eliminasi elemen vektor b pada baris i
          Next i
    Next k
    Sulih_Mundur (n,a(n,n), x(n), b(n))
End Sub
```

```
Private Sub SulihMundur(n,a(n,n), x(n), b(n) As Double)
Dim sigma As Double
Dim j, k As Integer
    x(n) = b(n)/a(n,n)
    For k = n-1 To n Step -1
        sigma = 0
        For j = k+1 To n
            sigma = sigma + a(k,j)*x(j)
        Next j
        x(k) = (b(k) – sigma)/a(k,k)
    Next k
End Sub
```

B.2. Perbaikan Metode Gauss

Metode eliminasi Gauss dalam beberapa kasus tidak memberikan ketelitian yang baik, bahkan seringkali memberikan hasil yang meleset, akibat dijumpai matriks *singular* (matriks yang tidak mempunyai invers) atau adanya pembagian dengan nol.

B.2.1. Metode Gauss dengan *Partial Pivoting*

Untuk menghindari pembagian dengan nol maka sebelum tiap baris dinormalkan, dilakukan penentuan nilai koefisien terbesar yang tersedia. Kemudian baris-baris tersebut dipertukarkan agar elemen terbesar merupakan elemen *pivot*. Misalnya, pada *pivot* $a_{kk} = 0$, baris ke-k tidak dapat digunakan untuk mengeliminasi elemen pada kolom di bawahnya, karena terjadinya pembagian dengan nol. Pivot yang bernilai nol harus dihindari dengan strategi *pivoting* yaitu mencari elemen pivot dengan nilai mutlak terbesar melalui perbandingan harga dari setiap elemen. Jadi, sebelum kita mengeliminasi unsur-unsur a_{ik}, $i = k+1, k+2, \cdots, N$ dengan unsur a_{kk}, lebih dulu dicari unsur pivot a_{ik} , $i = k, k+1, \cdots, N$ yang nilai mutlaknya terbesar atau dominan.

Jika misalnya $|a_{ibig,k}|$, $ibig \geq k$, adalah unsur yang dominan di antara $|a_{ik}|$, $i = k, k+1, \cdots, N$, maka baris # k dan baris # $ibig$ ditukar, termasuk menukar ruas kanan b_{ibig} dengan b_k. Jika kebetulan $ibig = k$, penukaran baris ke k dan baris ke $ibig$ tidak perlu dilakukan. Dengan demikian unsur a_{kk} sekarang dominan, yaitu mempunyai nilai mutlak terbesar antara a_{ik}, $i = k, k+1, \cdots, N$, dan eliminasi unsur ke k+1 dst. Hal di atas dilakukan untuk $k = 1, 2, \cdots, N - 1$.

Metode Gauss dengan partial *pivoting* memberi manfaat sbb.:

a. Meningkatkan keakuratan hasil solusi yang diperoleh;
b. Dapat mendeteksi jika det(**A**) = 0, yang akan tampak dari *BIG* = 0 atau $a_{NN} = 0$.

Berikut adalah contoh program Visual Basic untuk metode eliminasi Gauss dengan *pivoting*.

```
Private Sub Eliminasi_Gauss_Pivot_Click()
Dim a(n,n), x(n), b(n), pivot As Double
Dim m As Double
Dim i, j, k, r, t, s As Integer
Const n = 4
  a(1,1) = 1.1348: a(1,2) = 3.8326: a(1,3) = 1.1651: a(1,4) = 3.4017: b(1) = 0.91465
  a(2,1) = 0.5301: a(2,2) = 1.7875: a(2,3) = 2.5330: a(2,4) = 1.5435: b(2) = -1.66639
  a(3,1) = 3.4129: a(3,2) = 4.9317: a(3,3) = 8.7643: a(3,4) =  1.3142: b(3) = 2.07196
  a(4,1) = 1.2371: a(4,2) = 4.9998: a(4,3)=10.6721: a(4,4) =  0.0147: b(4) = 0 .58872

  k=1
  singular = false
  While (k<=n-1) And (not singular)
    pivo t= a(k,k)
    r = k
    For  t = k+1 To n              ‘cari elemen pivot dengan nilai mutlak terbesar
      If Abs(a(t,k)) > Abs(pivot) Then
          pivot = a(k,k)
          r = k
      End If
    Next t
    If pivot = 0 Then
      singular = true
      ElseIf  r > k Then
        For s = 1 To n             ‘pertukarkan baris k dengan baris r
          temp = a(k,s)
          a(k,s) = a(r,s)
          a(r,s) = temp
        Next s
        temp = b(k)
        b(k) = b(r)
        b(r) = temp
        For  i = k+1 To n          ‘eliminasi dari baris k+1 sampai baris ke n
          m = a(i,k)/a(k,k)        ‘hitung faktor pengali
          For j = k To n           ‘eliminasi elemen dari kolom k sampai n
            a(i,j) = a(i,j) – m*a(k,j)
          Next j
          b(i) = b(i) – m*b(k)     ‘eliminasi elemen vektor b pada baris i
```

```
        Next i
    End If
    k = k + 1
  Wend
  If not singular Then
    Sulih_Mundur (n,a(n,n), x(n), b(n))
  Else
    For i = 1 To n                  'solusi tidak ada, tetapi vektor x harus tetap diisi
      x(i) =-9999
    Next i
End Sub
```

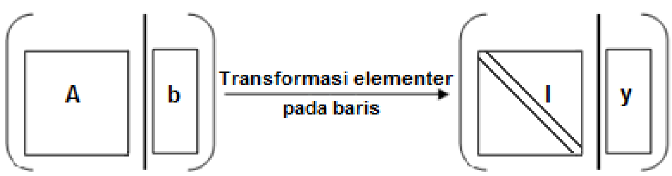
B.2.2. Metode Gauss dengan Perbaikan Iteratif

Dalam masalah **Ax** = **b**, **A** dan **b** diketahui dan **x** dicari. Setelah menerapkan metode Gauss, didapat solusi **x′** yang agak meleset dari **x** yang benar, antara lain karena keterbatasan ketelitian komputer. Karena **x**′ ≠ **x**, maka **b**′ = **Ax**′ ≠ **Ax** = **b** sehingga diperoleh **A**(**x** – **x**′) = **b** – **b**′. Jika diandaikan vektor **y** adalah solusi dari persamaan. ini, jadi **Ay** = **b** – **b**′ maka **A**(**x**′ + **y**) = **Ax**′ + **Ay** = **b**′ + **b** – **b**′ = **b** atau vektor **x**′ + **y** adalah hasil solusi yang lebih baik daripada **x**′, alias **y** adalah vektor koreksi, dan diharap **x**′ + **y** makin mendekati vektor solusi **x** yang sebenarnya.

Untuk menghindari adanya kesalahan akibat keterbatasan ketelitian komputer, maka untuk mendapatkan hasil yang lebih baik, sebaiknya gunakan peubah presisi ganda *(double precision)*.

B.3. Metode Eliminasi Gauss-Jordan

Metode Gauss-Jordan merupakan variasi dari metode eliminasi Gauss di mana matriks **A** ditransformasi menjadi matriks identitas **I** sehingga tidak diperlukan lagi teknik substitusi mundur untuk memperoleh solusi persamaan linier. Solusinya langsung diperoleh dari vektor kolom **B** hasil proses eliminasi.



Gambar 4.4 – Mentransformasi matriks menjadi matriks Identitas

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} \leftrightarrow \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix}. \qquad (4.14)$$

Tahap #1 dilakukan eliminasi Gauss biasa sehingga dihasilkan matriks segitiga atas. Tahap #2, semua unsur a_{ii} dibuat sama dengan 1 dengan membagi baris # i, i = 1, $\cdots$, N, dengan a_{ii} , termasuk ruas kanan b_i. Tahap #3, unsur a_{ij} , $i \neq j$, di kolom # j, j = N, N–1, $\cdots$, 2, dibuat sama dengan nol, dengan jalan: (baris # i) - ($a_{ij} \times$ baris # j), termasuk ruas kanan.

Berikut adalah contoh di atas yang dicari solusinya dengan metode Gauss-Jordan.

$$\text{Tahap \#1: } \begin{bmatrix} -2 & 2 & 1 & -2 \\ 0 & -1 & -1 & 1 \\ 0 & 0 & 3 & 3 \\ 0 & 0 & 0 & 6.5 \end{bmatrix} \begin{bmatrix} -3 \\ -1 \\ 21 \\ 26 \end{bmatrix} \Rightarrow \text{ Tahap \#2: } \begin{bmatrix} 1 & -1 & -.5 & 1 \\ 0 & 1 & 1 & -1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1.5 \\ 1 \\ 7 \\ 4 \end{bmatrix}.$$

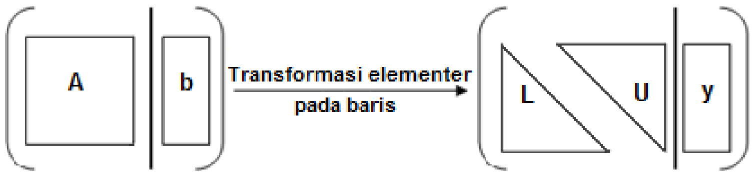
Selanjutnya unsur a_{44} digunakan untuk membuat semua unsur di atasnya sama dengan nol, dan demikian juga dengan unsur a_{33} dan a_{22} , sehingga berturut-turut didapat hasil sbb.:

$$\begin{bmatrix} 1 & -1 & -.5 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -2.5 \\ 5 \\ 3 \\ 4 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 \\ 2 \\ 3 \\ 4 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}.$$

Dari ruas kanan persamaan matriks terakhir langsung didapat solusi: $x_1 = y_1 = 1$; $x_2 = y_2 = 2$; $x_3 = y_3 = 3$, dan $x_4 = y_4 = 4$, sama dengan hasil dari metode Gauss di atas.

B.4. Metode Dekomposisi LU

Metode dekomposisi LU merupakan metode yang paling populer untuk memecahkan SPAL. Jika matriks **A** *non-singular* (matriks yang mempunyai inversi) maka dapat didekomposisi (diuraikan atau difaktorkan) menjadi produk dua matriks yaitu matriks segitiga bawah **L** *(lower)* dan matriks segitiga atas **U** *(upper)* dengan cara melakukan operasi baris elementer. Sistem **Ax = b** ditransformasi menjadi menjadi **LUx = b** dengan matriks segitiga bawah **L**, semua elemen diagonal utamanya adalah 1, sedangkan pada matriks segitiga atas **U** tidak ada aturan khusus.



Gambar 4.4 – Mentransformasi matriks menjadi matriks segitiga atas dan matrks segitiga bawah

Sehingga dapat ditulis sbb.:

A = LU

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ l_{21} & 1 & 0 & 0 \\ l_{31} & l_{32} & 1 & 0 \\ l_{41} & l_{42} & l_{43} & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ 0 & u_{22} & u_{23} & u_{24} \\ 0 & 0 & u_{33} & u_{34} \\ 0 & 0 & 0 & u_{44} \end{bmatrix} \tag{4.15}$$

Jika diandaikan **Ux = y**, maka **Ly = b** sehingga **Ax = b** dapat dicari solusinya lewat dua tahap, pertama menghitung **y** dari **Ly = b** dengan substitusi maju (karena **L** matrix segitiga bawah):

Ly = b

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ l_{21} & 1 & 0 & 0 \\ l_{31} & l_{32} & 1 & 0 \\ l_{41} & l_{42} & l_{43} & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} \tag{4.16}$$

Dengan modal harga **y** yang diperoleh kemudian menghitung **x** dari **Ux = y** dengan substitusi mundur (karena **U** matriks segitiga atas).

Ux = y

$$\begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ 0 & u_{22} & u_{23} & u_{24} \\ 0 & 0 & u_{33} & u_{34} \\ 0 & 0 & 0 & u_{44} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} \tag{4.17}$$

Terdapat 2 (dua) metode untuk mengurai matriks koefisien **A** menjadi matriks segitiga bawah **L** dan matriks segitiga atas **U** yaitu:

1. Metode LU Gauss;
2. Metode reduksi Crout.

B.4.1. Metode LU Gauss

Metode LU Gauss adalah mengurai matriks koefisien **A** menjadi matriks **L** dan **U** sebagai berikut:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ m_{21} & 1 & 0 & 0 \\ m_{31} & m_{32} & 1 & 0 \\ m_{41} & m_{42} & m_{43} & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ 0 & u_{22} & u_{23} & u_{24} \\ 0 & 0 & u_{33} & u_{34} \\ 0 & 0 & 0 & u_{44} \end{bmatrix} \qquad (4.18)$$

dimana: m_{ij} adalah faktor pengali pada proses eliminasi Gauss

Langkah-langkah pembentukan **L** dan **U** dari matriks A adalah:

Tahap #1: Nyatakan matriks koefisien **A** sebagai perkalian dengan matriks identitas **I**:

$$\mathbf{A} = \mathbf{IA}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \qquad (4.19)$$

Tahap #2: Eliminasi matriks A di ruas kanan menjadi matriks segita atas **U**.

Tahap #3: Setelah seluruh proses eliminasi Gauss selesai, matriks **I** menjadi matriks **L**, dan matriks **A** di ruas kanan menjadi matriks U

Soal: Dekomposisi matriks **A** berikut dengan metode LU Gauss

$$\mathbf{A} = \begin{bmatrix} 4 & 3 & -1 \\ -2 & -4 & 5 \\ 1 & 2 & 6 \end{bmatrix}$$

Jawab

1. Nyatakan **A** sebagai **A** = **IA**

$$\begin{bmatrix} 4 & 3 & -1 \\ -2 & -4 & 5 \\ 1 & 2 & 6 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 & 3 & -1 \\ -2 & -4 & 5 \\ 1 & 2 & 6 \end{bmatrix}$$

2. Eliminasi matriks **A** di ruas kanan menjadi matriks segitiga atas **U**, dan tempatkan faktor pengali m_{ij} di matriks **I**.

$$\begin{bmatrix} 4 & 3 & -1 \\ -2 & -4 & 5 \\ 1 & 2 & 6 \end{bmatrix} \begin{matrix} \\ R_2 - (-\frac{2}{4})R_1 \\ R_3 - \ (\frac{1}{4})R_1 \end{matrix} \begin{bmatrix} 4 & 3 & -1 \\ 0 & -2{,}5 & 4{,}5 \\ 0 & 1{,}25 & 5{,}25 \end{bmatrix}$$

3. Tempatkan m_{21} = -2/4 = -0,5 dan m_{31} = 1/4 = 0,25 ke matriks **L**

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 \\ -0{,}5 & 1 & 0 \\ 0{,}25 & 0 & 1 \end{bmatrix}$$

4. Teruskan proses eliminasi Gauss pada matriks **A**.

$$\begin{bmatrix} 4 & 3 & -1 \\ 0 & -2{,}5 & 4{,}5 \\ 0 & 1{,}25 & 5{,}25 \end{bmatrix} \begin{matrix} \\ \\ R_3 - (\frac{1{,}25}{-2{,}5})R2 \end{matrix} \begin{bmatrix} 4 & 3 & -1 \\ 0 & -2{,}5 & 4{,}5 \\ 0 & 0 & 8{,}5 \end{bmatrix} = \mathbf{U}$$

5. Tempatkan m_{32}=1,25/(-2,5)=-0,5 ke dalam matriks **L**.

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 \\ -0{,}5 & 1 & 0 \\ 0{,}25 & -0{,}5 & 1 \end{bmatrix}$$

6. Akhirnya diperoleh penguraian **A = LU**.

$$\begin{bmatrix} 4 & 3 & -1 \\ -2 & -4 & 5 \\ 1 & 2 & 6 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -0{,}5 & 1 & 0 \\ 0{,}25 & -0{,}5 & 1 \end{bmatrix} \begin{bmatrix} 4 & 3 & -1 \\ 0 & -2{,}5 & 4{,}5 \\ 0 & 0 & 8{,}5 \end{bmatrix}$$

B.4.2. Metode reduksi Crout

Tinjau matriks **A**, matriks **L** dan matriks **U** berikut:

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad \mathbf{L} = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix} \quad \mathbf{U} = \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix}$$

Karena **A** = **LU** , maka hasil perkalian **L** dan **U** dapat ditulis

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \mathbf{LU} = \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ l_{21}u_{11} & l_{21}u_{12} + u_{22} & l_{21}u_{23} \\ l_{31}u_{13} & l_{31}u_{12} + l_{32}u_{22} & l_{31}u_{13} + l_{32}u_{23} + u_{33} \end{bmatrix} \tag{4.20}$$

Dari persamaan dua buah matriks **A** = **LU** , diperoleh:

$$u_{11} = a_{11}\,, \quad u_{12} = a_{12}\,, \quad u_{13} = a_{13} \quad \}\,\text{Baris pertama } \mathbf{U} \tag{4.21}$$

$$\left.\begin{array}{l} l_{21}u_{11} = a_{21} \quad \rightarrow \quad l_{21} = \dfrac{a_{21}}{u_{11}} \\ l_{31}u_{11} = a_{31} \quad \rightarrow \quad l_{31} = \dfrac{a_{31}}{u_{11}} \end{array}\right\}\text{Kolom pertama } \mathbf{L} \tag{4.22}$$

$$\left.\begin{array}{l} l_{21}u_{12} + u_{22} = a_{22} \rightarrow u_{22} = a_{22} - l_{21}u_{12} \\ l_{21}u_{13} + u_{23} = a_{23} \rightarrow u_{23} = a_{23} - l_{21}u_{13} \end{array}\right\}\text{Baris kedua } \mathbf{U} \tag{4.23}$$

$$\left. l_{31}u_{12} + l_{32}u_{22} = a_{32} \rightarrow \; l_{32} = \frac{a_{32} - l_{31}u_{12}}{u_{22}} \right\}\text{Kolom kedua } \mathbf{L} \tag{4.24}$$

$$\left. l_{31}u_{12} + l_{32}u_{22} = a_{32} \rightarrow \; l_{32} = \frac{a_{32} - l_{31}u_{12}}{u_{22}} \right\}\text{Kolom kedua } \mathbf{L} \tag{4.25}$$

Secara umum, jika kita mempunyai matrix $\mathbf{A}_{\mathbf{NN}}$, rumus-rumus di atas dapat ditulis dengan ringkas sbb.:

$$u_{kj} = a_{kj} - \sum_{i=1}^{k-1} l_{ki}u_{ij} \quad ; k = 1, 2, \ldots , N ; j = k, k+1, \ldots , N-1 \tag{4.26}$$

$$l_{ik} = \frac{1}{u_{kk}}\left(a_{ik} - \sum_{j=1}^{k-1} l_{ij}u_{jk}\right) ; k = 1, 2, \ldots , N-1 ; i = k+1, \cdots, N. \tag{4.27}$$

Catatan: Untuk tahap pertama, yaitu k = 1, nilai = 0 di kedua rumus. Untuk menghemat memori, hasil u_{kj} disimpan di a_{kj} dan hasil l_{ik} disimpan di a_{ik} , sedangkan diagonal matriks L yang semua sama dengan 1, tidak perlu disimpan. Jadi matriks **L** dan **U** tidak perlu diadakan secara khusus.

Contoh: Hitung solusi SPAL berikut dengan metode dekomposisi LU menggunakan reduksi Crout.

$$
\begin{aligned}
x_1 + x_2 - x_3 &= 1 \\
2x_1 + 2x_2 + x_3 &= 5 \\
-x_1 + x_2 + x_3 &= 1
\end{aligned}
$$

Jawab

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & -1 \\ 2 & 2 & 1 \\ -1 & 1 & 1 \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} 1 \\ 5 \\ 1 \end{bmatrix}$$

Diperoleh:

7. Baris pertama **U**.

$$u_{11} = a_{11} = 1, \quad u_{12} = a_{12} = 1, \quad u_{13} = a_{13} = -1$$

8. Kolom pertama **L**

$$l_{21} = \frac{a_{21}}{u_{11}} = \frac{2}{1} = 2$$
$$l_{31} = \frac{a_{31}}{u_{11}} = \frac{-1}{1} = -1$$

9. Baris kedua **U**.

$$u_{22} = a_{22} - l_{21}u_{12} = 2 - 2.1 = 0$$

Karena u_{kk} tidak boleh nol, lakukan pertukaran baris:

$$\begin{array}{cc} \text{A} & \text{B} \\ R_2 \Leftrightarrow R_3 \begin{bmatrix} 1 & 1 & -1 \\ -1 & 1 & 1 \\ 2 & 2 & 1 \end{bmatrix} & R_2 \Leftrightarrow R_3 \begin{bmatrix} 1 \\ 1 \\ 5 \end{bmatrix} \end{array}$$

10. Hitung kembali nilai l_{21}, l_{31} dan u_{22} (nilai u_{11}, u_{12}, u_{13} tidak berubah).

$$l_{21} = \frac{a_{21}}{u_{11}} = \frac{-1}{1} = -1 \qquad u_{22} = a_{22} - l_{21}u_{12} = 1-(-1)(1) = 2$$

$$u_{23} = a_{23} - l_{21}u_{13} = 1-(-1)(-1) = 0$$

$$l_{31} = \frac{a_{31}}{u_{11}} = \frac{2}{1} = 2 \qquad l_{32} = \frac{a_{32} - l_{31}u_{12}}{u_{22}} = \frac{2-2(1)}{2} = 0$$

Maka diperoleh **L** dan **U** sebagai berikut:

$$\mathbf{U} = \begin{bmatrix} 1 & 1 & -1 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix} \qquad \mathbf{L} = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 2 & 0 & 1 \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} 1 \\ 1 \\ 5 \end{bmatrix}$$

Berturut-turut **y** dan **x** dihitung sebagai berikut:

$$\mathbf{Ly} = \mathbf{b} \quad \rightarrow \quad \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 2 & 0 & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 5 \end{bmatrix}$$

Dengan teknik penyulihan maju y_1, y_2 dan y_3 dapat dihitung.

$$\begin{aligned} &y_1 &&= 1 \\ &-y_1 + y_2 &&= 1 \rightarrow y_2 = 1 + y_1 = 1+1 = 2 \\ &2y_1 + 0y_2 + y_3 &&= 5 \rightarrow y_3 = 5 - 2y_1 = 3 \end{aligned}$$

Maka diperoleh:

$$\mathbf{Ux} = \mathbf{y} \quad \rightarrow \quad \begin{bmatrix} 1 & 1 & -1 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

Dengan teknik penyulihan mundur x_1, x_2 dan x_3 dapat dihitung.

$$\begin{aligned} &3x_3 &&= 3 \rightarrow x_3 = 1 \\ &2x_2 + 0x3 &&= 2 \rightarrow x_2 = 1 \\ &x_1 + x_2 - x_3 &&= 1 \rightarrow x1 = 1 - x_2 + x_3 = 1 \end{aligned}$$

Jadi solusi dari SPAL diatas adalah:

x = $[1, 1, 1]^T$

B.5. Metode Matriks Inversi

Misalnya $\mathbf{A}^{-1}$ adalah matriks inversi dari **A**. Hasil kali **A** dengan $\mathbf{A}^{-1}$ menghasilkan matriks identitas **I**.

$$\mathbf{AA}^{-1} = \mathbf{A}^{-1}\mathbf{A} = \mathbf{I} \qquad (4.28)$$

Bila matriks **A** dikalikan dengan **I** akan menghasilkan matriks **A** sendiri.

$$\mathbf{AI} = \mathbf{IA} = \mathbf{A} \qquad (4.29)$$

Berdasarkan persamaan (4.28) dan (4.29), SPAL **Ax = b** dapat diselesaikan sebagai berikut:

$$\mathbf{Ax} = \mathbf{b}$$

Kalikan ruas kiri dan ruas kanan dengan $\mathbf{A}^{-1}$.

$$\mathbf{A}^{-1}\mathbf{Ax} = \mathbf{A}^{-1}\mathbf{b} \qquad (4.20)$$

Maka diperoleh:

$$\mathbf{Ix} = \mathbf{A}^{-1}\mathbf{b} \qquad (4.21)$$

atau

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{b} \qquad (4.22)$$

Jadi penyelesaian SPAL **Ax = b** adalah $\mathbf{x} = \mathbf{A}^{-1}\mathbf{B}$ denyan syarat matriks **A** *non-singular* (matriks yang mempunyai inversi) yaitu det(**A**) ≠ 0.

Matriks inversi dapat diperoleh dengan metode eliminasi Gauss-Jordan, yaitu:

$$[\,\mathbf{A} \mid \mathbf{I}\,] \rightarrow [\,\mathbf{I} \mid \mathbf{A}^{-1}\,] \qquad (4.23)$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \\ p_{41} & p_{42} & p_{43} & p_{44} \end{bmatrix}$$

Penyelesaian dengan metode matriks inversi membutuhkan lebih banyak waktu komputasi. Metode matriks inversi baru bermanfaat bila digunakan untuk penyelesaian sejumlah SPAL dengan matriks koefisien **A** yang sama tetapi dengan beberapa vektor **b** yang berbeda-beda.

Contoh soal: Hitung matriks inversi dari matriks koefisien **A** berikut:.

$$A = \begin{bmatrix} 1{,}1 & -3{,}8 & 1{,}6 & 3{,}4 \\ 5{,}3 & 1{,}7 & 2{,}5 & -1{,}5 \\ -3{,}4 & 4{,}9 & 8{,}1 & 1{,}3 \\ 1{,}2 & 2{,}8 & -1{,}6 & 1{,}2 \end{bmatrix}$$

Jawab

Dari persamaan (4.23:) $\mathbf{AI = IA^{-1}}$ diperoleh:

$$\mathbf{A \mid I} = \left[\begin{array}{cccc|cccc} 1{,}1 & -3{,}8 & 1{,}6 & 3{,}4 & 1 & 0 & 0 & 0 \\ 5{,}3 & 1{,}7 & 2{,}5 & -1{,}5 & 0 & 1 & 0 & 0 \\ -3{,}4 & 4{,}9 & 8{,}1 & 1{,}3 & 0 & 0 & 1 & 0 \\ 1{,}2 & 2{,}8 & -1{,}6 & 1{,}2 & 0 & 0 & 0 & 1 \end{array}\right] \rightarrow$$

$$\left[\begin{array}{cccc|cccc} 1 & -3{,}454545 & 1{,}454545 & 3{,}090909 & 0{,}9090909 & 0 & 0 & 0 \\ 0 & 20{,}0090909 & -5{,}2090909 & -17{,}881818 & -4{,}8181818 & 1 & 0 & 0 \\ 0 & -6{,}8454545 & 13{,}0454545 & 11{,}8090909 & 3{,}090909 & 0 & 1 & 0 \\ 0 & 6{,}9454545 & -3{,}3454545 & -2{,}5090909 & -1{,}090909 & 0 & 0 & 1 \end{array}\right] \rightarrow$$

$$\left[\begin{array}{cccc|cccc} 1 & 0 & 0{,}555202 & 0{,}00363471 & 0{,}07723762 & 0{,}172648 & 0 & 0 \\ 0 & 1 & -0{,}260336 & -0{,}893684 & -0{,}240799 & 0{,}0499772 & 0 & 0 \\ 0 & 0 & 11{,}26333 & 5{,}691413 & 1{,}442526 & 0{,}3421172 & 1 & 0 \\ 0 & 0 & -1{,}537301 & 3{,}697955 & 0{,}5815538 & -0{,}3471149 & 0 & 1 \end{array}\right] \rightarrow$$

$$\left[\begin{array}{cccc|cccc} 1 & 0 & 0 & -0{,}2769114 & 0{,}006131356 & 0{,}1557848 & -0{,}04929287 & 0 \\ 0 & 1 & 0 & -0{,}7621356 & -0{,}2074576 & 0{,}05788484 & 0{,}0231136 & 0 \\ 0 & 0 & 1 & 0{,}5053044 & 0{,}1280727 & 0{,}030374416 & 0{,}08878365 & 0 \\ 0 & 0 & 0 & 4{,}4747606 & 0{,}7784402 & -0{,}3004203 & 0{,}1364872 & 1 \end{array}\right] \rightarrow$$

$$\left[\begin{array}{cccc|cccc} 1 & 0 & 0 & 0 & 0{,}0543035 & 0{,}1371939 & -0{,}040846 & 0{,}061883 \\ 0 & 1 & 0 & 0 & -0{,}0748747 & 0{,}00671763 & 0{,}0463599 & 0{,}1703187 \\ 0 & 0 & 1 & 0 & 0{,}04016875 & 0{,}0642988 & 0{,}07337107 & -0{,}112923 \\ 0 & 0 & 0 & 1 & 0{,}1739624 & -0{,}067136 & 0{,}03050157 & 0{,}2234756 \end{array}\right] = \mathbf{I \mid A^{-1}}$$

Dari contoh di atas tampak bahwa di setiap tahap, dari matriks lengkap 4×8, yang penuh selalu hanya 4×4, dan matrix identiti **I** yang semula di kanan, secara bertahap tergeser ke kiri. Fakta ini dapat digunakan sebagai strategi untuk menghitung $\mathbf{A}^{-1}$ tanpa harus menyediakan memori khusus untuk menyimpan $\mathbf{A}^{-1}$, suatu hal yang menguntungkan jika kita harus menghitung inversi matriks besar dengan keterbatasan memori komputer

C. Metode Iteratif

Metode langsung seperti metode eliminasi Gauss maupun metode eliminasi Gauss-Jordan banyak melibatkan ralat pembulatan yang dapat menyebabkan solusi yang diperoleh jauh dari solusi sebenarnya. Dengan metode iteratif, ralat pembulatan dapat diperkecil, karena iterasi dapat dilakukan sampai batas ralat sekecil mungkin yang kita perbolehkan. Artinya besar ralat dapat dikendalikan sampai batas yang bisa diterima.

Andaikan kita mempunyai sebuah matrix $\mathbf{A}_{4x4}$ dengan sifat: $a_{ii} \neq 0 \ \forall i$, maka sistem **Ax = b** baris demi baris dapat dibawa ke bentuk:

$$
\begin{aligned}
a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + a_{14}x_4 = b_1 &\rightarrow x_1 = \left[b_1 - \left(a_{12}x_2 + a_{13}x_3 + a_{14}x_4\right)\right]/a_{11} \\
a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + a_{24}x_4 = b_2 &\rightarrow x_2 = \left[b_2 - \left(a_{21}x_1 + a_{23}x_3 + a_{24}x_4\right)\right]/a_{22} \\
a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + a_{34}x_4 = b_3 &\rightarrow x_3 = \left[b_3 - \left(a_{31}x_1 + a_{32}x_2 + a_{34}x_4\right)\right]/a_{33} \\
a_{41}x_1 + a_{42}x_2 + a_{43}x_3 + a_{44}x_4 = b_4 &\rightarrow x_4 = \left[b_4 - \left(a_{41}x_1 + a_{42}x_2 + a_{43}x_3\right)\right]/a_{44}
\end{aligned}
\tag{4.24}
$$

Rumus ini dapat digunakan sebagai dasar untuk mencari solusi sistem **Ax = b** secara iteratif. Yang dimaksud dengan "metode solusi iteratif" ialah: Bertolak dari nilai awal x_i, i = 1, ⋯, 4, sebagai solusi sembarang yang salah, kemudian solusi yang salah ini secara bertahap dikoreksi sehingga diperoleh solusi yang makin mendekati solusi sebenarnya. Solusi awal yang dipilih sembarang adalah "informasi dugaan" *(a priori information)*.

Sebagai kondisi berhentinya iterasi, dapat digunakan pendekatan kesalahan mutlak atau kesalahan relatif sebagai syarat yang harus dipenuhi, misalnya:

$$\left|\frac{x_i^{(k+1)} - x_i^{(k)}}{x_i^{(k+1)}}\right| < \varepsilon \text{ , untuk: i = 1, ... , N} \tag{4.25}$$

di mana ε > 0 adalah bilangan kecil yang kita tentukan sebelumnya.

Syarat agar iterasi dapat konvergen adalah sistem dominan secara diagonal:

$$|a_{ii}| > \sum_{j=1, j\neq i}^{N} |a_{ij}| \text{ , untuk: i = 1, ... , N} \tag{4.26}$$

Jika syarat ini dipenuhi maka sistem dijamin kovergen. Meskipun sistem tidak dominan secara diagonal, iterasi masih mungkin konvergen. Kekonvergenan sistem juga ditentukan oleh tebakan awal. Tebakan awal yang terlalu jauh dari solusi sejatinya dapat menyebabkan iterasi divergen.

C.1. Metode Jacobi

1. Persamaan iteratif dimulai dengan memberikan tebakan awal sembarang sebagai solusi, misalnya $x_i = 0$, $i = 1, \cdots, 4$, yang disebut sebagai iterasi #0, dengan notasi $x_i^{(0)}$:

$$\mathbf{x}^{(0)} = (x_1^{(0)}, x_2^{(0)}, x_3^{(0)}, x_4^{(0)})^T \tag{4.27}$$

2. Selanjutnya dilakukan iterasi pertama, kedua, dan seterusnya untuk menghitung nilai-nilai nilai-nilai $x_i^{(k+1)}$, $i = 1, \cdots, 4$ dari hasil iterasi sebelumnya $x_i^{(k)}$, $i = 1, \cdots, 4$.

3. Iterasi pertama:

$$\begin{aligned}
x_1^{(1)} &= \frac{b_1 - a_{12}x_2^{(0)} - a_{13}x_3^{(0)} - a_{14}x_4^{(0)}}{a_{11}} \\
x_2^{(1)} &= \frac{b_2 - a_{21}x_1^{(0)} - a_{23}x_3^{(0)} - a_{24}x_4^{(0)}}{a_{22}} \\
x_3^{(1)} &= \frac{b_3 - a_{31}x_1^{(0)} - a_{32}x_2^{(0)} - a_{34}x_4^{(0)}}{a_{33}} \\
x_4^{(1)} &= \frac{b_4 - a_{41}x_1^{(0)} - a_{42}x_2^{(0)} - a_{43}x_3^{(0)}}{a_{44}}
\end{aligned} \tag{4.28}$$

4. Iterasi kedua:

$$\begin{aligned}
x_1^{(2)} &= \frac{b_1 - a_{12}x_2^{(1)} - a_{13}x_3^{(1)} - a_{14}x_4^{(1)}}{a_{11}} \\
x_2^{(2)} &= \frac{b_2 - a_{21}x_1^{(1)} - a_{23}x_3^{(1)} - a_{24}x_4^{(1)}}{a_{22}} \\
x_3^{(2)} &= \frac{b_3 - a_{31}x_1^{(1)} - a_{32}x_2^{(1)} - a_{34}x_4^{(1)}}{a_{33}} \\
x_4^{(2)} &= \frac{b_4 - a_{41}x_1^{(1)} - a_{42}x_2^{(1)} - a_{43}x_3^{(1)}}{a_{44}}
\end{aligned} \tag{4.29}$$

5. Iterasi ke k+1:

$$
\begin{aligned}
x_1^{(k+1)} &= \frac{b_1 - a_{12}x_2^{(k)} - a_{13}x_3^{(k)} - a_{14}x_4^{(k)}}{a_{11}} \\
x_2^{(k+1)} &= \frac{b_2 - a_{21}x_1^{(k)} - a_{23}x_3^{(k)} - a_{24}x_4^{(k)}}{a_{22}} \\
x_3^{(k+1)} &= \frac{b_3 - a_{31}x_1^{(k)} - a_{32}x_2^{(k)} - a_{34}x_4^{(k)}}{a_{33}} \\
x_4^{(k+1)} &= \frac{b_4 - a_{41}x_1^{(k)} - a_{42}x_2^{(k)} - a_{43}x_3^{(k}}{a_{44}}
\end{aligned} \tag{4.30}
$$

6. Rumus umum untuk *N* persamaan dengan *N* peubah:

$$
x_i^{(k+1)} = \frac{b_i - \sum_{j=1, j \neq i}^{N} a_{ij} x_j^{(k)}}{a_{ii}}, \; k = 1, \cdots, N. \tag{4.31}
$$

7. Jika metode ini konvergen, iterasi dihentikan setelah syarat berikut dipenuhi:

$$
\max_i \left| x_i^{(k+1)} - x_i^{(k)} \right| < \varepsilon , \tag{4.32}
$$

di mana ε > 0 bilangan kecil yang kita tentukan sebelumnya.

C.2. Metode Gauss-Seidel

Kecepatan konvergen pada iterasi Jacobi dapat lebih dipercepat bila setiap nilai x_i yang baru dihasilkan segera dipakai pada persamaan berikutnya untuk menentukan nilai x_{i+1} yang lainnya.

1. Seperti pada metode iterasi Jacobi, iterasi Gauss-Seidel dimulai dengan memberi nilai awal $x_i = 0 \; \forall i$, yang disebut sebagai iterasi #0, dengan notasi $x_i^{(0)}$.

2. Iterasi pertama:

$$
\begin{aligned}
x_1^{(1)} &= \frac{b_1 - a_{12}x_2^{(0)} - a_{13}x_3^{(0)} - a_{14}x_4^{(0)}}{a_{11}} \\
x_2^{(1)} &= \frac{b_2 - a_{21}x_1^{(1)} - a_{23}x_3^{(0)} - a_{24}x_4^{(0)}}{a_{22}} \\
x_3^{(1)} &= \frac{b_3 - a_{31}x_1^{(1)} - a_{32}x_2^{(1)} - a_{34}x_4^{(0)}}{a_{33}} \\
x_4^{(1)} &= \frac{b_4 - a_{41}x_1^{(1)} - a_{42}x_2^{(1)} - a_{43}x_3^{(1)}}{a_{44}}
\end{aligned} \tag{4.33}
$$

3. Iterasi kedua:

$$x_1^{(2)} = \frac{b_1 - a_{12}x_2^{(1)} - a_{13}x_3^{(1)} - a_{14}x_4^{(1)}}{a_{11}}$$
$$x_2^{(2)} = \frac{b_2 - a_{21}x_1^{(2)} - a_{23}x_3^{(1)} - a_{24}x_4^{(1)}}{a_{22}} \qquad (4.34)$$
$$x_3^{(2)} = \frac{b_3 - a_{31}x_1^{(2)} - a_{32}x_2^{(2)} - a_{34}x_4^{(1)}}{a_{33}}$$
$$x_4^{(2)} = \frac{b_4 - a_{41}x_1^{(2)} - a_{42}x_2^{(2)} - a_{43}x_3^{(2)}}{a_{44}}$$

4. Iterasi ke k+1:

$$x_1^{(k+1)} = \frac{b_1 - a_{12}x_2^{(k)} - a_{13}x_3^{(k)} - a_{14}x_4^{(k)}}{a_{11}}$$
$$x_2^{(k+1)} = \frac{b_2 - a_{21}x_1^{(k+1)} - a_{23}x_3^{(k)} - a_{24}x_4^{(k)}}{a_{22}} \qquad (4.35)$$
$$x_3^{(k+1)} = \frac{b_3 - a_{31}x_1^{(k+1)} - a_{32}x_2^{(k+1)} - a_{34}x_4^{(k)}}{a_{33}}$$
$$x_4^{(k+1)} = \frac{b_4 - a_{41}x_1^{(k+1)} - a_{42}x_2^{(k+1)} - a_{43}x_3^{(k+1}}{a_{44}}$$

5. Rumus umum:

$$x_i^{(k+1)} = \frac{b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{N} a_{ij}x_j^{(k)}}{a_{ii}}, \; k = 1, \cdots, N \qquad (4.36)$$

6. Jika metode ini konvergen, proses dihentikan pada iterasi #(k+1) jika dipenuhi syarat konvergensi:

$$\max_i \left| x_i^{(k+1)} - x_i^{(k)} \right| < \varepsilon \qquad (4.37)$$

   di mana ε > 0 bilangan kecil.

C.3. Metode Iterasi dengan Relaksasi (SOR)

Metode SOR *(Successive Over-Relaxation)* merupakan perbaikan dari Metode Gauss-Seidel dengan memasukkan faktor relaksasi (faktor pembobot) pada setiap tahap untuk mempercepat konvergensi. Parameter relaksasi ω mempunyai sifat-sifat sbb.:

- Metode Gauss-Seidel adalah kejadian khusus dari metode SOR, yaitu untuk ω = 1.
- Jika dipilih nilai ω > 1 secara *tepat*, maka metode SOR akan lebih cepat konvergen dari pada metode Gauss-Seidel.
- Menurut pengalaman, nilai ω optimal antara 1 dan 2 (ω∈ [1, 2]). Nilai ini optimal karena pemilihan ω yang terlalu kecil atau terlalu besar justru akan memperlambat konvergensi.

Metode SOR diawali dengan menulis persamaan (4.36) dalam bentuk:

$$x_i^{(k+1)} = \frac{b_i - \sum_{j<i} a_{ij} x_j^{(k+1)} - \sum_{j>i} a_{ij} x_j^{(k)}}{a_{ii}} = x_i^{(k)} + r_i^{(k)} \tag{4.38}$$

dimana $r_i^{(k)}$ adalah besarnya langkah yang harus ditambahkan pada hasil iterasi ke-k, yaitu $x_i^{(k)}$, untuk dapat menghasilkan nilai iterasi berikutnya, yaitu $x_i^{(k+1)}$. Karena itu ada alasan kuat untuk menduga: Andaikan setiap langkah $r_i^{(k)}$ agak diperbesar menjadi $\omega r_i^{(k)}$, dengan parameter ω > 1, mungkin usaha ini dapat mempercepat konvergensi. Dengan dugaan ini, dari pada persamaan (4.38), kita akan mencoba rumus iterasi sbb.:

$$x_i^{(k+1)} = x_i^{(k)} + \omega r_i^{(k)} \tag{4.39}$$

atau

$$x_i^{(k+1)} = (1-\omega)x_i^{(k)} + \frac{\omega}{a_{ii}}\left(b_i - \sum_{j<i} a_{ij} x_j^{(k+1)} - \sum_{j>i} a_{ij} x_j^{(k)}\right) \tag{4.40}$$

Pengalaman mengajarkan bahwa metode SOR memang sukses untuk banyak masalah, asal ω dipilih hampir optimal. Suatu cara sederhana menentukan ω ialah dengan coba-coba *(trial & error)*.

Metode relaksasi paling berhasil diterapkan pada matriks yang diperoleh dari penerapan metode beda hingga *(finite difference methode)* pada masalah nilai batas yqng akan dibahas di Bab VI.

Catatan:

1. Jika pada metode Jacobi, $x_i^{(k+1)}$ dihitung dari $x_i^{(k)}$ ∀*i*, maka pada metode Gauss-Seidel, hanya $x_1^{(k+1)}$ yang dihitung dari $x_i^{(k)}$ maka hasil $x_1^{(k+1)}$ yang didapat langsung digunakan untuk menghitung $x_2^{(k+1)}$. Demikian juga hasil $x_1^{(k+1)}$, $x_2^{(k+1)}$ yang didapat *langsung digunakan* untuk menghitung $x_3^{(k+1)}$, dst. Inilah yang mempercepat konvergensi.

2. Jika metode ini akan diterapkan pada masalah sistem dengan *N* buah persamaan linear dan *N* buah peubah, maka semua angka 4 di atas harus dilanjutkan hingga *N*.

3. Menurut pengalaman untuk $N < 10$, jika metode ini konvergen, syarat konvergensi di atas akan dipenuhi sebelum iterasi ke-60 (ITMAX), sehingga angka ini dapat digunakan sebagai batas maksimum banyaknya iterasi. Jadi program komputer harus dibuat di mana iterasi dengan sendirinya berhenti jika sudah mencapai ITMAX. Hal ini untuk menghindari iterasi sampai tak-terhingga jika syarat konvergensi di atas belum juga dipenuhi.

4. Suatu syarat bagi suatu sistem **Ax = b** agar metode ini konvergen ialah bahwa matrix **A** mempunyai diagonal dominan, artinya:

$$|a_{ii}| > |a_{ij}|, i \neq j,\ i,j = 1, \cdots, N$$

5. Jika matriks **A** ditulis: **A** = **L** + **D** + **U**, di mana **L** matrix segitiga bawah ($l_{ij} = 0,\ i \leq j$), **U** matrix segitiga atas ($u_{ij} = 0,\ i \geq j$), dan **D** matrix diagonal ($d_{ij} = 0,\ i \neq j$), dari natrix **A**, maka tiga metode di atas dapat ditulis dalam notasi matrix sbb.:

   - Metode Jacobi maka persamaan (4.31) menjadi:

$$\mathbf{x}^{(k+1)} = \mathbf{D}^{-1}[\mathbf{b} - (\mathbf{L} + \mathbf{U})\mathbf{x}^{(k)}]\ .$$

   - Metode Gauss-Seidell maka persamaan (4.36) menjadi:

$$\mathbf{x}^{(k+1)} = (\mathbf{D} + \mathbf{L})^{-1}(\mathbf{b} - \mathbf{U}\mathbf{x}^{(k)})\ .$$

   - Metode SOR maka persamaan (4.40) menjadi:

$$\mathbf{x}^{(k+1)} = (1-\omega)\mathbf{x}^{(k)} + \omega\mathbf{D}^{-1}[\mathbf{b} - \mathbf{L}\mathbf{x}^{(k+1)} - \mathbf{U}\mathbf{x}^{(k)}]$$

   atau

$$(\mathbf{D} + \omega\mathbf{L})\mathbf{x}^{(k+1)} = [(1-\omega)\mathbf{D} - \omega\mathbf{U}]\mathbf{x}^{(k)} + \omega\mathbf{b}\ .$$

Tentukan solusi SPL berikut dengan metode Iteratif Jacobi dan Gauss-Seidel:

$$\begin{aligned} 4x - y + z &= 7 \\ 4x - 8y + z &= -21 \\ -2x + y + 5z &= 15 \end{aligned}$$

Nilai awal $P_0 = (x_0, y_0, z_0) = (1, 2, 2)$
Solusi sebenarnya adalah (2, 4, 3)

Jawab:

A. Metode Jacobi:

$$x_{r+1} = \frac{7 + y_r - z_r}{4}$$
$$y_{r+1} = \frac{21 + 4x_r - z_r}{8}$$
$$z_{r+1} = \frac{15 + 2x_r - y_r}{5}$$

Iterasi pertama:

$$x_1 = \frac{7 + 2 - 2}{4} = 1{,}75$$
$$y_1 = \frac{21 + 4(1) + 2}{8} = 3{,}375$$
$$z_1 = \frac{15 + 2(1) - 2}{5} = 3$$

Iterasi ke dua

$$x_2 = \frac{7 + 3{,}375 - 3}{4} = 1{,}84375$$
$$y_2 = \frac{21 + 4(3{,}375) - 3}{8} = 3{,}875$$
$$z_2 = \frac{15 + 2(1{,}75) - 3{,}375}{5} = 3{,}025$$

Iterasi ke tiga

$$x_2 = \frac{7 + 3{,}875 - 3{,}025}{4} = 1{,}9625$$
$$y_2 = \frac{21 + 4(3{,}875) - 3{,}025}{8} = 4{,}184375$$
$$z_2 = \frac{15 + 2(1{,}84375) - 3{,}875}{5} = 6{,}695$$

Iterasi ke sembilan belas:

$$x_{19} = 2$$
$$y_{19} = 4$$
$$z_{19} = 3$$

B. Metode Gauss-Seidel:

$$x_{r+1} = \frac{7 + y_r - z_r}{4}$$
$$y_{r+1} = \frac{21 + 4x_{r+1} - z_r}{8}$$
$$z_{r+1} = \frac{15 + 2x_{r+1} - y_{r+1}}{5}$$

Iterasi pertama:

$$x_1 = \frac{7 + 2 - 2}{4} = 1{,}75$$
$$y_1 = \frac{21 + 4(1{,}75) - 2}{8} = 3{,}75$$
$$z_1 = \frac{15 + 2(1{,}75) - 3{,}75}{5} = 2{,}95$$

Iterasi ke dua

$$x_2 = \frac{7 + 3{,}75 - 2{,}95}{4} = 1{,}95$$
$$y_2 = \frac{21 + 4(1{,}95) - 2{,}95}{8} = 3{,}96875$$
$$z_2 = \frac{15 + 2(1{,}95) - 3{,}968375}{5} = 2{,}98625$$

Iterasi ke tiga:

$$x_2 = \frac{7 + 3{,}96875 - 2{,}98625}{4} = 1{,}995625$$

$$y_2 = \frac{21 + 4(1{,}995625) - 2{,}98625}{8} = 3{,}247984$$

$$z_2 = \frac{15 + 2(1{,}995625) - 3{,}247984}{5} = 5{,}196775$$

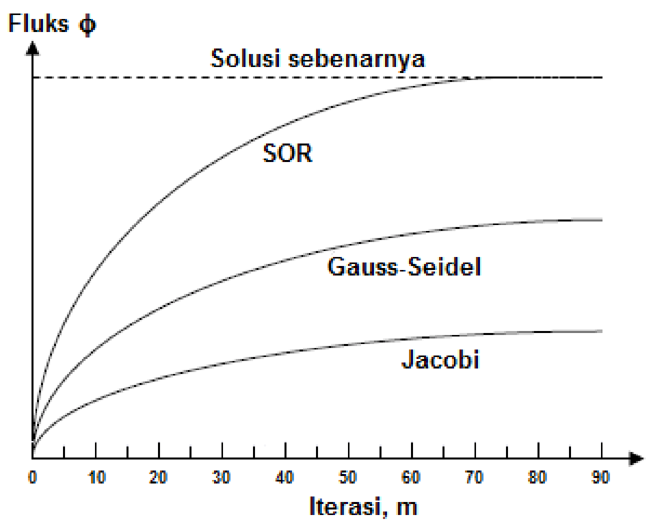
Iterasi ke sepuluh:

$$x_{10} = 2$$

$$y_{10} = 4$$

$$z_{10} = 3$$

Contoh di atas memperlihatkan bahwa kecepatan konvergensi dengan metode Gauss-Seidel lebih cepat dari pada metode Jacobi di mana dengan metode Jacobi dibutuhkan iterasi ke sembilan belas untuk mencapai nilai konvergennya sedangkan dengan metode Gauss-Seidel konvergensi sudah dapat dicapai pada iterasi ke sepuluh.



Gambar 4.5 – Perbandingan konvergensi antara metode Jacobi, Gauss-Seidel dan SOR

Metode iterasi jarang digunakan untuk menyelesaikan SPAL berukuran kecil karena dengan metode langsung, seperti metode eliminasi Gauss, hasilnya lebih efisien daripada metode iteratif. Akan tetapi untuk menyelesaikan SPAL berukuran besar dan proporsi koefisien nolnya besar, seperti sistem-sistem yang banyak ditemukan dalam sistem persamaan diferensial, maka harus digunakan metode iterasi. Gambar 4.5 memperlihatkan perbandingan penyelesaian SPAL berukuran besar menggunakan metode Jacobi, Gauss-Seidel dan SOR untuk menghitung fluks neutron ϕ. Diperlihatkan bahwa proses perhitungan metode

Gauss-Seidel memerlukan iterasi yang lebih sedikit dibandingkan dengan metode Jacobi. Akan tetapi metode SOR sebagai varian dari metode Gauss-Seidel yaitu dengan memasukkan faktor relaksasi, akan memberikan hasil konvergensi yang paling cepat.