BÖLÜM 1

LİNEER DENKLEM TAKIMLARININ ÇÖZÜM YÖNTEMLERİ

1.1- Giriş

1.2 Matrisler

1.3 Lineer denklem takımlarının çözüm yöntemleri

- 1.3.1 Gauss eliminasyon yöntemi
- 1.3.2 Gauss-Jordan Yöntemi
- 1.3.3 Thomas yöntemi
- 1.3.4 LU Ayrıştırma yöntemleri
- 1.3.5 Jacobi basit iterasyon yöntemi
- 1.3.6 Gauss-Sidel iterasyon yöntemi
- 1.3.7 SOR yöntemi

1.4 Matris tersinin sayısal hesabı

1.1 Giriş

Mühendislik problemlerinin sayısal yöntemlerinin çözümünde çoğu zaman problem bir lineer denklem takımının çözümü problemine indirgenir ve bu denklem takımının uygun ve hızlı bir yöntemle çözümü söz konusu olur. Lineer denklem takımları bu amaçla matris formda yazılarak çözülmeye çalışılır.

Bu bölümde önce matrislerle ilgili bazı hatırlatmalar yapılarak matrislerin tiplerinden ve matris işlemlerinden kısaca bahsedilecektir.

Daha sonra lineer denklem takımlarının çözümünde kullanılan sayısal yöntemlere yer verilecektir.

1.2 Matrisler

1.2.1 Matris tipleri

($M \times N$) boyutlarındaki
dikdörtgensel matris

$$A = [A] = [a_{ij}] = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1j} & \cdots & a_{1N} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2j} & \cdots & a_{2N} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3j} & \cdots & a_{3N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{i1} & a_{i2} & a_{i3} & \cdots & a_{ij} & \cdots & a_{iN} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{M1} & a_{M2} & a_{M3} & \cdots & a_{Mj} & \cdots & a_{MN} \end{bmatrix}$$

Kare matris

($M=N$) olan matris

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1j} & \cdots & a_{1N} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2j} & \cdots & a_{2N} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3j} & \cdots & a_{3N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{i1} & a_{i2} & a_{i3} & \cdots & a_{ij} & \cdots & a_{iN} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{N1} & a_{N2} & a_{N3} & \cdots & a_{Nj} & \cdots & a_{NN} \end{bmatrix}$$

Köşegen (Diyagonal) matris

Köşegen dışında bütün elemanları sıfır olan *kare matris*

$$D_N = [\alpha_i \delta_{ij}] = \begin{bmatrix} \alpha_1 & 0 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & \alpha_2 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & 0 & \alpha_3 & \cdots & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & \alpha_i & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 & \cdots & \alpha_N \end{bmatrix}$$

NOT: $\delta_{ij} = \begin{cases} 1 & i = j \quad ise \\ 0 & i \neq j \quad ise \end{cases}$ Kronecker deltası

Birim matris

Bütün köşegen elemanları *1* olan *köşegen matris*

$$I_N = [\delta_{ij}] = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 & \cdots & 1 \end{bmatrix}$$

Skalar matris

Bütün köşegen elemanları bir skaler büyüklüğe eşit olan *köşegen matris*

$$[b\delta_{ij}] = \begin{bmatrix} b & 0 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & b & 0 & \cdots & 0 & \cdots & 0 \\ 0 & 0 & b & \cdots & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & b & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 & \cdots & b \end{bmatrix}$$

Sıfır matrisi

Bütün elemanları sıfıra eşit olan matris

$$O = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 & \cdots & 0 \end{bmatrix}$$

Bol Sıfırlı matris

Çoğu elemanı sıfıra eşit olan *matris*

Simetrik matris

Köşegene göre simetrik konumlu elemanları aynı olan *kare matris*

$$a_{ij} = a_{ji}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & \cdots & \cdots & a_{1N} \\ a_{12} & a_{22} & a_{23} & \cdots & \cdots & \cdots & a_{2N} \\ a_{13} & a_{23} & a_{33} & \cdots & \cdots & \cdots & a_{3N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{1N} & a_{2N} & a_{3N} & \cdots & \cdots & \cdots & a_{NN} \end{bmatrix}$$

Çarpık simetrik matris

Köşegene göre simetrik konumlu elemanları zıt işaretle aynı olan *kare matris*

$$a_{ij} = -a_{ji}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & \cdots & \cdots & a_{1N} \\ -a_{12} & a_{22} & a_{23} & \cdots & \cdots & \cdots & a_{2N} \\ -a_{13} & -a_{23} & a_{33} & \cdots & \cdots & \cdots & a_{3N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ -a_{1N} & -a_{2N} & -a_{3N} & \cdots & \cdots & \cdots & a_{NN} \end{bmatrix}$$

Üst-üçgensel matris

Köşegen altındaki bütün elemanları sıfır olan *kare matris*

$$U = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1j} & \cdots & a_{1N} \\ 0 & a_{22} & a_{23} & \cdots & a_{2j} & \cdots & a_{2N} \\ 0 & 0 & a_{33} & \cdots & a_{3j} & \cdots & a_{3N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & a_{jj} & \cdots & a_{iN} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 & \cdots & a_{NN} \end{bmatrix}$$

Alt-üçgensel matris

Köşegen üstündeki bütün elemanları sıfır olan *kare matris*

$$L = \begin{bmatrix} a_{11} & 0 & 0 & \cdots & 0 & \cdots & 0 \\ a_{21} & a_{22} & 0 & \cdots & 0 & \cdots & 0 \\ a_{31} & a_{32} & a_{33} & \cdots & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{i1} & a_{i2} & a_{i3} & \cdots & a_{ii} & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{N1} & a_{N2} & a_{N3} & \cdots & a_{Ni} & \cdots & a_{NN} \end{bmatrix}$$

Satır matrisi veya *Satır vektörü*

Tek satırdan ibaret olan matris ($1 \times N$)

$$\begin{bmatrix} a_1 & a_2 & \cdots & a_N \end{bmatrix}$$

Sütun matrisi veya *Sütun vektörü*

Tek süturdan ibaret olan matris ($N \times 1$)

$$\begin{Bmatrix} a_1 \\ a_2 \\ \cdots \\ a_N \end{Bmatrix}$$

Birim vektörü

$$\begin{Bmatrix} 1 \\ 0 \\ 0 \end{Bmatrix}, \quad \begin{Bmatrix} 0 \\ 1 \\ 0 \end{Bmatrix}, \quad \begin{Bmatrix} 0 \\ 0 \\ 1 \end{Bmatrix}$$

Bant matrisi

Köşegen civarındaki bir bant dışında kalan bütün elemanları sıfır olan *kare matris*

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & 0 & 0 & 0 & 0 \\ a_{21} & a_{22} & a_{23} & a_{24} & 0 & 0 & 0 \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} & 0 & 0 \\ 0 & a_{42} & a_{43} & a_{44} & a_{45} & a_{46} & 0 \\ 0 & 0 & a_{53} & a_{54} & a_{55} & a_{56} & a_{57} \\ 0 & 0 & 0 & a_{64} & a_{65} & a_{66} & a_{67} \\ 0 & 0 & 0 & 0 & a_{75} & a_{76} & a_{77} \end{bmatrix}$$

Üç-diyagonalli matris

Köşegeniyle altındaki ve üstündeki komşu elemanları dışında kalan bütün elemanları sıfır olan *bant matris*

$$\begin{bmatrix} a_{11} & a_{12} & 0 & 0 & 0 & 0 & 0 \\ a_{21} & a_{22} & a_{23} & 0 & 0 & 0 & 0 \\ 0 & a_{32} & a_{33} & a_{34} & 0 & 0 & 0 \\ 0 & 0 & a_{43} & a_{44} & a_{45} & 0 & 0 \\ 0 & 0 & 0 & a_{54} & a_{55} & a_{56} & 0 \\ 0 & 0 & 0 & 0 & a_{65} & a_{66} & a_{67} \\ 0 & 0 & 0 & 0 & 0 & a_{76} & a_{77} \end{bmatrix}$$

1.2.2 Matris işlemleri

- Ancak aynı boyutlardaki iki matris toplanabilir ve çıkartılabilir.

Toplama ve çıkartmada matrislerin sırasının bir önemi yoktur.

- Aynı boyutlardaki iki matris ancak karşılıklı elemanları aynı ise birbirine eşittir.

Matrislerin çarpması

- İki matrisin çarpma işleminde matrislerin sırası ve boyutları önemlidir. Çarpma yapılabilmesi için birinci matrisin sütun sayısının ikinci matrisin satır sayısına eşit olması gereklidir.

$A\ (n \times m)$ ve $B\ (m \times r)$ ise $A.B$ çarpımı mümkündür.

$$[a_{ij}].[b_{ij}]=[c_{ij}] \qquad c_{ij}=\sum_{k=1}^{N} a_{ik} b_{kj}$$

Matrislerin çarpmadaki sırası değiştiğinde genel olarak çarpım sonucu aynı olmaz

$$A.B \neq B.A$$

- Bir matrisin bir skalerle çarpımı, matrisin herbir elemanının bu skalerle çarpımı anlamına gelir.

$$k.A=C \qquad c_{ij}=k.a_{ij}$$

- Vektörlerin skaler çarpımı

$$\begin{bmatrix} a_1 & a_2 & \cdots & a_N \end{bmatrix} \cdot \begin{Bmatrix} b_1 \\ b_2 \\ \cdots \\ b_N \end{Bmatrix} = \begin{bmatrix} a_1 b_1 + a_2 b_2 + \cdots + a_N b_N \end{bmatrix}$$

- Vektörlerin dış çarpımı

(outer product)

$$\begin{Bmatrix} a_1 \\ a_2 \\ \cdots \\ a_N \end{Bmatrix} \cdot \begin{bmatrix} b_1 & b_2 & \cdots & b_N \end{bmatrix} = \begin{bmatrix} a_1 b_1 & a_1 b_2 & \cdots & a_1 b_N \\ a_2 b_1 & a_2 b_2 & \cdots & a_2 b_N \\ \cdots & \cdots & \cdots & \cdots \\ a_N b_1 & a_N b_2 & \cdots & a_N b_N \end{bmatrix}$$

Matrisin transpozesi

Matrisin transpozesi aynı diyagonal elemanını içeren satır elemanlarıyla sütun elemanlarının yerleri değiştirilerek elde edilir.

$$A=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \cdots & \cdots & \cdots & \cdots \\ a_{N1} & a_{N2} & \cdots & a_{NN} \end{bmatrix} \rightarrow A^T=\begin{bmatrix} a_{11} & a_{21} & \cdots & a_{N1} \\ a_{12} & a_{22} & \cdots & a_{N2} \\ \cdots & \cdots & \cdots & \cdots \\ a_{1N} & a_{2N} & \cdots & a_{NN} \end{bmatrix}$$

Matrisi transpozesinin transpozesi matrisin kendisine eşittir.

$$(A^T)^T = A$$

Matrisin izi (Trace)

Diyagonal elemanlarının toplamıdır

$$Tr(A)=\sum_{i=1}^{N} a_{ii}$$

Matrisin minörleri

Bir kare matrisin herhangi bir a_{ij} *elemanının minörü*, bu elemanın üzerinde yer aldığı i 'inci satırdaki ve j 'inci sütundaki bütün elemanlar çıkartıldıktan sonra kalan matrisin determinantı olarak tanımlanır.

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1j} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2j} & \cdots & a_{2N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{i1} & a_{i2} & \cdots & a_{ij} & \cdots & a_{iN} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{N1} & a_{N2} & \cdots & a_{Nj} & \cdots & a_{NN} \end{bmatrix} \qquad M_{ij} = \begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1j-1} & a_{1j+1} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2j-1} & a_{2j+1} & \cdots & a_{2N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{i-1,1} & a_{i-1,2} & \cdots & a_{i-1,j-1} & a_{i-1,j+1} & \cdots & a_{i-1,N} \\ a_{i+1,1} & a_{i+1,2} & \cdots & a_{i+1,j-1} & a_{i+1,j+1} & \cdots & a_{i+1,N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{N1} & a_{N2} & \cdots & a_{Nj-1} & a_{Nj+1} & \cdots & a_{NN} \end{vmatrix}$$

Matrisin kofaktörleri

Bir kare matrisin herhangi bir a_{ij} *elemanının kofaktörü*, bu elemanın minörünün $(-1)^{i+j}$ ile çarpımı olarak tanımlanır.

Matrisin determinantı

Bir matrisin determinantı herhangi bir satırındaki veya herhangi bir sütunundaki bütün elemanların kofaktörleriyle çarpımlarının toplamı olarak tanımlanır.

$$\det A = [A] = \begin{vmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1i} & \cdots & a_{1j} & \cdots & a_{1N} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2i} & \cdots & a_{2j} & \cdots & a_{2N} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3i} & \cdots & a_{3j} & \cdots & a_{3N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{i1} & a_{i2} & a_{i3} & \cdots & a_{ii} & \cdots & a_{ij} & \cdots & a_{iN} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{j1} & a_{j2} & a_{j3} & \cdots & a_{ji} & \cdots & a_{jj} & \cdots & a_{jN} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{N1} & a_{N2} & a_{N3} & \cdots & a_{Ni} & \cdots & a_{Nj} & \cdots & a_{NN} \end{vmatrix} = \sum_{i=1}^{N} a_{ik} M_{ik} (-1)^{i+k} = \sum_{j=1}^{N} a_{kj} M_{kj} (-1)^{k+j}$$

Matrisin özdeğerleri ve özvektörleri

Bir matrisin özdeğerleri karakteristik polinomunun kökleri olarak tanımlanır.

Karakteristik polinom $$p_A(\lambda) = \det(A - \lambda I) = |A - \lambda I| = \begin{vmatrix} a_{11} - \lambda & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} - \lambda & \cdots & a_{2N} \\ \cdots & \cdots & \cdots & \cdots \\ a_{N1} & a_{N2} & \cdots & a_{NN} - \lambda \end{vmatrix}$$

Özdeğerler $p_A(\lambda) = 0$ eşitliğini sağlayan (λ_i, $i=1,2,\ldots N$) kökleri

Özvektörler $Aw = \lambda w$ eşitliğini sağlayan $\{w_j\}$ vektörleri

Matrisin tersi

Bir A matrisinin tersi $A.B = I_n$ eşitliğini sağlayan B matrisidir.

Bir matrisin tersi, bu matristen oluşturulan *ek matrisin* bu *matrisin determinantına* bölümü ile elde edilir.

Burada geçen ek matris ise matrisin elemanlarının kofaktörlerinde oluşturulan matrisin transpozesidir.

Örnek

$$A = \begin{bmatrix} 1 & 3 & 3 \\ 1 & 4 & 3 \\ 1 & 3 & 4 \end{bmatrix}$$

Kofaktörler matrisi

$$K = \begin{bmatrix} \begin{vmatrix} 4 & 3 \\ 3 & 4 \end{vmatrix} & -\begin{vmatrix} 1 & 3 \\ 1 & 4 \end{vmatrix} & \begin{vmatrix} 1 & 4 \\ 1 & 3 \end{vmatrix} \\ -\begin{vmatrix} 3 & 3 \\ 3 & 4 \end{vmatrix} & \begin{vmatrix} 1 & 3 \\ 1 & 4 \end{vmatrix} & -\begin{vmatrix} 1 & 3 \\ 1 & 3 \end{vmatrix} \\ \begin{vmatrix} 3 & 3 \\ 4 & 3 \end{vmatrix} & -\begin{vmatrix} 1 & 3 \\ 1 & 3 \end{vmatrix} & \begin{vmatrix} 1 & 3 \\ 1 & 4 \end{vmatrix} \end{bmatrix} = \begin{bmatrix} 7 & -1 & -1 \\ -3 & 1 & 0 \\ -3 & 0 & 1 \end{bmatrix}$$

Ek matris

$$E = \begin{bmatrix} 7 & -3 & -3 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}$$

Matrisin determinantı

$$\det(A) = \begin{bmatrix} 1 & 3 & 3 \\ 1 & 4 & 3 \\ 1 & 3 & 4 \end{bmatrix} = 1\begin{vmatrix} 4 & 3 \\ 3 & 4 \end{vmatrix} - 1\begin{vmatrix} 3 & 3 \\ 3 & 4 \end{vmatrix} + 1\begin{vmatrix} 3 & 3 \\ 4 & 3 \end{vmatrix} = 7 - 3 - 3 = 1$$

Ters matris

$$A^{-1} = \begin{bmatrix} 7 & -3 & -3 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}$$

Kontrol

$$AA^{-1} = \begin{bmatrix} 1 & 3 & 3 \\ 1 & 4 & 3 \\ 1 & 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} 7 & -3 & -3 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = I_3$$

1.3. Lineer denklem takımlarının sayısal çözümü

Bir lineer denklem takımını genel olarak

$$
\begin{array}{ccccccccccccccc}
a_{11}x_1 & + a_{12}x_2 & + a_{13}x_3 & \cdots & + a_{1i}x_3 & \cdots & + a_{1j}x_3 & \cdots & + a_{1N}x_N & = & b_1 \\
a_{21}x_1 & + a_{22}x_2 & + a_{23}x_3 & \cdots & + a_{2i}x_3 & \cdots & + a_{2j}x_3 & \cdots & + a_{2N}x_N & = & b_2 \\
a_{31}x_1 & + a_{32}x_2 & + a_{33}x_3 & \cdots & + a_{3i}x_3 & \cdots & + a_{3j}x_3 & \cdots & + a_{3N}x_N & = & b_3 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & & \cdots \\
a_{i1}x_1 & + a_{i2}x_2 & + a_{i3}x_3 & & + a_{ii}x_3 & & + a_{ij}x_3 & & + a_{iN}x_N & = & b_i \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & & \cdots \\
a_{j1}x_1 & + a_{j2}x_2 & + a_{j3}x_3 & & + a_{ji}x_3 & & + a_{jj}x_3 & & + a_{jN}x_N & = & b_j \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & & \cdots \\
a_{N1}x_1 & + a_{N2}x_2 & + a_{N3}x_3 & \cdots & + a_{Ni}x_3 & \cdots & + a_{Nj}x_3 & \cdots & + a_{NN}x_N & = & b_N
\end{array}
\tag{1.1}
$$

veya matris formda

$$
\begin{bmatrix}
a_{11} & a_{12} & a_{13} & \cdots & a_{1i} & \cdots & a_{1j} & \cdots & a_{1N} \\
a_{21} & a_{22} & a_{23} & \cdots & a_{2i} & \cdots & a_{2j} & \cdots & a_{2N} \\
a_{31} & a_{32} & a_{33} & \cdots & a_{3i} & \cdots & a_{3j} & \cdots & a_{3N} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
a_{i1} & a_{i2} & a_{i3} & & a_{ii} & & a_{ij} & & a_{iN} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
a_{j1} & a_{j2} & a_{j3} & & a_{ji} & & a_{jj} & & a_{jN} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
a_{N1} & a_{N2} & a_{N3} & \cdots & a_{Ni} & \cdots & a_{Nj} & \cdots & a_{NN}
\end{bmatrix}
\cdot
\begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ \cdots \\ x_i \\ \cdots \\ x_j \\ \cdots \\ x_N \end{Bmatrix}
=
\begin{Bmatrix} b_1 \\ b_2 \\ b_3 \\ \cdots \\ b_i \\ \cdots \\ b_j \\ \cdots \\ b_N \end{Bmatrix}
\tag{1.2}
$$

şeklinde ifade etmek mümkündür.

Lineer denklem takımlarının sayısal çözümlenmesinde kullanılan yöntemleri

-Eliminasyon Yöntemleri:

- İteratif Yöntemler:

olarak sınıflandırmak mümkündür.

Eliminasyon yöntemlerinde en çok bilinen birisi ”*Gauss eliminasyon yöntemi*‟ ve türevleridir. Bu yöntemlerde bazen pivot uygulamalarına gerek duyulur. Bir diğer grup etkin eliminasyon yöntemi ”*LU ayrıştırma yöntemleri*‟ olarak bilinir.

Büyük lineer denklem takımları için genel olarak iteratif yöntemler tercih edilir. Bunlar arasında “*Jacobi basit iterasyon yöntemi*”, *Gauss-Seidel iterasyon yöntemi*” ve “*Ardarda aşırı gevşetme (Succesive Over Relaxation – SOR) yöntemi* sayılabilir.

Katsayılar matrisinin bir bant matrisi olması ve blok bant matrisi olması gibi durumlarda daha özel yöntemler gerekmektedir. Örneğin üç-diyagonalli bant matrisler halinde “*Thomas yöntemi*” tercih edilmektedir.

1.3.1- Gauss eliminasyon yöntemi

Gauss-eliminasyon yöntemi dolu ve diyagonal elemanları baskın olan matrislerin çözümü için uygun bir yöntemdir. İki aşamalıdır:

- Üst-üçgenselleştirme aşaması

Lineer denklem sistemindeki *denklemlerden herhangi birinin iki tarafının sıfırdan farklı bir sayıyla bölünmesi* veya *çarpılması* denklemi değiştirmez. Yine bu *denklemlerden ikisinin birbiriyle toplanması* veya *çıkartılması* lineer denklem takımının çözümünü değiştirmez.

Buna göre (1.1a) denklem sistemindeki *birinci denklemin her iki tarafı a_{11} ile böldükten sonra* sırasıyla

- a_{21} ile çarpılıp ikinci denklemden çıkartılarak
- a_{31} ile çarpılıp üçüncü denklemden çıkartılarak

  ....
- a_{i1} ile çarpılıp *i* 'inci denklemden çıkartılarak

  ....
- a_{N1} ile çarpılıp *N* 'inci denklemden çıkartılarak

denklem sistemi matris formda

$$\begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1i}^{(1)} & \cdots & a_{1j}^{(1)} & \cdots & a_{1N}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2i}^{(2)} & \cdots & a_{2j}^{(2)} & \cdots & a_{2N}^{(2)} \\ 0 & a_{32}^{(2)} & a_{33}^{(2)} & \cdots & a_{3i}^{(2)} & \cdots & a_{3j}^{(2)} & \cdots & a_{3N}^{(2)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & a_{i2}^{(2)} & a_{i3}^{(2)} & & a_{ii}^{(2)} & & a_{ij}^{(2)} & & a_{iN}^{(2)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & a_{j2}^{(2)} & a_{j3}^{(2)} & & a_{ji}^{(2)} & & a_{jj}^{(2)} & & a_{jN}^{(2)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & a_{N2}^{(2)} & a_{N3}^{(2)} & \cdots & a_{Ni}^{(2)} & \cdots & a_{Nj}^{(2)} & \cdots & a_{NN}^{(2)} \end{bmatrix} \cdot \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ \cdots \\ x_i \\ \cdots \\ x_j \\ \cdots \\ x_N \end{Bmatrix} = \begin{Bmatrix} b_1^{(1)} \\ b_2^{(2)} \\ b_3^{(2)} \\ \cdots \\ b_i^{(2)} \\ \cdots \\ b_j^{(2)} \\ \cdots \\ b_N^{(2)} \end{Bmatrix} \tag{1.3}$$

şekline getirilebilir. Burada

$$\left\{ \begin{array}{l} \left(\begin{array}{l} a_{ij}^{(1)} = a_{ij} \\ a_{ij}^{(2)} = a_{ij}^{(1)} - \dfrac{a_{1j}^{(1)}}{a_{11}^{(1)}} \cdot a_{i1}^{(1)} \end{array} \right) \quad ; \quad j = 1,2,....N \\ b_i^{(2)} = b_i^{(1)} - \dfrac{b_1^{(1)}}{a_{11}^{(1)}} \cdot a_{i1}^{(1)} \end{array} \right\} \quad ; \quad i = 2,3,....N$$

Benzeri işlemler *ikinci denklem* ile sonraki denklemler arasında tekrarlanarak (1.3.3) denklem sistemi

$$\begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1i}^{(1)} & \cdots & a_{1j}^{(1)} & \cdots & a_{1N}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2i}^{(2)} & \cdots & a_{2j}^{(2)} & \cdots & a_{2N}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \cdots & a_{3i}^{(3)} & \cdots & a_{3j}^{(3)} & \cdots & a_{3N}^{(3)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & a_{i3}^{(3)} & & a_{ii}^{(3)} & & a_{ij}^{(3)} & & a_{iN}^{(3)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & a_{j3}^{(3)} & & a_{ji}^{(3)} & & a_{jj}^{(3)} & & a_{jN}^{(3)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & a_{N3}^{(3)} & \cdots & a_{Ni}^{(3)} & \cdots & a_{Nj}^{(3)} & \cdots & a_{NN}^{(3)} \end{bmatrix} \cdot \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ \cdots \\ x_i \\ \cdots \\ x_j \\ \cdots \\ x_N \end{Bmatrix} = \begin{Bmatrix} b_1^{(1)} \\ b_2^{(2)} \\ b_3^{(3)} \\ \cdots \\ b_i^{(3)} \\ \cdots \\ b_j^{(3)} \\ \cdots \\ b_N^{(3)} \end{Bmatrix} \tag{1.4}$$

şekline getirilebilir. Burada da

$$\left(\begin{aligned} a_{ij}^{(3)} &= a_{ij}^{(2)} - \frac{a_{2j}^{(2)}}{a_{22}^{(2)}} \cdot a_{i2}^{(2)} \quad ; \quad j = 2,3,....N \\ b_i^{(3)} &= b_i^{(2)} - \frac{b_2^{(2)}}{a_{22}^{(2)}} \cdot a_{i2}^{(2)} \end{aligned} \right) \quad ; \quad i = 3,4,....N$$

Bu işlemler sondan bir önceki denklem ile sonuncu denklem arasında gerçekleştirilinceye kadar tekrar edilecektir.

Tekrarlanan bu işlemler genelleştirmek amacıyla herhangi bir k 'ıncı aşamada yazılırsa denklem sistemi matris formda

$$\begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & a_{1N}^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & a_{2N}^{(2)} \\ 0 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & a_{3N}^{(3)} \\ 0 & 0 & a_{k-1,k-1}^{(k-1)} & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & a_{kk}^{(k)} & a_{kk+1}^{(k)} & & a_{kj}^{(k)} & & a_{kN}^{(k)} \\ 0 & 0 & 0 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & a_{ik+1}^{(k+1)} & & a_{ij}^{(k+1)} & & a_{iN}^{(k+1)} \\ 0 & 0 & 0 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & a_{Nk+1}^{(k+1)} & \cdots & a_{Nj}^{(k+1)} & \cdots & a_{NN}^{(k+1)} \end{bmatrix} \cdot \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ \cdots \\ x_i \\ \cdots \\ x_j \\ \cdots \\ x_N \end{Bmatrix} = \begin{Bmatrix} b_1^{(1)} \\ b_2^{(2)} \\ b_3^{(3)} \\ \cdots \\ b_i^{(k)} \\ \cdots \\ b_j^{(k+1)} \\ \cdots \\ b_N^{(k+1)} \end{Bmatrix} \tag{1.5}$$

şekline gelir. Burada

$$\left\{ \left(\begin{aligned} a_{ij}^{(k+1)} &= a_{ij}^{(k)} - \frac{a_{kj}^{(k)}}{a_{kk}^{(k)}} \cdot a_{ik}^{(k)} \quad ; \quad j = k,k+1,....N \\ b_i^{(k+1)} &= b_i^{(k)} - \frac{b_k^{(k)}}{a_{kk}^{(k)}} \cdot a_{ik}^{(k)} \end{aligned} \right) \quad ; \quad i = k+1,k+2,....N \right\} \quad ; \quad k = 1,2,....N-1$$

dir.

Bütün tekrar aşamalarından sonra denklem sistemi, katsayılar matrisi üst-üçgensel bir matris halini alarak

$$
\begin{bmatrix}
a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1i}^{(1)} & \cdots & a_{1j}^{(1)} & \cdots & a_{1N}^{(1)} \\
0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2i}^{(2)} & \cdots & a_{2j}^{(2)} & \cdots & a_{2N}^{(2)} \\
0 & 0 & a_{33}^{(3)} & \cdots & a_{3i}^{(3)} & \cdots & a_{3j}^{(3)} & \cdots & a_{3N}^{(3)} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
0 & 0 & 0 & \cdots & a_{ii}^{(i)} & & a_{ij}^{(i)} & \cdots & a_{iN}^{(i)} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
0 & 0 & 0 & \cdots & 0 & & a_{jj}^{(j)} & \cdots & a_{jN}^{(j)} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
0 & 0 & 0 & \cdots & 0 & \cdots & 0 & \cdots & a_{NN}^{(N-1)}
\end{bmatrix}
\cdot
\begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ \cdots \\ x_i \\ \cdots \\ x_j \\ \cdots \\ x_N \end{Bmatrix}
=
\begin{Bmatrix} b_1^{(1)} \\ b_2^{(2)} \\ b_3^{(3)} \\ \cdots \\ b_i^{(i)} \\ \cdots \\ b_j^{(j)} \\ \cdots \\ b_N^{(N-1)} \end{Bmatrix}
\tag{1.6}
$$

şekline gelir.

- Geri süpürme aşaması

Katsayılar matrisi üst-üçgensel hale getirilen denklem sisteminin çözümü en son bilinmeyenden başlanarak geriye doğru gerçekleştirilir.

Bu işlemler sırasında indislerin karışmaması için denklem sistemini, katsayılarına yeni isim vererek

$$
\begin{array}{l}
d_{11}x_1 + d_{12}x_2 + d_{13}x_3 \cdots + d_{1i}x_i \cdots + d_{1j}x_j \cdots + d_{1N-1}x_{N-1} + d_{1N}x_N = e_1 \\
\qquad d_{22}x_2 + d_{23}x_3 \cdots + d_{2i}x_i \cdots + d_{2j}x_j \cdots + d_{2N-1}x_{N-1} + d_{2N}x_N = e_2 \\
\qquad\qquad d_{33}x_3 \cdots + d_{3i}x_i \cdots + d_{3j}x_j \cdots + d_{3N-1}x_{N-1} + d_{3N}x_N = e_3 \\
\cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \\
\qquad\qquad\qquad d_{ii}x_i \quad + d_{ij}x_j \quad + d_{iN-1}x_{N-1} + d_{iN}x_N = e_i \\
\cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \\
\\
\cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots + d_{N-1,N-1}x_{N-1} + d_{N-1,N}x_N = e_{N-1} \\
\qquad + \qquad \cdots \qquad \cdots \qquad \cdots \qquad\qquad d_{NN}x_N = e_N
\end{array}
\tag{1.7}
$$

şeklinde yazalım. Buna göre sonuncu bilinmeyen en son denklemden kolaylıkla

$$x_N = \frac{e_N}{d_{NN}}$$

ve sondan bir önceki bilinmeyen de bir önceki denklem kullanılarak

$$x_{N-1} = \frac{e_{N-1} - d_{N-1,N}\,x_N}{d_{N-1,N-1}}$$

şeklinde hesaplanabilir. Bu işlemler birinci denklemden birinci bilinmeyen bulununcaya kadar tekrar edilir. Genelleştirme amacıyla geri-süpürme işleminin herhangi bir *i* 'inci aşaması için

$$x_i = \frac{e_i - \sum_{j=i+1}^{N} d_{ij}x_j}{d_{ii}} \quad ; \quad i = N-1, N-2, \ldots .2, 1$$

algoritması yazılabilir.

Gauss-Eliminasyon yönteminin bilgisayar uygulaması

Bilgisayar uygulamasında lineer denklem takımının katsayılar matrisi bir kare matris yerine sütun sayısı satır sayısından bir fazla olan bir dikdörtgensel matris şeklinde tanımlanıp sağ taraf vektörü de bu matrisin en son sütununa yerleştirilerek bütün işlemler bu matris üzerinde yapılabilir.

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1i} & \cdots & a_{1j} & \cdots & a_{1N} & a_{1N+1} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2i} & \cdots & a_{2j} & \cdots & a_{2N} & a_{2N+1} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3i} & \cdots & a_{3j} & \cdots & a_{3N} & a_{3N+1} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{i1} & a_{i2} & a_{i3} & & a_{ii} & & a_{ij} & & a_{iN} & a_{iN+1} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{j1} & a_{j2} & a_{j3} & & a_{ji} & & a_{jj} & & a_{jN} & a_{jN+1} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{N1} & a_{N2} & a_{N3} & \cdots & a_{Ni} & \cdots & a_{Nj} & \cdots & a_{NN} & a_{NN+1} \end{bmatrix} \qquad (1.8)$$

Şayet katsayılar matrisinin bir başka amaç için aynen saklanması gerekiyorsa bir başka matriste yedeklenmesinde yarar vardır.

Bu durumda üst-üçgenselleştirme aşamasının *k* 'ıncı adımında katsayılar matrisi

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & \cdots & \cdots & \cdots & \cdots & a_{1N} & a_{1N+1} \\ 0 & a_{22} & \cdots & \cdots & \cdots & \cdots & \cdots & a_{2N} & a_{2N+1} \\ 0 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & a_{kk} & a_{kk+1} & \cdots & a_{kj} & \cdots & a_{kN} & a_{kN+1} \\ 0 & 0 & 0 & a_{k+1k+1} & \cdots & a_{k+1j} & \cdots & a_{k+1N} & a_{k+1N+1} \\ 0 & 0 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & a_{ik+1} & \cdots & a_{ij} & \cdots & a_{iN} & a_{iN+1} \\ 0 & 0 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & a_{Nk+1} & \cdots & a_{Nj} & \cdots & a_{NN} & a_{NN+1} \end{bmatrix} \qquad (1.9)$$

gerekli bilgisayar algoritmaları

$$\left\{ \begin{pmatrix} P \leftarrow a_{ik} / a_{kk} \\ a_{ij} \leftarrow a_{ij} - a_{kj} \cdot P \, ; \quad j = k+1, k+2, ...N+1 \end{pmatrix} ; \quad i = k+1, k+2, ...N \right\} ; \quad k = 1,2,....N-1$$

ve geri-süpürme aşamasında gerekli algoritmalar da

$$x_N = \frac{a_{NN+1}}{a_{NN}}$$

$$x_i = \frac{a_{iN+1} - \sum_{j=i+1}^{N} a_{ij} x_j}{a_{ii}} \quad ; \quad i = N-1, N-2, ...2,1$$

şeklinde yazılabilir.

Örnek uygulama

Denklem sistemi

$$\begin{bmatrix} 4 & -2 & 1 \\ -3 & -1 & 4 \\ 1 & -1 & 3 \end{bmatrix} \cdot \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} = \begin{Bmatrix} 15 \\ 8 \\ 13 \end{Bmatrix}$$

Katsayılar matrisi

$$\left[\begin{array}{ccc:c} 4 & -2 & 1 & 15 \\ -3 & -1 & 4 & 8 \\ 1 & -1 & 3 & 13 \end{array}\right]$$

Birinci sütun sıfırlanarak

$$\begin{matrix} R_2 - (-3/4) \times R_1 & \rightarrow \\ R_3 - (1/4) \times R_1 & \rightarrow \end{matrix} \left[\begin{array}{ccc:c} 4 & -2 & 1 & 15 \\ 0 & -2.5 & 4.75 & 19.25 \\ 0 & -0.5 & 2.75 & 9.25 \end{array}\right]$$

İkinci sütun sıfırlanarak

$$R_3 - (-0.5/-2.5) \times R_2 \quad \rightarrow \quad \left[\begin{array}{ccc:c} 4 & -2 & 1 & 15 \\ 0 & -2.5 & 4.75 & 19.25 \\ 0 & 0 & 1.80 & 5.40 \end{array}\right]$$

Geri süpürme ile

$$x_3 = \frac{5.40}{1.80} = 3$$

$$x_2 = \frac{19.25 - 4.75 \times 3}{-2.5} = -2$$

$$x_1 = \frac{15 - [(-2) \times (-2) + 1 \times 3]}{4} = 2$$

Uyarı

Bazı matrislerde üst-üçgenselleştirme aşamasının herhangi bir adımında, daha önce yapılmış olan işlemlerin sonucunda a_{kk} *diyagonal elemanı*nın değeri *sıfır* olabilir. Bu gibi durumlarda sıfırla bölme sorunu nedeniyle yukarıda izah edilen Gauss yoketme yöntemi devam ettirilemez.

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & \cdots & \cdots & \cdots & a_{1N} & a_{1N+1} \\ 0 & a_{22} & a_{23} & \cdots & \cdots & \cdots & \cdots & a_{2N} & a_{2N+1} \\ 0 & 0 & a_{33} & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & & a_{k-1k-1} & a_{k-1k} & a_{k-1k+1} & \cdots & a_{k-1N} & a_{k-1N+1} \\ 0 & 0 & 0 & 0 & \boxed{0} & a_{kk+1} & \cdots & a_{kN} & a_{kN+1} \\ 0 & 0 & 0 & 0 & a_{k+1k} & a_{k+1k+1} & \cdots & a_{k+1N} & a_{k+1N+1} \\ 0 & 0 & 0 & 0 & a_{k+2k} & a_{k+2k+1} & \cdots & a_{k+2N} & a_{k+2N+1} \\ 0 & 0 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & a_{Nk} & a_{Nk+1} & \cdots & a_{NN} & a_{NN+1} \end{bmatrix}$$

Bu sorunu gidermenin bir yolu *kısmi pivot* uygulamasıdır.

Gauss yoketme yönteminde pivot uygulamaları

Lineer bir denklem takımındaki iki denklemin sırasının değiştirilmesi çözümü hiçbir şekilde değiştirmez.

Buna göre Gauss yönteminin üst-üçgenselleştirme aşamasında herhangi bir *k* ıncı adımda a_{kk} diyagonal elemanı sıfır olmuşsa, bundan sonraki denklemlerden, bu diyagonal elemanının altındaki elemanı sıfır olmayan herhangi birisinin yeri *k*'ıncı denklemle değiştirilebilir. Bu uygulamaya "*kısmi pivot uygulaması*" denir.

Sıfır diyagonal altındaki elemanlardan herhangi birisi yerine maksimum olanı tercih edilerek bu denklemin yeri değiştirilirse bu stratejiye de "*maksimum pivot stratejisi*" adı verilir.

Katsayılar matrisinde başlangıçtan itibaren veya Gauss eliminasyon yönteminin üst-üçgenselleştirme aşamasının herhangi bir adımında diyagonal elemanlarından birinin veya bir çoğunun sıfır olması yanında küçük olması halinde de maksimum pivot stratejisi uygulanması yararlı olur.

Maksimum pivot stratejisi genel olarak katsayılar matrisinin, en büyük elemanları diyagonal üzerine gelecek şekilde satırlarının ve sütunlarının yerlerinin değiştirilmesi şeklindedir.

Katsayılar matrisinin iki satırının (sağ taraf elemanı da dahil olmak üzere) yer değiştirmesinin çözümü değiştirmeyeceği daha önce de belirtilmişti.

Katsayılar matrisinde *iki sütunun yer değiştirmesi* ise *bilinmeyenlerin sırasının değiştirilmesi anlamına gelir* ki bilinmeyenlerin sırasının bu işlemler boyunca nasıl değiştiğinin tespit edilmesi ve çözüm elde edildikten sonra bilinmeyenlerin sırasının eski haline getirilebilmesi için bir sıralama işlemi yapılması gerekir.

Örnek

Yandaki 4 bilinmeyenli denklem sisteminde katsayılar matrisinin en büyük elemanı (maksimum pivot) 6 değerine sahip olup dördüncü satırın üçüncü sütununda yer almaktadır.

$$\begin{bmatrix} 2 & 1 & 1 & 5 \\ 2 & 5 & 3 & 1 \\ 4 & 2 & 2 & 2 \\ 2 & 3 & \boxed{6} & 1 \end{bmatrix} \cdot \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{Bmatrix} = \begin{Bmatrix} 9 \\ 11 \\ 5 \\ 12 \end{Bmatrix}$$

Bu maksimum pivotun ilk diyagonal elemanı olması için:

- önce birinci satırla dördüncü satırın yerlerinin değişmesi

$$\begin{bmatrix} 2 & 3 & 6 & 1 \\ 2 & 5 & 3 & 1 \\ 4 & 2 & 2 & 2 \\ 2 & 1 & 1 & 5 \end{bmatrix} \cdot \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{Bmatrix} = \begin{Bmatrix} 12 \\ 11 \\ 5 \\ 9 \end{Bmatrix}$$

- sonra da birinci sütunla üçüncü sütunun yerlerinin değişmesi gerekmektedir. Ancak bu sütun değişikliği durumunda bilinmeyenler vektöründe birinci bilinmeyen ile üçüncü bilinmeyeni yerinin de değiştirilmesi gerekir.

$$\begin{bmatrix} 6 & 3 & 2 & 1 \\ 3 & 5 & 2 & 1 \\ 2 & 2 & 4 & 2 \\ 1 & 1 & 2 & 5 \end{bmatrix} \cdot \begin{Bmatrix} x_3 \\ x_2 \\ x_1 \\ x_4 \end{Bmatrix} = \begin{Bmatrix} 12 \\ 11 \\ 5 \\ 9 \end{Bmatrix}$$

İkinci aşamada katsayılar matrisinin birinci satır ve sütunu dışında kalan diğer elemanları arasından yeni bir pivot seçilecektir. En büyük iki elemanın da değeri 5 olup bunlardan birisi ikinci köşegen üzerinde yer almaktadır. Buna göre zaten pivot köşegen üzerinde olduğundan bir değişikliğe gerek olmayacaktır.

Üçüncü aşamada katsayılar matrisinin birinci ve ikinci satırları ve sütunları dışında kalan diğer elemanları arasından yeni bir pivot aranacaktır. Bu pivot dördüncü satır ve dördüncü sütunda yer alan 5 değeri olup, buna göre önce dördüncü satırla üçüncü satırın, ardından da dördüncü sütunla üçüncü sütunun yerlerinin değişmesi gerekmektedir.Ayrıca bilinmeyenler vektöründe üçüncü ve dördüncü sırada yer alan bilinmeyenlerin yerleri de değişecektir.

$$\begin{bmatrix} 6 & 3 & 2 & 1 \\ 3 & 5 & 2 & 1 \\ 1 & 1 & 2 & 5 \\ 2 & 2 & 4 & 2 \end{bmatrix} \cdot \begin{Bmatrix} x_3 \\ x_2 \\ x_1 \\ x_4 \end{Bmatrix} = \begin{Bmatrix} 12 \\ 11 \\ 9 \\ 5 \end{Bmatrix} \rightarrow \begin{bmatrix} 6 & 3 & 1 & 2 \\ 3 & 5 & 1 & 2 \\ 1 & 1 & 5 & 2 \\ 2 & 2 & 2 & 4 \end{bmatrix} \cdot \begin{Bmatrix} x_3 \\ x_2 \\ x_4 \\ x_1 \end{Bmatrix} = \begin{Bmatrix} 12 \\ 11 \\ 9 \\ 5 \end{Bmatrix}$$

Görüldüğü gibi satırların yerleri değişmekle birlikte bilinmeyenlerin sırası değişmemiştir. Ancak sütunların yeri değişirken bilinmeyenlerin de yerleri değişmiştir.

Bu şekilde pivotlanan denklem sistemine bundan sonraki aşamada Gauss-Eliminasyon Yöntemi uygulanarak bilinmeyenler

$$x_3 = 1.312\,, \quad x_2 = 1.469\,, \quad x_4 = 1.625\,, \quad x_1 = -0.953$$

olarak elde edilir.

Ancak bilgisayar uygulamasında bilinmeyenler bir *x* dizisi içerisinde olacak ve çözüm bu dizi içerisinde

$$x_1 = -0.953\,, \quad x_2 = 1.469\,, \quad x_3 = 1.312\,, \quad x_4 = 1.625$$

şeklinde farklı bir sıra ile yer alacaktır. İşte bu nedenle pivotlama boyunca sütunların nasıl yer değiştiğinin tespit edilmesi ve çözümden sonra *x* dizisi içindeki bilinmeyen değerlerinin gerekli sıraya konulması gerekmektedir.

Gauss yoketme yönteminde LU ayrıştırma uygulaması

Gauss eliminasyon yönteminde diyagonal altındaki elemanların sıfırlanması için yapılan a_{ik}/a_{kk} bölme işleminin sonucu, diyagonalin altında sıfır olması gereken elemanın yerine yazıldığı taktirde, sonuçta üst-üçgensel olması gereken matrisin diyagonal altında yeni bir matris oluşur.

Örnekolarak daha önceki denklem takımı alınırsa
$$\left[\begin{array}{ccc|c} 4 & -2 & 1 & 15 \\ -3 & -1 & 4 & 8 \\ 1 & -1 & 3 & 13 \end{array}\right]$$

$$\begin{array}{l} R_2 - (-3/4)\times R_1 \rightarrow \\ R_3 - (1/4)\times R_1 \rightarrow \end{array} \left[\begin{array}{ccc|c} 4 & -2 & 1 & 15 \\ 0 & -2.5 & 4.75 & 19.25 \\ 0 & -0.5 & 2.75 & 9.25 \end{array}\right] \quad \text{yerine} \quad \left[\begin{array}{ccc|c} 4 & -2 & 1 & 15 \\ -3/4 & -2.5 & 4.75 & 19.25 \\ 1/4 & -0.5 & 2.75 & 9.25 \end{array}\right]$$

$$R_3 - \left(\frac{-0.5}{-2.5}\right)\times R_2 \rightarrow \left[\begin{array}{ccc|c} 4 & -2 & 1 & 15 \\ 0 & -2.5 & 4.75 & 19.25 \\ 0 & 0 & 1.80 & 5.40 \end{array}\right] \quad \text{yerine} \quad \left[\begin{array}{ccc|c} 4 & -2 & 1 & 15 \\ -3/4 & -2.5 & 4.75 & 19.25 \\ 1/4 & \frac{-0.5}{-2.5} & 1.80 & 5.40 \end{array}\right]$$

Böylece katsayılar matrisi
$$\begin{bmatrix} 4 & -2 & 1 \\ -0.75 & -2.5 & 4.75 \\ 0.25 & 0.20 & 1.80 \end{bmatrix}$$

şeklini alır. Bu matris
$$\begin{bmatrix} 1 & 0 & 0 \\ -0.75 & 1 & 0 \\ 0.25 & 0.20 & 1 \end{bmatrix} \cdot \begin{bmatrix} 4 & -2 & 1 \\ 0 & -2.5 & 4.75 \\ 0 & 0 & 1.80 \end{bmatrix}$$

şeklinde düzenlenirse, buradaki alt-üçgensel matrisle üst-üçgensel matrisin çarpımının yukarıdaki orijinal katsayılar matrisini verdiği gösterilebilir.

Yani
$$\begin{bmatrix} 1 & 0 & 0 \\ -0.75 & 1 & 0 \\ 0.25 & 0.20 & 1 \end{bmatrix} \cdot \begin{bmatrix} 4 & -2 & 1 \\ 0 & -2.5 & 4.75 \\ 0 & 0 & 1.80 \end{bmatrix} = \begin{bmatrix} 4 & -2 & 1 \\ -3 & -1 & 4 \\ 1 & -1 & 3 \end{bmatrix}$$

Böylece katsayılar matrisi $A = L.U$ şeklinde çarpanlara ayrıştırılmış olmaktadır.

NOT:

LU çarpanlara ayırma işlemi bir matrisin determinantının hesabı için etkin bir yöntemdir. Şöyle ki:

Yukarıdaki örnekte söz konusu olana katsayılar matrisinin determinantı, kofaktörler yardımıyla hesaplanırsa

$$\det(A)=\begin{vmatrix} 4 & -2 & 1 \\ -3 & -1 & 4 \\ 1 & -1 & 3 \end{vmatrix}=4\begin{vmatrix} -1 & 4 \\ -1 & 3 \end{vmatrix}-(-3)\begin{vmatrix} -2 & 1 \\ -1 & 3 \end{vmatrix}+1\begin{vmatrix} -2 & 1 \\ -1 & 4 \end{vmatrix}=4\times 1+3\times(-5)+1\times(-7)=-18$$

şeklinde elde edilir.

Diğer taraftan iki matrisin çarpımının determinantı bu matrislerin determinantlarının çarpımına eşittir. Buna göre örnekteki matrisin determinantı bunun LU çarpanlarının determinantları çarpımına eşit olacaktır.

$$\det(A)=\begin{vmatrix} 1 & 0 & 0 \\ -0.75 & 1 & 0 \\ -0.25 & 0.20 & 1 \end{vmatrix}\times\begin{vmatrix} 4 & -2 & 1 \\ 0 & -2.5 & 4.75 \\ 0 & 0 & 1.80 \end{vmatrix}$$

Bu matrislerin her birinin determinantlarının diyagonal elemanları çarpımına eşit olduğu kolaylıkla görülmektedir. Buna göre

$$\det(A)=(1\times 1\times 1)\times\left(4\times -2.5\times 1.80\right)=-18$$

elde edilir.

1.3.2 Gauss-Jordan Yöntemi

Gauss-eliminasyon yöntemindeki geri-süpürme aşaması üst-üçgenselleştirme aşamasıyla birleştirilebilir.

Bu tipteki yöntemlerden biri olan Gauss-Jordan yönteminde diyagonal altındaki elemanlar sıfır yapılırken aynı anda diyagonal üzerindeki elemanlar da sıfırlanır, ayrıca diyagonal elemanlarının değerleri *1* yapılır. Böylece işlemlerin sonucunda katsayılar matrisi birim matrise dönüşürken sağ taraf vektörü de bilinmeyenler (yani çözüm) vektörüne dönüşür.

Örnek uygulama

Örnek katsayılar matrisi

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

1 ve 4 üncü satırların yeri değiştirilerek

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

1 inci satır diyagonal elemanıyla bölünerek
$$\left[\begin{array}{cccc:c} 1 & 0.1667 & -1 & -0.8333 & 1 \\ 2 & 2 & 3 & 2 & -2 \\ 4 & -3 & 0 & 1 & -7 \\ 0 & 2 & 0 & 1 & 0 \end{array}\right]$$

1. diyagonal altındaki elemanlar sıfırlanarak
$$\left[\begin{array}{cccc:c} 1 & 0.1667 & -1 & -0.8333 & 1 \\ 0 & 1.6667 & 5 & 3.6667 & -4 \\ 0 & -3.6667 & 4 & 4.3334 & -11 \\ 0 & 2 & 0 & 1 & 0 \end{array}\right]$$

2. ve *3*. satırların yerini değiştirerek
$$\left[\begin{array}{cccc:c} 1 & 0.1667 & -1 & -0.8333 & 1 \\ 0 & -3.6667 & 4 & 4.3334 & -11 \\ 0 & 1.6667 & 5 & 3.6667 & -4 \\ 0 & 2 & 0 & 1 & 0 \end{array}\right]$$

2. satırı diyagonal elemanıyla bölerek
$$\left[\begin{array}{cccc:c} 1 & 0.1667 & -1 & -0.8333 & 1 \\ 0 & 1 & -1.0909 & -1.1818 & 3 \\ 0 & 1.6667 & 5 & 3.6667 & -4 \\ 0 & 2 & 0 & 1 & 0 \end{array}\right]$$

2. diyagonal üst ve altındaki elemanları sıfır yaparak
$$\left[\begin{array}{cccc:c} 1 & 0 & -0.8182 & -.0.6364 & 0.5 \\ 0 & 1 & -1.0909 & -1.1818 & 3 \\ 0 & 0 & 6.8182 & 5.6364 & -9 \\ 0 & 0 & 2.1818 & 3.3636 & -6 \end{array}\right]$$

3. satırı diyagonal elemanıyla bölerek
$$\left[\begin{array}{cccc:c} 1 & 0 & -0.8182 & -.0.6364 & 0.5 \\ 0 & 1 & -1.0909 & -1.1818 & 3 \\ 0 & 0 & 1 & 0.8267 & -1.32 \\ 0 & 0 & 2.1818 & 3.3636 & -6 \end{array}\right]$$

3. diyagonal üst ve altındaki elemanları sıfır yaparak
$$\left[\begin{array}{cccc:c} 1 & 0 & 0 & 0.0400 & -0.58 \\ 0 & 1 & 0 & -0.2800 & 1.56 \\ 0 & 0 & 1 & 0.8267 & -1.32 \\ 0 & 0 & 0 & 1.5600 & -3.12 \end{array}\right]$$

4. satırı diyagonal elemanıyla bölerek
$$\left[\begin{array}{cccc:c} 1 & 0 & 0 & 0.0400 & -0.58 \\ 0 & 1 & 0 & -0.2800 & 1.56 \\ 0 & 0 & 1 & 0.8267 & -1.32 \\ 0 & 0 & 0 & 1 & -2 \end{array}\right]$$

4. diyagonal üstündeki elemanları sıfır yaparak
$$\left[\begin{array}{cccc:c} 1 & 0 & 0 & 0 & -0.5 \\ 0 & 1 & 0 & 0 & 1. \\ 0 & 0 & 1 & 0 & 0.3333 \\ 0 & 0 & 0 & 1 & -2 \end{array}\right]$$

Son sütundaki elemanlar bilinmeyenlerin çözümüdür.

İşlem sayısı

Gauss eliminasyon yönteminde her iki aşamada yapılan işlemlerin toplam sayısı ($2n^3/3+3n^2/2-7n/6$) olup işlem mertebesini kısaca ($2n^3/3$) olarak belirmek mümkündür.

Buna karşılık Gauss-Jordan yöntemindeki işlem mertebesi (n^3) olup Gauss eliminasyon yöntemine kıyasla yaklaşık *%50* kadar daha fazla işlem yapılır.

Ölçeklendirme

Eliminasyon yöntemlerinde katsayılar matrisinin elemanlarının büyüklükleri arasında çok fazla farklılıklar varsa bu durum işlemler sırasında önemli yuvarlatma hatalarına yol açabilir. Bunu önlemenin bir yolu her bir denklemin katsayılarını bu katsayıların en büyüğü ile bölmektir.

Örnek

$$\begin{bmatrix} 3 & 2 & 100 & 105 \\ -1 & 3 & 100 & 102 \\ 1 & 2 & -1 & 2 \end{bmatrix}$$

Denklem sisteminde ilk iki denklemdeki en büyük katsayılar 100 iken sonuncu denklemin en büyük katsayısı 2 dir. Buna göre ilk iki denklemi 100 ile ve sonuncu denklemi de 2 ile bölerek çözümleme işlemlerini

$$\begin{bmatrix} 0.03 & 0.02 & 1 & 1.05 \\ -0.01 & 0.03 & 1 & 1.02 \\ 0.5 & 1 & -0.5 & 1 \end{bmatrix}$$

denklem sistemi üzerinde yapmak yuvarlatma hatalarını azaltacaktır.

1.3.3- Thomas yöntemi

Hesaplamalı akışkanlar dinamiğinde ve Hesaplamalı mühendisliğin bazı problemlerinde zaman zaman üç-diyagonalli katsayılar matrisine sahip lineer denklem takımlarıyla karşılaşılır. Üç-diyagonalli katsayılar matrisine sahip böyle bir lineer denklem takımı matris biçiminde normal olarak aşağıdaki gibi gösterilebilir.

$$\begin{bmatrix} a_{11} & a_{12} & 0 & \cdots & \cdots & \cdots & 0 \\ a_{21} & a_{22} & a_{23} & \cdots & \cdots & \cdots & 0 \\ 0 & a_{32} & a_{33} & \cdots & \cdots & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & a_{i,i-1} & a_{i,i} & a_{i,i+1} & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & 0 & a_{N,N-1} & a_{NN} \end{bmatrix} \cdot \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ \cdots \\ x_i \\ \cdots \\ x_N \end{Bmatrix} = \begin{Bmatrix} b_1 \\ b_2 \\ b_3 \\ \cdots \\ b_i \\ \cdots \\ b_N \end{Bmatrix}$$

Ancak katsayılar matrisinin çoğu sıfır olan elemanları için bilgisayar hafızasında gereksiz yer işgal etmemek ve gereksiz işlemlerden kaçınmak amacıyla ($N\times N$) boyutlarında bir katsayılar matrisi yerine ($N\times 3$) boyutlarında bir katsayılar matrisi kullanacak biçimde bir düzenleme ve buna uygun bir çözüm algoritması kullanılması tercih edilir.

Çözüm için çok tercih edilen bir yöntem Thomas algoritmasıdır. Thomas algoritması aslında Gauss eliminasyon yönteminin üç kolonlu bir dikdörtgensel matris kullanılarak yapılan özel bir uygulamasıdır.

Yukarıdaki denklem sistemi

$$a_{i,i-1} \rightarrow l_i\,; \qquad a_{ii} \rightarrow d_i\,; \qquad a_{i,i+1} \rightarrow u_i\,; \qquad b_i \rightarrow r_i$$

olmak üzere düzenlenirse:

$$\begin{bmatrix} d_1 & u_1 & 0 & \cdots & \cdots & \cdots & 0 \\ l_2 & d_2 & u_2 & \cdots & \cdots & \cdots & 0 \\ 0 & l_3 & d_3 & \cdots & \cdots & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & l_i & d_i & u_i & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & 0 & l_N & d_N \end{bmatrix} \cdot \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ \cdots \\ x_i \\ \cdots \\ x_N \end{Bmatrix} = \begin{Bmatrix} r_1 \\ r_2 \\ r_3 \\ \cdots \\ r_i \\ \cdots \\ r_N \end{Bmatrix}$$

Gauss eliminasyon yönteminin esasının diyagonal altında kalan bütün elemanların sıfır olmasını sağlayacak işlemler olduğu hatırlanırsa bu denklem sistemindeki ilk denklem l_2 ile ve ikinci denklem de d_1 ile çarpılıp birinci denklem ikincisinden çıkarıldıktan sonra her iki taraf d_1 ile bölünerek

$$\begin{aligned} l_2 d_1 x_1 + l_2 u_1 x_2 &= l_2 r_1 \\ d_1 l_2 x_1 + d_1 d_2 x_2 + d_1 u_2 x_3 &= d_1 r_2 \end{aligned} \qquad \Rightarrow \qquad \left(d_2 - \frac{l_2 u_1}{d_1} \right) x_2 + u_2 x_3 = r_2 - \frac{r_1 l_2}{d_1}$$

elde edilir. Böylece ikinci denklemdeki x_1 bilinmeyeni yok edilmiş veya diğer bir deyişle katsayılar matrisinde diyagonal elemanının altındaki katsayı sıfırlanmış olmaktadır. Bu eşitlik

$$\boxed{d_2' = d_2 - \frac{l_2 u_1}{d_1} \quad ; \quad r_2' = r_2 - \frac{r_1 l_2}{d_1}} \qquad \text{olmak üzere} \qquad \boxed{d_2' x_2 + u_2 x_3 = r_2'}$$

şeklinde yazılabilir. Böylece denklem sistemi

$$\begin{bmatrix} d_1 & u_1 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & d_2' & u_2 & \cdots & \cdots & \cdots & 0 \\ 0 & l_3 & d_3 & \cdots & \cdots & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & l_i & d_i & u_i & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & 0 & l_N & d_N \end{bmatrix} \cdot \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ \cdots \\ x_i \\ \cdots \\ x_N \end{Bmatrix} = \begin{Bmatrix} r_1 \\ r_2' \\ r_3 \\ \cdots \\ r_i \\ \cdots \\ r_N \end{Bmatrix}$$

şekline gelir. Yukarıda yapılan eliminasyon işlemi denklem sistemindeki ikinci ve üçüncü denklem arasında tekrarlanırsa, yani ikinci denklem l_3 ile ve üçüncü denklem de d_2' ile çarpılıp yine birinci denklem ikincisinden çıkarılıp, karşılıklı d_2' ile bölünerek

$$\begin{aligned} l_3 d_2' x_2 + l_3 u_2 x_3 &= l_3 r_2' \\ d_2' l_3 x_2 + d_2' d_3 x_3 + d_2' u_3 x_4 &= d_2' r_3 \end{aligned} \qquad \Rightarrow \qquad \left(d_3 - \frac{l_3 u_2}{d_2'} \right) x_3 + u_3 x_4 = r_3 - \frac{l_3 r_2'}{d_2'}$$

veya yine

$$\boxed{d_3' = d_3 - \frac{l_3 u_2}{d_2'} \quad ; \quad r_3' = r_3 - \frac{l_2 r_2'}{d_2'}} \qquad \text{olmak üzere} \qquad \boxed{d_3' x_3 + u_3 x_4 = r_3'}$$

elde edilir. Bu denklemde de x_2 bilinmeyeninin ortadan kalktığı ve katsayılar matrisinde üçüncü satırda diyagonal altındaki elemanın sıfır yapıldığı görülmektedir.

Benzeri işlemler daha sonraki denklemler için de tekrarlanabilir. Bunun için yukarıda çıkartılan bağıntılar karşılaştırılarak bir genelleştirme yapılırsa

$$\boxed{d_i' = d_i - \frac{l_i u_{i-1}}{d_{i-1}'} \quad ; \quad r_i' = r_i - \frac{l_i r_{i-1}'}{d_{i-1}'} \qquad (i = 2,3,....N)}$$

elde edilir. Bu durumda denklem sistemi de

$$\begin{bmatrix} d_1 & u_1 & 0 & .\cdots & .\cdots & .\cdots & 0 \\ 0 & d_2' & u_2 & .\cdots & .\cdots & .\cdots & 0 \\ 0 & 0 & d_3' & .\cdots & .\cdots & .\cdots & 0 \\ .\cdots & .\cdots & .\cdots & .\cdots & .\cdots & .\cdots & .\cdots \\ 0 & 0 & 0 & 0 & d_i' & u_i & 0 \\ .\cdots & .\cdots & .\cdots & .\cdots & .\cdots & .\cdots & .\cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & d_N' \end{bmatrix} \cdot \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ .\cdots \\ x_i \\ .\cdots \\ x_N \end{Bmatrix} = \begin{Bmatrix} r_1 \\ r_2' \\ r_3' \\ .\cdots \\ r_i' \\ .\cdots \\ r_N' \end{Bmatrix}$$

şekline gelir. Bu denklem sisteminde en sonuncu denklemden x_N bilinmeyeninin kolaylıkla çözülebileceği görülmektedir:

$$x_N = \frac{r_N'}{d_N'}$$

Bulunan bu büyüklük bir üstteki denklemde kullanılarak

$$x_{N-1} = \frac{r_{N-1}' - u_{N-1} x_N}{d_{N-1}'}$$

bulunur. Benzeri işlem herhangi bir x_i bilinmeyeni için

$$\boxed{x_i = \frac{r_i' - u_i x_{i+1}}{d_i'}}$$

şeklinde gerçekleştirilir.

1.3.4- LU ayrıştırma yöntemleri

$\boxed{A \cdot X = B}$ şeklindeki lineer denklem takımı, katsayılar matrisi

$\Downarrow$ $\Leftarrow$ $A = L \cdot U$ şeklinde çarpanlarına ayrılarak

$L \cdot U \cdot X = B$ şekline veya

$\Downarrow$ $\Leftarrow$ $\boxed{U \cdot X = Y}$ tanımlaması yaparak

$\boxed{L \cdot Y = B}$ getirilebilir.

Bu son denklem takımı *ileri-süpürme* yoluyla *Y vektörü* için kolaylıkla çözülebilir.

Bulunan çözüm bir önceki denklemde kullanılırsa bu denklem de geri-süpürme yoluyla *X vektörü* için çözülebilir.

Buna göre bir *LU ayrıştırma yöntemi*nin aşamaları aşağıdaki gibi sıralanabilir:

- $A = L \cdot U$ ayrıştırması
- $L \cdot Y = B$ denkleminin Y için ileri-süpürme yoluyla çözülmesi
- $U \cdot X = Y$ denkleminin X için geri-süpürme yoluyla çözülmesi

LU Ayrıştırma aşaması:

$$L \cdot U = \begin{bmatrix} l_{11} & 0 & 0 & \cdots & 0 & \cdots & \cdots & \cdots & 0 \\ l_{21} & l_{22} & 0 & \cdots & 0 & \cdots & \cdots & \cdots & 0 \\ l_{31} & l_{32} & l_{33} & \cdots & 0 & \cdots & \cdots & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ l_{k1} & l_{k2} & l_{k3} & \cdots & l_{kk} & \cdots & \cdots & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ l_{i1} & l_{i2} & l_{i3} & \cdots & l_{ik} & \cdots & \cdots & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ l_{N1} & l_{N2} & l_{N3} & \cdots & l_{Nk} & \cdots & \cdots & \cdots & l_{NN} \end{bmatrix} \cdot \begin{bmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1k} & \cdots & u_{1j} & \cdots & u_{1N} \\ 0 & u_{22} & u_{23} & \cdots & u_{2k} & \cdots & u_{2j} & \cdots & u_{2N} \\ 0 & 0 & u_{33} & \cdots & u_{3k} & \cdots & u_{3j} & \cdots & u_{3N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & u_{kk} & \cdots & u_{kj} & & u_{kN} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 & \cdots & \cdots & & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 & \cdots & 0 & \cdots & u_{NN} \end{bmatrix}$$

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1i} & \cdots & a_{1j} & \cdots & a_{1N} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2i} & \cdots & a_{2j} & \cdots & a_{2N} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3i} & \cdots & a_{3j} & \cdots & a_{3N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{i1} & a_{i2} & a_{i3} & & a_{ii} & & a_{ij} & & a_{iN} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{j1} & a_{j2} & a_{j3} & & a_{ji} & & a_{jj} & & a_{jN} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{N1} & a_{N2} & a_{N3} & \cdots & a_{Ni} & \cdots & a_{Nj} & \cdots & a_{NN} \end{bmatrix}$$

L matrisinin *k* ‘ıncı satırı *U* matrisinin *j=k,k+1,…N* ‘inci sütunları ile çarpılarak

$$l_{k1} \cdot u_{1j} + l_{k2} \cdot u_{2j} + \ldots + l_{kk} \cdot u_{kj} = a_{kj} \quad \rightarrow \quad \boxed{u_{kj} = \frac{1}{l_{kk}} \left(a_{kj} - \sum_{p=1}^{k-1} l_{kp} \cdot u_{pj} \right) \quad ; \quad j = k, k+1, \ldots, N}$$

L matrisinin *i=k,k+1,…,N* ‘inci satırları *U* matrisinin *k* ‘ıncı sütunuyla çarpılarak

$$l_{i1} \cdot u_{1k} + l_{i2} \cdot u_{2k} + \ldots + l_{ik} \cdot u_{kk} = a_{ik} \quad \rightarrow \quad \boxed{l_{ik} = \frac{1}{u_{kk}} \left(a_{ik} - \sum_{p=1}^{k-1} l_{ip} \cdot u_{pk} \right); \qquad i = k, k+1, \ldots, N}$$

Bu bağıntılar

k=1 için
$$\boxed{\begin{aligned} u_{1j} &= \frac{a_{1j}}{l_{11}} \quad ; \quad j = 1,2,\ldots,N \\ l_{i1} &= \frac{a_{i1}}{u_{11}} \; ; \qquad i = 1,2,\ldots,N \end{aligned}}$$

k=2 için

$$u_{2j} = \frac{1}{l_{22}}\left(a_{2j} - l_{21}\cdot u_{1j}\right) \quad ; \quad j = 2,3,\dots,N$$
$$l_{i2} = \frac{1}{u_{22}}\left(a_{i2} - l_{i1}\cdot u_{12}\right); \qquad i = 2,3,\dots,N$$

k=3 için

$$u_{3j} = \frac{1}{l_{33}}\left[a_{3j} - \left(l_{31}\cdot u_{1j} + l_{32}\cdot u_{2j}\right)\right] \quad ; \quad j = 3,4,\dots,N$$
$$l_{i3} = \frac{1}{u_{33}}\left[a_{i3} - \left(l_{i1}\cdot u_{13} + l_{i2}\cdot u_{23}\right)\right]; \qquad i = 3,4,\dots,N$$

olup, hesapların başlatılabilmesi ve sürdürülebilmesi için u_{kk} ve l_{kk} elemanlarından birinin önceden belirlenmesi gerektiği görülmektedir. Nitekim uygulamada bu elemanlardan birisinin değeri *1* olarak seçilir. Bu seçime bağlı olarak yöntem iki farklı isimle tanınmaktadır:

Doolittle Yöntemi

*Doolittle yöntemi*nde alt-üçgensel matrisin diyagonal elemanları *1* alınmakta olup, buna göre yukarıdaki üç adım

k=1 için

$$u_{1j} = a_{1j} \quad ; \quad j = 1,2,\dots,N$$
$$l_{11} = 1\ ; \quad l_{i1} = \frac{a_{i1}}{u_{11}}\ ; \qquad i = 2,3,\dots,N$$

k=2 için

$$u_{2j} = a_{2j} - l_{21}\cdot u_{1j} \quad ; \quad j = 2,3,\dots,N$$
$$l_{22} = 1\ ; \quad l_{i2} = \frac{1}{u_{22}}\left(a_{i2} - l_{i1}\cdot u_{12}\right); \qquad i = 3,4,\dots,N$$

k=3 için

$$u_{3j} = a_{3j} - \left(l_{31}\cdot u_{1j} + l_{32}\cdot u_{2j}\right) \quad ; \quad j = 3,4,\dots,N$$
$$l_{33} = 1\ ; \quad l_{i3} = \frac{1}{u_{33}}\left[a_{i3} - \left(l_{i1}\cdot u_{13} + l_{i2}\cdot u_{23}\right)\right]; \qquad i = 4,5,\dots,N$$

şekline gelir. Görüldüğü gibi hesaplamalarda bir sıra izlenerek herbir adımda önce üst-üçgensel matrisin bir satırının elemanlarının daha sonra da alt-üçgensel matrisin bir sütununun elemanlarının bulunması gerekmektedir.

Buna göre Doolittle yöntemi için genel algoritma aşağıdaki şekilde yazılabilir.

$$\left.\begin{array}{ll} u_{kj} = a_{kj} - \sum_{p=1}^{k-1} l_{kp}\cdot u_{pj}\ ; & j = k,k+1,\dots,N \\ l_{ik} = \dfrac{1}{u_{kk}}\left(a_{ik} - \sum_{p=1}^{k-1} l_{ip}\cdot u_{pk}\right); & i = k,k+1,\dots,N \end{array}\right. \quad k = 1,2,\dots,N$$

Crout Yöntemi

*Crout yöntemi*nde üst-üçgensel matrisin diyagonal elemanları *1* alınmakta olup, buna göre yukarıdaki üç adım

k=1 için

$$l_{i1} = a_{i1}\ ; \qquad i = 1,2,\dots,N$$
$$u_{11} = 1\ , \qquad u_{1j} = \frac{a_{1j}}{l_{11}} \quad ; \quad j = 2,3,\dots,N$$

k=2 için

$$\boxed{\begin{array}{l} l_{i2} = a_{i2} - l_{i1} \cdot u_{12} \ ; \qquad i = 2,3,\ldots,N \\ u_{22} = 1 \ , \qquad u_{2j} = \dfrac{1}{l_{22}}\left(a_{2j} - l_{21} \cdot u_{1j}\right) \quad ; \quad j = 3,4,\ldots,N \end{array}}$$

k=3 için

$$\boxed{\begin{array}{l} l_{i3} = a_{i3} - \left(l_{i1} \cdot u_{13} + l_{i2} \cdot u_{23}\right); \qquad i = 3,4,\ldots,N \\ u_{33} = 1 \ , \qquad u_{3j} = \dfrac{1}{l_{33}}\left[a_{3j} - \left(l_{31} \cdot u_{1j} + l_{32} \cdot u_{2j}\right)\right] \quad ; \quad j = 4,5,\ldots,N \end{array}}$$

şekline gelir. Görüldüğü gibi hesaplamalarda yine bir sıra izlenerek bu defa herbir adımda önce alt-üçgensel matrisin bir sütununun elemanlarının daha sonra da üst-üçgensel matrisin bir satırının elemanlarının bulunması gerekmektedir.

Buna göre Crout yöntemi için genel algoritma aşağıdaki şekilde yazılabilir.

$$\boxed{\left.\begin{array}{ll} l_{ik} = a_{ik} - \sum\limits_{p=1}^{k-1} l_{ip} \cdot u_{pk} \ ; & i = k, k+1,\ldots,N \\ u_{kj} = \dfrac{1}{l_{kk}}\left(a_{kj} - \sum\limits_{p=1}^{k-1} l_{kp} \cdot u_{pj}\right) \quad ; & j = k, k+1,\ldots,N \end{array}\right\} \ ; \qquad k = 1,2,\ldots,N}$$

İleri süpürme aşaması

$L \cdot Y = B$ denkleminin Y için çözümü ilk elemandan başlayarak ileri-süpürme yoluyla gerçekleştirilir.

$$\begin{bmatrix} l_{11} & 0 & 0 & 0 & \cdots & \cdots & 0 & \cdots & 0 \\ l_{21} & l_{22} & 0 & 0 & \cdots & \cdots & 0 & \cdots & 0 \\ l_{31} & l_{32} & l_{33} & 0 & \cdots & \cdots & 0 & \cdots & 0 \\ l_{41} & l_{42} & l_{33} & l_{44} & \cdots & \cdots & 0 & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ l_{i1} & l_{i2} & l_{i3} & l_{i4} & \cdots & l_{i,i-1} & l_{ii} & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ l_{N1} & l_{N2} & l_{N3} & l_{N4} & \cdots & l_{N,i-1} & l_{Ni} & \cdots & l_{NN} \end{bmatrix} \cdot \begin{Bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ \cdots \\ \cdots \\ y_i \\ \cdots \\ y_N \end{Bmatrix} = \begin{Bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ \cdots \\ \cdots \\ b_i \\ \cdots \\ b_N \end{Bmatrix}$$

L matrisinin birinci satırı ile Y vektörünün çarpımından

$$\boxed{y_1 = b_1 / l_{11}}$$

L matrisinin i 'inci satırının Y vektörü ile çarpımından

$$l_{i1} \cdot y_1 + l_{i2} \cdot y_2 + \ldots + l_{i,i-1} \cdot y_{i-1} + y_i = b_i \qquad \rightarrow \qquad \boxed{y_i = \frac{1}{l_{ii}}\left(b_i - \sum_{p=1}^{i-1} l_{ip} \cdot y_p\right); \qquad i = 2,3,\ldots N}$$

Geri süpürme aşaması

$U \cdot X = Y$ denkleminin X için çözümü de son elemandan başlayarak geri-süpürme yoluyla gerçekleştirilir.

$$\begin{bmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1i} & u_{1,i+1} & \cdots & u_{1,N-1} & u_{1,N} \\ 0 & u_{22} & u_{23} & \cdots & u_{2i} & u_{2,i+1} & \cdots & u_{2,N-1} & u_{2,N} \\ 0 & 0 & u_{33} & \cdots & u_{3i} & u_{3,i+1} & \cdots & u_{3,N-1} & u_{3,N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & u_{ii} & u_{i,i+1} & \cdots & u_{i,N-1} & u_{i,N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & u_{N-1,N-1} & u_{N-1,N} \\ 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & u_{N,N} \end{bmatrix} \cdot \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ \cdots \\ x_i \\ \cdots \\ \cdots \\ x_{N-1} \\ x_N \end{Bmatrix} = \begin{Bmatrix} y_1 \\ y_2 \\ y_3 \\ \cdots \\ y_i \\ \cdots \\ \cdots \\ y_{N-1} \\ y_N \end{Bmatrix}$$

U matrisinin sonuncu satırı X vektörüyle çarpılarak

$$u_{NN} \cdot x_N = y_N \qquad \rightarrow \qquad \boxed{x_N = y_N / u_{NN}}$$

U matrisinin i 'inci satırı X vektörüyle çarpılarak

$$u_{ii} \cdot x_i + u_{i,i+1} \cdot x_{i+1} + \ldots + u_{i,N} \cdot x_N = y_i \quad \rightarrow \quad \boxed{x_i = \frac{1}{u_{ii}}\left(y_i - p\sum_{k=i+1}^{N} u_{ip} \cdot x_p \right); \qquad i = N-1, N-2, \ldots, 2, 1}$$

elde edilir.

1.3.5- Jacobi basit iterasyon yöntemi

Çok büyük katsayılar matrisi içeren lineer denklem sistemlerinin eliminasyon yöntemleriyle çözümü çoğu zaman ekonomik olmaz. Bu gibi durumlarda iteratif yöntemler seçilir. Bunlardan en basit birisi Jacobi yöntemidir.

$$A \cdot X = B \tag{1}$$

lineer denklem takımı, A katsayılar matrisi

$$A = L + D + U \tag{2}$$

$$L = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ a_{21} & 0 & 0 & \cdots & 0 \\ a_{31} & a_{32} & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{N1} & a_{N2} & a_{N3} & \cdots & 0 \end{bmatrix}; D = \begin{bmatrix} a_{11} & 0 & 0 & \cdots & 0 \\ 0 & a_{22} & 0 & \cdots & 0 \\ 0 & 0 & a_{33} & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & a_{NN} \end{bmatrix}; U = \begin{bmatrix} 0 & a_{12} & a_{13} & \cdots & a_{1N} \\ 0 & 0 & a_{23} & \cdots & a_{2N} \\ 0 & 0 & 0 & \cdots & a_{3N} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}$$

şeklinde üç matrisin toplamı olmak üzere

$$L \cdot X + D \cdot X + U \cdot X = B \quad \rightarrow \quad X = D^{-1} \cdot [B - L \cdot X - U \cdot X] \tag{3}$$

şekline getirilebilir.

Bu durumda x_i bilinmeyenleri için uygun seçilecek başlangıç değerleri (3) eşitliğinde kullanılarak yeni x_i değerleri hesaplanabileceği ve bu işlemlerin iteratif olarak devam ettirilebileceği görülmektedir. Jacobi basit iterasyon yöntemi olarak bilinen bu yöntemin herhangi bir iterasyon adımı için kapalı formda

$$X^{(k+1)} = D^{-1} \cdot \left[B - L \cdot X^{(k)} - U \cdot X^{(k)}\right] \tag{4}$$

veya açık biçimde

$$\begin{Bmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ x_3^{(k+1)} \\ \cdots \\ x_N^{(k+1)} \end{Bmatrix} = D^{-1} \cdot \left(\begin{Bmatrix} b_1 \\ b_2 \\ b_3 \\ \cdots \\ b_N \end{Bmatrix} - \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ a_{21} & 0 & 0 & \cdots & 0 \\ a_{31} & a_{32} & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{N1} & a_{N2} & a_{N3} & \cdots & 0 \end{bmatrix} \begin{Bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ x_3^{(k)} \\ \cdots \\ x_N^{(k)} \end{Bmatrix} - \begin{bmatrix} 0 & a_{12} & a_{13} & \cdots & a_{1N} \\ 0 & 0 & a_{23} & \cdots & a_{2N} \\ 0 & 0 & 0 & \cdots & a_{3N} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \begin{Bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ x_3^{(k)} \\ \cdots \\ x_N^{(k)} \end{Bmatrix} \right) \qquad (5)$$

yazılabilir. *D* diyagonal matrisinin tersinin

$$D = \begin{bmatrix} a_{11} & 0 & 0 & \cdots & 0 \\ 0 & a_{22} & 0 & \cdots & 0 \\ 0 & 0 & a_{33} & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & a_{NN} \end{bmatrix} \quad \rightarrow \quad D^{-1} = \begin{bmatrix} 1/a_{11} & 0 & 0 & \cdots & 0 \\ 0 & 1/a_{22} & 0 & \cdots & 0 \\ 0 & 0 & 1/a_{33} & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 1/a_{NN} \end{bmatrix}$$

Şeklinde olacağı gösterilebilir. Bu durumda (5) matris eşitliğinin herhangi bir *i* 'inci satırı için

$$x_i^{(k+1)} = \frac{b_i - \sum_{j=1}^{i-1} a_{ij} \cdot x_j^{(k)} - \sum_{j=i+1}^{N} a_{ij} \cdot x_j^{(k)}}{a_{ii}} \quad ; \quad i = 1,2,\ldots.N \qquad (6)$$

yazılarak iterasyon algoritması açık biçimde elde edilebilir.

1.3.6- Gauss-Sidel iterasyon yöntemi

Jacobi yöntemi çabuk yakınsamayan bir yöntemdir. Yöntemin yakınsamasını hızlandırmak için her iterasyon adımında bir önceki çözümle yeni bulunan çözümlerin

$$X^{(k+1)} = D^{-1} \cdot \left[B - L \cdot X^{(k+1)} - U \cdot X^{(k)} \right] \qquad (7)$$

$$\begin{Bmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ x_3^{(k+1)} \\ \cdots \\ x_N^{(k+1)} \end{Bmatrix} = D^{-1} \cdot \left(\begin{Bmatrix} b_1 \\ b_2 \\ b_3 \\ \cdots \\ b_N \end{Bmatrix} - \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ a_{21} & 0 & 0 & \cdots & 0 \\ a_{31} & a_{32} & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{N1} & a_{N2} & a_{N3} & \cdots & 0 \end{bmatrix} \begin{Bmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ x_3^{(k+1)} \\ \cdots \\ x_N^{(k+1)} \end{Bmatrix} - \begin{bmatrix} 0 & a_{12} & a_{13} & \cdots & a_{1N} \\ 0 & 0 & a_{23} & \cdots & a_{2N} \\ 0 & 0 & 0 & \cdots & a_{3N} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \begin{Bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ x_3^{(k)} \\ \cdots \\ x_N^{(k)} \end{Bmatrix} \right) \qquad (8)$$

$$x_i^{(k+1)} = \frac{b_i - \sum_{j=1}^{i-1} a_{ij} \cdot x_j^{(k+1)} - \sum_{j=i+1}^{N} a_{ij} \cdot x_j^{(k)}}{a_{ii}} \quad ; \quad i = 1,2,\ldots.N \qquad (9)$$

şeklindeki bir kombinasyonundan yararlanmak mümkündür.

Görüldüğü gibi iterasyonun herhangi bir *k*'ıncı adımında ***X*** bilinmeyenler vektörünün *i* 'inci elemanı, x_i, hesaplanırken ***X*** 'in bu iterasyon adımında hesaplanmış olan ilk (*i*-*1*) elemanı ile bir önceki iterasyon adımında hesaplanmış olan (*i*+*1*) 'inci ve daha sonraki elemanları kullanılmaktadır.

1.3.7- Ardarda aşırı gevşetme (Successive over-relaxation) yöntemi

Gauss-Sidel iterasyon ifadesi, aynı terimin

$$x_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}\cdot x_j^{(k+1)} - \sum_{j=i+1}^{N} a_{ij}\cdot x_j^{(k)} - a_{ii}\cdot x_i^{(k)} + a_{ii}\cdot x_i^{(k)} \right) \quad ; \quad i = 1,2,....N \tag{10}$$

şeklinde bir defa ilave edilip bir defa da çıkartılması sonucunda

$$x_i^{(k+1)} = x_i^{(k)} + \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}\cdot x_j^{(k+1)} - \sum_{j=i}^{N} a_{ij}\cdot x_j^{(k)} \right) \quad ; \quad i = 1,2,....N \tag{11}$$

şeklinde düzenlenebilir. Buradaki

$$\overline{b}_i^{(k)} = \sum_{j=1}^{i-1} a_{ij}\cdot x_j^{(k+1)} - \sum_{j=i}^{N} a_{ij}\cdot x_j^{(k)} \quad ; \quad i = 1,2,....N \tag{12}$$

büyüklüğü, dikkat edilirse aslında *k* 'ıncı iterasyon adımında bulunan çözüm vektörünün katsayılar matrisi ile çarpımı olup, tam çözümün elde edilmesi halinde bu büyüklüğün b_i ye eşit olacağı açıktır. Ancak iterasyon sırasında çözümler tam çözümden farklı olacağından bu büyüklük de denklem sisteminin sağ taraf vektöründen farklı olacaktır. Aradaki fark

$$r_i^{(k)} = b_i - \overline{b}_i^{(k)} \quad ; \quad i = 1,2,....N \tag{13}$$

"kalıntı (residus)" olarak adlandırılır. Buna göre (11) bağıntısının sağındaki ikinci terim kalıntı terimi olup, bir önceki iterasyon adımında elde edilmiş çözümlere ilave edilen bir düzeltme terimi gibi yorumlanabilir:

$$x_i^{(k+1)} = x_i^{(k)} + \delta_i^{(k)} \quad ; \quad i = 1,2,....N \tag{14}$$

$$\delta_i^{(k)} = \frac{r_i^{(k)}}{a_{ii}} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}\cdot x_j^{(k+1)} - \sum_{j=i}^{N} a_{ij}\cdot x_j^{(k)} \right) \quad ; \quad i = 1,2,....N \tag{15}$$

Şimdi bu düzeltme terimi bir ω büyüklüğü ile çarpılarak etkisi azaltılabilir veya çoğaltılabilir.

$$x_i^{(k+1)} = x_i^{(k)} + \omega\cdot\delta_i^{(k)} \quad ; \quad i = 1,2,....N \tag{16}$$

$$x_i^{(k+1)} = x_i^{(k)} + \omega\cdot\frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}\cdot x_j^{(k+1)} - \sum_{j=i}^{N} a_{ij}\cdot x_j^{(k)} \right) \quad ; \quad i = 1,2,....N \tag{17}$$

Veya yeni bir düzenleme ile

$$x_i^{(k+1)} = x_i^{(k)} + \omega\cdot\frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}\cdot x_j^{(k+1)} - \sum_{j=i+1}^{N} a_{ij}\cdot x_j^{(k)} - a_{ii}\cdot x_i^{(k)} \right) \quad ; \quad i = 1,2,....N \tag{18}$$

$$x_i^{(k+1)} = x_i^{(k)} + \omega\cdot\left[\frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}\cdot x_j^{(k+1)} - \sum_{j=i+1}^{N} a_{ij}\cdot x_j^{(k)} \right) - x_i^{(k)} \right] \quad ; \quad i = 1,2,....N \tag{19}$$

$$x_i^{(k+1)} = \omega\cdot\frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}\cdot x_j^{(k+1)} - \sum_{j=i+1}^{N} a_{ij}\cdot x_j^{(k)} \right) + (1-\omega)x_i^{(k)} \quad ; \quad i = 1,2,....N \tag{20}$$

şekline getirilebilir.

Bu son ifade, görüldüğü gibi, herhangi bir iterasyon adımında Gauss-Sidel şemasıyla bulunan değerler ile bir önceki iterasyon adımında bulunan değerlerin kontrol edilebilen bir ağırlık oranında kombinasyonunu vermektedir. Buradaki ω gevşetme faktörü olup bu yöntem

$0 < \omega < 1$ için ardarda az gevşetme (Successive under relaxation)

$1 < \omega < 2$ için ardarda aşırı gevşetme (Successive over relaxation)

yöntemi olarak anılır.

1.4 Matris tersinin sayısal hesabı

Matrislerde bölme gibi bir işlem olmayıp, bu işlem matrisin tersi kullanılarak gerçekleştirilir.

$$\boxed{B/A=C} \quad \Rightarrow \quad \boxed{B \cdot A^{-1}=C}$$

Bu işlem aynı zamanda lineer denklem takımlarının çözümlenmesinde de kullanılabilir.

$$\boxed{A \cdot X=B} \quad \Rightarrow \quad A^{-1} A \cdot X=A^{-1}\, B \Rightarrow \quad \boxed{X=A^{-1}\, B}$$

Bir matrisin tersi eliminasyon yöntemleriyle lineer denklem takımlarının çözülmesine benzer şekilde gerçekleştirilebilir. Bir A matrisinin tersinin bulunması aslında

$$A\, A^{-1}=I_n$$

Şeklinde, sağ taraf vektörü birden fazla olan bir denklem sisteminin çözülmesine eşdeğerdir.

Buna göre katsayılar matrisinin sağ tarafına ilave sütunlarla bir birim matris ilave edilerek Gauss-eliminasyon yönytemiyle veya Gauss-Jordan eliminasyon yöntemiyle matrisin tersini elde etmek mümkündür. Daha önce izah edilen işlemlerden sonra birim matris ters matrise dönüşecektir.

Örnek:

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

Genişletilmiş matris
$$\left[\begin{array}{ccc|ccc} 1 & -1 & 2 & 1 & 0 & 0 \\ 3 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 2 & 0 & 0 & 1 \end{array}\right]$$

Gauss-Jordan eliminasyon yöntemiyle
$$\left[\begin{array}{ccc|ccc} 1 & 0 & 0 & 0 & 2/5 & -1/5 \\ 0 & 1 & 0 & -1 & 1 & 1 \\ 0 & 0 & 1 & 0 & -1/5 & 3/5 \end{array}\right]$$