CHAPTER 2

LINEAR SYSYEMS: FUNDAMENTALS AND DIRECT METHODS

SOLVED EXAMPLE PROBLEMS

for

NUMERICAL METHODS FOR SCIENTISTS AND ENGINEERS With Pseudocodes

By Zekeriya ALTAÇ

November 2024

EXAMPLE 2.1: Performing Matrix Algebra

Use the matrix $\mathbf{A}$ to evaluate the matrix equation: $\mathbf{A}^3 - 2\mathbf{A}^2 - 5\mathbf{A} + 20\,\mathbf{I}$.

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

SOLUTION:

We first evaluate $\mathbf{A}^2$:

$$\mathbf{A}^2 = \mathbf{AA} = \begin{bmatrix} 2 & -3 & 1 \\ -2 & 1 & 3 \\ 4 & -2 & -1 \end{bmatrix} \begin{bmatrix} 2 & -3 & 1 \\ -2 & 1 & 3 \\ 4 & -2 & -1 \end{bmatrix} = \begin{bmatrix} 14 & -11 & -8 \\ 6 & 1 & -2 \\ 8 & -12 & -1 \end{bmatrix},$$

Next, the matrix $\mathbf{A}^3$ yields

$$\mathbf{A}^3 = \mathbf{AA}^2 = \begin{bmatrix} 2 & -3 & 1 \\ -2 & 1 & 3 \\ 4 & -2 & -1 \end{bmatrix} \begin{bmatrix} 14 & -11 & -8 \\ 6 & 1 & -2 \\ 8 & -12 & -1 \end{bmatrix} = \begin{bmatrix} 18 & -37 & -11 \\ 2 & -13 & 11 \\ 36 & -34 & -27 \end{bmatrix}$$

Multiplication of matrices $\mathbf{A}$, $\mathbf{I}$, and $\mathbf{A}^2$, respectively, with 5, 20, and 2 are found as

$$5\mathbf{A} = \begin{bmatrix} 10 & -15 & 5 \\ -10 & 5 & 15 \\ 20 & -10 & -5 \end{bmatrix}, \quad 20\mathbf{I} = \begin{bmatrix} 20 & 0 & 0 \\ 0 & 20 & 0 \\ 0 & 0 & 20 \end{bmatrix}, \quad 2\mathbf{A}^2 = \begin{bmatrix} 28 & -22 & -16 \\ 12 & 2 & -4 \\ 16 & -24 & -2 \end{bmatrix}$$

Finally, substituting all the terms into the matrix equation and performing the element-by -element sum, we find

$$\mathbf{A}^3 + 2\mathbf{A}^2 + 5\mathbf{A} - 20\mathbf{I} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Discussion: A matrix equation involves matrix operations like addition, subtraction, multiplication with a scalar, or its powers. Matrix equations, typically, have a matrix (or vector) that is not known. Matrices are added or subtracted element-wise when they have the same dimensions. But matrix multiplication requires the dot product of rows of the first matrix with columns of the second.

EXAMPLE 2.2: Expressing Linear Systems in Matrix Form

Express the given system of linear equations in matrix equation form, **Ax** = **b**.

$$(a)\ \begin{cases} x + 2y - 3z = -5 \\ -5y + 3z = 13 \\ -z = -1 \end{cases}, \qquad (b)\ \begin{cases} u + 3v - w = 2 \\ 2v + 3u - 2w = 5 \\ w - 3u - 2v = 8 \end{cases} \qquad (c)\ \begin{cases} 4x_1 + 3x_2 = 6 \\ x_1 + 2x_2 + x_3 = 3 \\ x_2 + 5x_3 + 3x_4 = -11 \\ x_3 + 3x_4 + x_5 = -9 \\ -x_4 + 2x_5 + x_6 = 3 \\ x_5 + 4x_6 = -2 \end{cases}$$

SOLUTION:

To convert a system of linear equations into matrix form, we need to express it in terms of matrices and vectors. To do that, make sure the variables (e.g., x_1, x_2, x_3, and so on) are in the same order in each equation:

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \qquad \begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \ \vdots \qquad \vdots \qquad \cdots \qquad \vdots \ = \ \vdots \\ a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \end{cases}$$

As shown below, the coefficient matrices emerge when given linear systems are rearranged so that each unknown is placed in the same column:

$$(a)\ \begin{cases} x + 2y - 3z = -5 \\ 0\,x - 5y + 3z = 13 \\ 0\,x + 0\,y - z = -1 \end{cases} \Rightarrow \qquad \begin{bmatrix} 1 & 2 & -3 \\ 0 & -5 & 3 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -5 \\ 13 \\ -1 \end{bmatrix}$$

$$(b)\ \begin{cases} u + 3v - w = 2 \\ 3u + 2v - 2w = 5 \\ -3u - 2v + w = 8 \end{cases} \Rightarrow \qquad \begin{bmatrix} 1 & 3 & -1 \\ 3 & 2 & -2 \\ -3 & -2 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} 2 \\ 5 \\ 8 \end{bmatrix}$$

$$(c)\ \begin{cases} 4x_1 + 3x_2 = 6 \\ x_1 + 2x_2 + x_3 = 3 \\ x_2 + 5x_3 + 3x_4 = -11 \\ x_3 + 3x_4 + x_5 = -9 \\ -x_4 + 2x_5 + x_6 = 3 \\ x_5 + 4x_6 = -2 \end{cases} \Rightarrow \qquad \begin{bmatrix} 4 & 3 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 0 \\ 0 & 1 & 5 & 3 & 0 & 0 \\ 0 & 0 & 1 & 3 & 1 & 0 \\ 0 & 0 & 0 & -1 & 2 & 1 \\ 0 & 0 & 0 & 0 & 1 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} = \begin{bmatrix} 7 \\ 7 \\ 19 \\ 17 \\ -14 \\ -5 \end{bmatrix}$$

Discussion: When putting a system of linear equations into matrix form, there are a few important points to keep in mind to ensure accuracy and avoid errors:

1. Consistent Ordering of Variables: Make sure the variables (x, y, and z, so on) are in the same order in each equation. If the order changes, the resulting coefficient matrix will be incorrect. For example, if the system involves x, y, and z, arrange all rows in the order x, y, and z without swapping columns.

2. Correctly Forming the Coefficient Matrix: Each entry in the coefficient matrix **A** should correspond precisely to the coefficient of a variable in that row's equation. If a variable is missing in an equation, use a zero ('0') as the coefficient for that position. This ensures that the matrix has the correct dimensions and structure.

3. Consistency in Dimensions: For a system with m equations and n variables (*under-determined* or *over-determined* systems), the coefficient matrix **A** should be of size $m \times n$, the variable vector **x** should be $n \times 1$, and the constant vector **b** should be $m \times 1$.

EXAMPLE 2.3: Evaluating Vector Norms

Evaluate 1-, 2-, and ∞-norms of these vectors:

$$(a)\ \mathbf{x} = [-4\ \ 2\ \ 1\ \ -3\ \ 1], \qquad (b)\ \mathbf{y} = [1\ \ 2\ \ -3\ \ -7], \qquad (c)\ \mathbf{z} = [1\ \ -3\ \ 1\ \ 1]$$

SOLUTION:

A vector norm is a function that assigns a non-negative length or size to a vector in a vector space. Norms are widely used to measure distances, magnitudes, and sizes of vectors, and they have various applications in mathematics, physics, machine learning, and data science.

The ℓ_1 norm of a vector $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ is the sum of the absolute values of its components. The ℓ_2 norm of a vector is the most common vector norm, representing the Euclidean distance from the origin to the point $\mathbf{x}$. The infinity norm (ℓ_∞) is the maximum absolute value of the vector's components. Accordingly, the norms of the given vectors are found as follows:

$$\begin{array}{llll} (a) & \|\mathbf{x}\|_1 = 11, & \|\mathbf{x}\|_2 = \sqrt{31}, & \|\mathbf{x}\|_\infty = 4. \\ (b) & \|\mathbf{y}\|_1 = 13, & \|\mathbf{y}\|_2 = 3\sqrt{7}, & \|\mathbf{y}\|_\infty = 7. \\ (c) & \|\mathbf{z}\|_1 = 6, & \|\mathbf{z}\|_2 = \sqrt{12}, & \|\mathbf{z}\|_\infty = 3. \end{array}$$

Discussion: Vector norms are used across various fields in mathematics, physics, engineering, computer science, and data science to measure the *size* or *magnitude* of vectors in different ways. Here are some of the key applications and contexts where vector norms play an essential role:

1. Measuring Distance and Similarity: *Distance between Points*: In geometry and computer science, norms (especially the ℓ_2 norm) measure the distance between points. For example, the Euclidean distance between two vectors $\mathbf{x}$ and $\mathbf{y}$ is given by $\|\mathbf{x} - \mathbf{y}\|$.

Similarity in Machine Learning: In machine learning, norms are used to compute distances in feature space to determine the similarity between data points. For instance, clustering algorithms use the ℓ_2 norm to assign points to clusters.

2. Optimization and Regularization: Norms help to regularize models by penalizing large values of parameters, thereby reducing overfitting. Norms also define boundaries within which a solution must lie. For example, the optimization problem may require that the solution has a bounded ℓ_2 norm.

3. Error Measurement in Numerical Analysis In numerical analysis, norms measure the error between an approximate solution and the true solution. The ℓ_2 and ℓ_∞ norms are often used to quantify errors in solutions of differential equations or approximations in iterative methods.

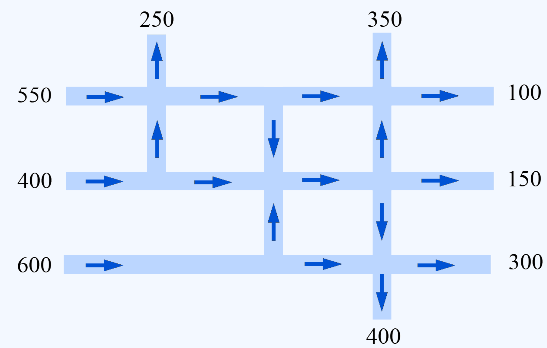
4. Machine Learning and Deep Learning In gradient-based optimization, norms are used to control the size of parameter updates. For instance, the gradient norm can be limited to prevent large updates, which helps avoid instability. Data preprocessing also involves normalizing feature vectors to have unit norms, improving model training stability, and ensuring features have similar importance. In distance-based models and support vector machines, norms measure distances in feature space to classify or cluster data points effectively.

5. Geometric Interpretation and Transformations: Norms provide a way to quantify the *length* of a vector in any direction, allowing for easy scaling and transformation of vectors.

Each norm has unique characteristics suited to specific tasks, and choosing the right norm depends on the problem requirements—whether it emphasizes sparsity, smoothness, or robustness to outliers.

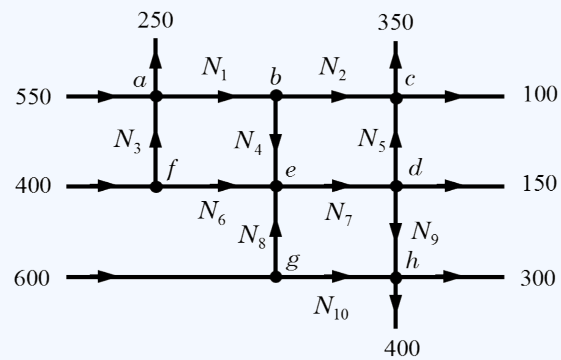
EXAMPLE 2.4: Solving An Under-determined System of Equations

The traffic patterns for a district are shown below, where indicated is the vehicles per hour (vph). Assuming the total flow into a junction is equal to the total flow out of the junction. Obtain the system of flow equations for the district in question and find the possible flow rates along each street.



SOLUTION:

(a) To find the traffic pattern for this district, we label the directions and junctions as follows:



Junction	Inflow		Outflow	Equilibrium Equations
(a)	$N_3 + 550$	=	$N_1 + 250$	$N_1 - N_3 = 300$
(b)	N_1	=	$N_2 + N_4$	$N_1 - N_2 - N_4 = 0$
(c)	$N_2 + N_5$	=	450	$N_2 + N_5 = 450$
(d)	N_7	=	$N_5 + N_9 + 150$	$-N_5 + N_7 - N_9 = 150$
(e)	$N_4 + N_6 + N_8$	=	N_7	$N_4 + N_6 - N_7 + N_8 = 0$
(f)	400	=	$N_3 + N_6$	$N_3 + N_6 = 400$
(g)	600	=	$N_8 + N_{10}$	$N_8 + N_{10} = 600$
(h)	$N_9 + N_{10}$	=	700	$N_9 + N_{10} = 700$

The augmented matrix can be expressed as follows:

$$\begin{bmatrix} 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 300 \\ 1 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 450 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & -1 & 0 & 150 \\ 0 & 0 & 0 & 1 & 0 & 1 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 400 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 600 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 700 \end{bmatrix}$$

This is an underdetermined system. We perform the following row operations sequentially: (1) $r_2 \leftarrow r_2 - r_1$, (2) $r_2 \leftarrow r_2 + r_3$, $r_2 \leftrightarrow r_3$, (3) $r_1 \leftarrow r_1 + r_3$, $r_4 \leftarrow r_4 - r_3$, (4) $r_1 \leftarrow r_1 + r_4$, $r_3 \leftarrow r_3 + r_4$, (5) $r_2 \leftarrow r_2 - r_5$, $r_4 \leftarrow r_4 + r_5$, $r_6 \leftarrow r_6 - r_5$, which yields

$$\sim \left[\begin{array}{cccccc ccc c r} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 700 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1300 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 400 \\ 0 & 0 & 0 & 1 & 0 & 1 & -1 & 0 & 0 & -1 & -600 \\ 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & -1 & -850 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 600 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 700 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right]$$

or

$$\begin{array}{lll} N_1 + N_6 = 700, & N_2 + N_7 + N_{10} = 1300, & N_3 + N_6 = 400, \\ -N_4 - N_6 + N_7 + N_{10} = 600, & -N_5 + N_7 + N_{10} = 850, & N_8 + N_{10} = 600, \\ N_9 + N_{10} = 700. & & \end{array}$$

This implies that N_6, N_7, and N_{10} are independent (or free) variables. It should be noted that $N_k \geqslant 0$ for all k. In this regard, we can easily deduce the following: $N_6 \leqslant 250$, $N_{10} \leqslant 600$, and $850 \leqslant N_7 + N_{10} \leqslant 1300$ provided that $250 \leqslant N_7 \leqslant 700$.

Setting $N_6 = \alpha$, $N_7 = \beta$, and $N_{10} = \gamma$, the general solution becomes $N_1 = 700 - \alpha$, $N_2 = 1300 - \beta - \gamma$, $N_3 = 400 - \alpha$, $N_4 = \beta + \gamma - \alpha - 600$, $N_5 = \beta + \gamma - 850$, $N_8 = 600 - \gamma$, and $N_9 = 700 - \gamma$. Depending on the values α, β, and γ, the traffic pattern varies.

Discussion: In an underdetermined system, there are more variables than there are independent equations. This usually results in an infinite number of possible solutions, as there are not enough constraints to determine a unique solution. A common approach is to express the solution in terms of free parameters, as we have done so in this example. This approach gives one a solution set that depends on one or more free variables. To narrow down to a solution, additional criteria or constraints will be required.

EXAMPLE 2.5: Matrix Inversion with Gauss-Jordan Method

Invert the following 3×3 matrix using *Gauss-Jordan* method.

$$\begin{bmatrix} -15 & -19 & 11 \\ 6 & 7 & -4 \\ -4 & -5 & 3 \end{bmatrix}$$

SOLUTION:

First we construct the augmented (**AI**) matrix:

$$\left[\begin{array}{ccc|ccc} -15 & -19 & 11 & 1 & 0 & 0 \\ 6 & 7 & -4 & 0 & 1 & 0 \\ -4 & -5 & 3 & 0 & 0 & 1 \end{array}\right]$$

We aim to create (1,0,0) in the first column through suitable elementary row operations. In this regard, $r_1 \leftarrow r_1 - 4r_3$ operation generates '1' at (1,1) position:

$$\approx \left[\begin{array}{ccc|ccc} 1 & 1 & -1 & 1 & 0 & -4 \\ 6 & 7 & -4 & 0 & 1 & 0 \\ -4 & -5 & 3 & 0 & 0 & 1 \end{array}\right] \begin{array}{l} \leftarrow r_1 - 4r_3 \\ \\ \\ \end{array}$$

Next, with $r_2 \leftarrow r_2 - 6r_1$ and $r_3 \leftarrow r_3 + 4r_1$, the second and third row elements are zeroed out:

$$\approx \left[\begin{array}{ccc|ccc} 1 & 1 & -1 & 1 & 0 & -4 \\ 0 & 1 & 2 & -6 & 1 & 24 \\ 0 & -1 & -1 & 4 & 0 & -15 \end{array}\right] \begin{array}{l} \\ \leftarrow r_2 - 6r_1 \\ \leftarrow r_3 + 4r_1 \end{array}$$

Since the 2nd element of column 2 is already '1', we eliminate the first and last element of the second column with $r_1 \leftarrow r_1 - r_2$ and $r_3 \leftarrow r_3 + r_2$:

$$\approx \left[\begin{array}{ccc|ccc} 1 & 0 & -3 & 7 & -1 & -28 \\ 0 & 1 & 2 & -6 & 1 & 24 \\ 0 & 0 & 1 & -2 & 1 & 9 \end{array}\right] \begin{array}{l} \leftarrow r_1 - r_2 \\ \\ \leftarrow r_3 + r_2 \end{array}$$

Since the 3rd element of column 3 is already '1', we do not need to normalize it. We can eliminate the 1st and 2nd elements of the 3rd column with $r_1 \leftarrow r_1 + 3r_3$ and $r_2 \leftarrow r_2 - 2r_3$:

$$\approx \left[\begin{array}{ccc|ccc} 1 & 0 & 0 & 1 & 2 & -1 \\ 0 & 1 & 0 & -2 & -1 & 6 \\ 0 & 0 & 1 & -2 & 1 & 9 \end{array}\right] \begin{array}{l} \leftarrow r_1 + 3r_3 \\ \leftarrow r_2 - 2r_3 \\ \\ \end{array}$$

Now, the left side of the augmented matrix is an identity matrix. The one the right is the inverse matrix; that is,

$$\begin{bmatrix} -15 & -19 & 11 \\ 6 & 7 & -4 \\ -4 & -5 & 3 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 2 & -1 \\ -2 & -1 & 6 \\ -2 & 1 & 9 \end{bmatrix}$$

Discussion: The inverse of a matrix is required in linear algebra, engineering, computer science, economics, and physics to solve equations, analyze systems, and perform transformations. While matrix inverses have many uses, it's essential to note that not all matrices have inverses.

EXAMPLE 2.6: Solving System of Equations with Gauss-Jordan Method

Consider the following ordinary differential equation (ODE):

$$x^4U'''' + 6x^3U''' + 2x^2U'' - 4xU' + 4U = 0, \quad \tfrac{1}{2} \leqslant x \leqslant 2$$

which has a general solution $U(x) = C_1/x^2 + C_2/x + x\,C_3 + x^2\,C_4$ where C_1, C_2, C_3, and C_4 are undetermined coefficients. An analytical solution of this ODE subjected to $U(1/2) = U(2) = 1$ and $U'(1/2) = U'(2) = 0$ conditions is sought. Obtain the linear system for the coefficients and then solve it using the Gauss-Jordan elimination method.

SOLUTION:

Using the boundary conditions, $U(1/2) = U(2) = 1$ and $U'(1/2) = U'(2) = 0$ conditions, we arrive at the following system of linear equations, which can be expressed as a matrix equation:

$$\left\{\begin{aligned} &4C_1 + 2C_2 + \tfrac{1}{2}C_3 + \tfrac{1}{4}C_4 = 1 \\ &\tfrac{1}{4}C_1 + \tfrac{1}{2}C_2 + 2C_3 + 4C_4 = 1 \\ &-16C_1 - 4C_2 + C_3 + C_4 = 0, \\ &-\tfrac{1}{4}C_1 - \tfrac{1}{4}C_2 + C_3 + 4C_4 = 0 \end{aligned}\right\} \Rightarrow \begin{bmatrix} 4 & 2 & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{2} & 2 & 4 \\ -16 & -4 & 1 & 1 \\ -\frac{1}{4} & -\frac{1}{4} & 1 & 4 \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

We begin with normalizing the first element of the 1st column. Next, we apply the elementary row operations indicated on the far right side to zero out the 2nd, 3rd, and 4th elements.

$$\approx \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{8} & \frac{1}{16} & \frac{1}{4} \\ 0 & \frac{3}{8} & \frac{63}{32} & \frac{255}{64} & \frac{15}{16} \\ 0 & 4 & 3 & 2 & 4 \\ 0 & -\frac{1}{8} & \frac{33}{32} & \frac{257}{64} & \frac{1}{16} \end{bmatrix} \begin{matrix} r'_1 \leftarrow \frac{1}{4}r_1 \\ r'_2 \leftarrow r_2 - \frac{1}{4}r'_1 \\ r'_3 \leftarrow r_3 + 4r_1 \\ r'_4 \leftarrow r_4 + \frac{1}{4}r'_1 \end{matrix}$$

To work on the second column, we normalize the 2nd element of the 2nd column. Then, we similarly apply the elementary row operations indicated on the far right side to zero out the 1st, 3rd, and 4th elements.

$$\approx \begin{bmatrix} 1 & 0 & -\frac{5}{2} & -\frac{21}{4} & -1 \\ 0 & 1 & \frac{21}{4} & \frac{85}{8} & \frac{5}{2} \\ 0 & 0 & -18 & -\frac{81}{2} & -6 \\ 0 & 0 & \frac{27}{16} & \frac{171}{32} & \frac{3}{8} \end{bmatrix} \begin{matrix} r'_1 \leftarrow r_1 - \frac{1}{2}r'_2 \\ r'_2 \leftarrow \frac{8}{3}r'_2 \\ r'_3 \leftarrow r_3 - 4r_2 \\ r'_4 \leftarrow r_4 + \frac{1}{8}r'_2 \end{matrix}$$

As we continue with the 3rd and 4th columns, we first normalize the diagonal element, if necessary. Then, the other elements of the column are zeroed out. Finally, when the left side of the augmented matrix becomes an identity matrix, the last column is the solution vector.

$$\approx \begin{bmatrix} 1 & 0 & 0 & \frac{3}{8} & -\frac{1}{6} \\ 0 & 1 & 0 & -\frac{19}{16} & \frac{3}{4} \\ 0 & 0 & 1 & \frac{9}{4} & \frac{1}{3} \\ 0 & 0 & 0 & \frac{99}{64} & -\frac{3}{16} \end{bmatrix} \begin{matrix} r'_1 \leftarrow r_1 + \frac{5}{2}r_3 \\ r'_2 \leftarrow r_2 - \frac{21}{4}r'_3 \\ r'_3 \leftarrow -\frac{1}{18}r_3 \\ r'_4 \leftarrow r_4 - \frac{27}{16}r'_3 \end{matrix} \approx \begin{bmatrix} 1 & & & & -\frac{4}{33} \\ & 1 & & & \frac{20}{33} \\ & & 1 & & \frac{20}{33} \\ & & & 1 & -\frac{4}{33} \end{bmatrix} \begin{matrix} r'_1 \leftarrow r_1 - \frac{3}{8}r'_4 \\ r'_2 \leftarrow r_2 + \frac{19}{16}r'_4 \\ r'_3 \leftarrow r_3 - \frac{9}{4}r_3 \\ r'_4 \leftarrow \frac{64}{99}r_4 \end{matrix}$$

Thus, we find $C_1 = C_4 = -4/33$, $C_2 = C_3 = 20/33$, yielding $U(x) = 4(5x+5x^3-1-x^4))/(33x^2)$.

Discussion: Linear systems are encountered in many areas across science, engineering, economics, computer science, and even social sciences. Their solution with the Gauss-Jordan method is pretty straight forward.

EXAMPLE 2.7: Solving Tridiagonal System of Equations with Pivoting

Use the Gauss-elimination method with *partial pivoting* to solve the following tridiagonal system of linear equations:

$$\begin{bmatrix} 1 & 3 & 0 & 0 & 0 & 0 \\ 3 & 4 & -3 & 0 & 0 & 0 \\ 0 & 3 & 4 & -3 & 0 & 0 \\ 0 & 0 & 3 & 5 & -4 & 0 \\ 0 & 0 & 0 & 3 & 5 & -4 \\ 0 & 0 & 0 & 0 & 3 & 5 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{bmatrix} = \begin{bmatrix} 7 \\ 2 \\ 9 \\ 16 \\ 15 \\ 11 \end{bmatrix}$$

SOLUTION:

We start by constructing the augmented matrix. Noting that the largest element in column 1 is 3, we swap 1st and 2nd rows (i.e., $r_2 \rightleftarrows r_1$):

$$\begin{bmatrix} 1 & 3 & 0 & 0 & 0 & 0 & 7 \\ 3 & 4 & -3 & 0 & 0 & 0 & 2 \\ 0 & 3 & 4 & -3 & 0 & 0 & 9 \\ 0 & 0 & 3 & 5 & -4 & 0 & 16 \\ 0 & 0 & 0 & 3 & 5 & -4 & 15 \\ 0 & 0 & 0 & 0 & 3 & 5 & 11 \end{bmatrix} \sim \begin{bmatrix} 3 & 4 & -3 & 0 & 0 & 0 & 2 \\ 1 & 3 & 0 & 0 & 0 & 0 & 7 \\ 0 & 3 & 4 & -3 & 0 & 0 & 9 \\ 0 & 0 & 3 & 5 & -4 & 0 & 16 \\ 0 & 0 & 0 & 3 & 5 & -4 & 15 \\ 0 & 0 & 0 & 0 & 3 & 5 & 11 \end{bmatrix} r_2 \rightleftarrows r_1$$

After zeroing out the element below diagonal, we swap the 2nd and 3rd rows to make the maximum of the 2nd column as the pivot element:

$$\sim \begin{bmatrix} 3 & 4 & -3 & 0 & 0 & 0 & 2 \\ 0 & \frac{5}{3} & 1 & 0 & 0 & 0 & \frac{19}{3} \\ 0 & 3 & 4 & -3 & 0 & 0 & 9 \\ 0 & 0 & 3 & 5 & -4 & 0 & 16 \\ 0 & 0 & 0 & 3 & 5 & -4 & 15 \\ 0 & 0 & 0 & 0 & 3 & 5 & 11 \end{bmatrix} \leftarrow r_2 + (-\tfrac{1}{3})\, r_1 \quad \sim \begin{bmatrix} 3 & 4 & -3 & 0 & 0 & 0 & 2 \\ 0 & 3 & 4 & -3 & 0 & 0 & 9 \\ 0 & \frac{5}{3} & 1 & 0 & 0 & 0 & \frac{19}{3} \\ 0 & 0 & 3 & 5 & -4 & 0 & 16 \\ 0 & 0 & 0 & 3 & 5 & -4 & 15 \\ 0 & 0 & 0 & 0 & 3 & 5 & 11 \end{bmatrix} r_2 \rightleftarrows r_3$$

We will continue zeroing out the element below the pivot and make the maximum element (in absolute value sense) in the column the pivot.

$$\sim \begin{bmatrix} 3 & 4 & -3 & 0 & 0 & 0 & 2 \\ 0 & 3 & 4 & -3 & 0 & 0 & 9 \\ 0 & 0 & -\frac{11}{9} & \frac{5}{3} & 0 & 0 & \frac{4}{3} \\ 0 & 0 & 3 & 5 & -4 & 0 & 16 \\ 0 & 0 & 0 & 3 & 5 & -4 & 15 \\ 0 & 0 & 0 & 0 & 3 & 5 & 11 \end{bmatrix} \leftarrow r_3 + (-\tfrac{5}{9}) r_2 \sim \begin{bmatrix} 3 & 4 & -3 & 0 & 0 & 0 & 2 \\ 0 & 3 & 4 & -3 & 0 & 0 & 9 \\ 0 & 0 & 3 & 5 & -4 & 0 & 16 \\ 0 & 0 & -\frac{11}{9} & \frac{5}{3} & 0 & 0 & \frac{4}{3} \\ 0 & 0 & 0 & 3 & 5 & -4 & 15 \\ 0 & 0 & 0 & 0 & 3 & 5 & 11 \end{bmatrix} r_3 \rightleftarrows r_4$$

The element below the diagonal of the 3rd row is zeroed out by applying the $r_4 + (11/27) r_3$ elementary operation:

$$\sim \begin{bmatrix} 3 & 4 & -3 & 0 & 0 & 0 & 2 \\ 0 & 3 & 4 & -3 & 0 & 0 & 9 \\ 0 & 0 & 3 & 5 & -4 & 0 & 16 \\ 0 & 0 & 0 & \frac{100}{27} & -\frac{44}{27} & 0 & \frac{212}{27} \\ 0 & 0 & 0 & 3 & 5 & -4 & 15 \\ 0 & 0 & 0 & 0 & 3 & 5 & 11 \end{bmatrix} \leftarrow r_4 + (\tfrac{11}{27}) r_3$$

Next, we swap 5th and 4th rows (i.e., $r_4 \rightleftarrows r_5$) since the largest element in column 4 is 3:

$$\sim \begin{bmatrix} 3 & 4 & -3 & 0 & 0 & 0 & 2 \\ 0 & 3 & 4 & -3 & 0 & 0 & 9 \\ 0 & 0 & 3 & 5 & -4 & 0 & 16 \\ 0 & 0 & 0 & 3 & 5 & -4 & 15 \\ 0 & 0 & 0 & \frac{100}{27} & -\frac{44}{27} & 0 & \frac{212}{27} \\ 0 & 0 & 0 & 0 & 3 & 5 & 11 \end{bmatrix} r_4 \rightleftarrows r_5$$

Now, we zero out the element 100/27 in the 5th row as follows:

$$\sim \begin{bmatrix} 3 & 4 & -3 & 0 & 0 & 0 & 2 \\ 0 & 3 & 4 & -3 & 0 & 0 & 9 \\ 0 & 0 & 3 & 5 & -4 & 0 & 16 \\ 0 & 0 & 0 & 3 & 5 & -4 & 15 \\ 0 & 0 & 0 & 0 & -\frac{632}{81} & \frac{400}{81} & -\frac{32}{3} \\ 0 & 0 & 0 & 0 & 3 & 5 & 11 \end{bmatrix} \leftarrow r_5 + (-\tfrac{100}{81})r_4$$

And, as required for pivoting, we swap 5th and 6th rows to give:

$$\sim \begin{bmatrix} 3 & 4 & -3 & 0 & 0 & 0 & 2 \\ 0 & 3 & 4 & -3 & 0 & 0 & 9 \\ 0 & 0 & 3 & 5 & -4 & 0 & 16 \\ 0 & 0 & 0 & 3 & 5 & -4 & 15 \\ 0 & 0 & 0 & 0 & 3 & 5 & 11 \\ 0 & 0 & 0 & 0 & -\frac{632}{81} & \frac{400}{81} & -\frac{32}{3} \end{bmatrix} r_5 \rightleftarrows r_6$$

Finally, we zero out the element -632/81 in the last row to yield:

$$\sim \begin{bmatrix} 3 & 4 & -3 & 0 & 0 & 0 & 2 \\ 0 & 3 & 4 & -3 & 0 & 0 & 9 \\ 0 & 0 & 3 & 5 & -4 & 0 & 16 \\ 0 & 0 & 0 & 3 & 5 & -4 & 15 \\ 0 & 0 & 0 & 0 & 3 & 5 & 11 \\ 0 & 0 & 0 & 0 & 0 & \frac{4360}{243} & \frac{4360}{243} \end{bmatrix} \leftarrow r_6 + (\tfrac{632}{243})r_5$$

and normalizing the last row leads to

$$\sim \begin{bmatrix} 3 & 4 & -3 & 0 & 0 & 0 & 2 \\ 0 & 3 & 4 & -3 & 0 & 0 & 9 \\ 0 & 0 & 3 & 5 & -4 & 0 & 16 \\ 0 & 0 & 0 & 3 & 5 & -4 & 15 \\ 0 & 0 & 0 & 0 & 3 & 5 & 11 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \leftarrow (\tfrac{243}{4360})r_6 \qquad \Rightarrow \begin{Bmatrix} 3y_1 + 4y_2 - 3y_3 = 2 \\ 3y_2 + y_3 = 9 \\ 8y_3 - 4y_5 = 16 \\ 3y_3 + 5y_5 - 4y_6 = 15 \\ 3y_5 + 5y_6 = 11 \\ y_6 = 1 \end{Bmatrix}$$

When the back-substitution algorithm is employed to get the unknowns from last (y_6) to the first (y_1), we find $\{y_1, y_2, y_3, y_4, y_5, y_6\} = \{1, 2, 3, 3, 2, 1\}$.

Discussion: Pivoting requires rearranging the rows of a matrix to ensure that the largest possible coefficients are used as pivot elements during each elimination step. Pivoting reduces round-off errors, improves numerical stability, and avoids division by zero. That is why pivoting is generally recommended for most practical applications of Gaussian elimination, especially in computer-based implementations.

EXAMPLE 2.8: Solving Overdetermined Systems of Linear Equations

A magic square is defined as a square divided into smaller squares each containing a number, such that sums of the numbers in each vertical, horizontal, and diagonal row add up to the same value. Using the matrix concept, construct a 3×3 magic square whose rows, columns, and diagonals add up to 30. This provides 8 equations, which implies we will have one or two free variables. Using the proposed values for these variables establish the augmented matrix. Apply the Gauss-Jordan method to solve the system for the unknowns.

a_{11}	a_{12}	a_{13}
a_{21}	a_{22}	a_{23}
a_{31}	a_{32}	a_{33}

SOLUTION:

We have a total of 8 equations: $a_{11}+a_{12}+a_{13} = 30$, $a_{21}+a_{22}+a_{23} = 30$, $a_{31}+a_{32}+a_{33} = 30$, and so on. However, we have 9 unknowns. In other words, we will be dealing with an underdetermined linear system, where there are fewer equations than unknowns. The system of equations can be expressed in matrix form as

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}_{8\times 9} \begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \\ a_{21} \\ a_{22} \\ a_{23} \\ a_{31} \\ a_{32} \\ a_{33} \end{bmatrix}_{9\times 1} = \begin{bmatrix} 30 \\ 30 \\ 30 \\ 30 \\ 30 \\ 30 \\ 30 \\ 30 \end{bmatrix}_{8\times 1}$$

Employing row reduction leads to

$$\left[\begin{array}{ccccccccc|c} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 20 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 20 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & -1 & -10 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & -2 & -20 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 10 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 2 & 40 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 30 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \Rightarrow \begin{aligned} a_{11} + a_{33} &= 20 \\ a_{12} + a_{32} &= 20 \\ a_{13} - a_{32} - a_{33} &= -10 \\ a_{21} - a_{32} - 2a_{33} &= -20 \\ a_{22} &= 10 \\ a_{23} + a_{32} + 2a_{33} &= 40 \\ a_{31} + a_{32} + a_{33} &= 30 \end{aligned}$$

The solution is obtained dependent linearly on two variables, namely a_{23} and a_{33}. Choosing arbitrarily $a_{23} = 5$ and $a_{33} = 9$, we find

11	15	4
3	10	17
16	5	9

Discussion: Magic squares are often studied in recreational mathematics and number theory and have appeared in various cultures and art forms. As we have seen in this example, a magic square may not be unique, i.e., there may be several solutions depending on the arbitrary values adopted.

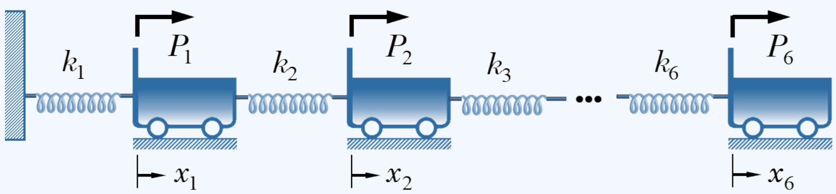
EXAMPLE 2.9: Solving Tridiagonal System of Linear Equations

Consider the equilibrium state of a system of interconnected six trolleys that are subjected to loads (k =1,2,...,6) as shown in the figure below: The displacements of the trolleys are governed by the equilibrium equations:

$$\begin{aligned}
&P_1 - k_1x_1 + k_2(x_2 - x_1) = 0\\
&P_2 - k_2(x_2 - x_1) + k_3(x_3 - x_2) = 0\\
&P_3 - k_3(x_3 - x_2) + k_4(x_4 - x_3) = 0\\
&P_4 - k_4(x_4 - x_3) + k_5(x_5 - x_3) = 0\\
&P_5 - k_5(x_5 - x_4) + k_6(x_6 - x_6) = 0\\
&P_6 - k_6(x_6 - x_5) = 0
\end{aligned}$$

where x and k denote displacement and spring constant, respectively. (a) Express the equilibrium equations in **Ax=b** form; (b) use the data given to find the displacements using the *Thomas algorithm*. Use the given data:

$$k_1 = 625\,\text{N/m},\; k_2 = 1250\text{N/m},\; k_3 = k_4 = 5000\,\text{N/m},\; k_5 = 3400\,\text{N/m},\; k_6 = 1000\,\text{N/m},$$
$$P_1 = 500\,\text{N},\, P_2 = -875\,\text{N},\; P_3 = -1000\,\text{N},\; P_4 = 3200\,\text{N},\, P_5 = -2150\,\text{N},\; P_6 = 450\,\text{N}$$



SOLUTION:

Substituting the data and rearranging the equilibrium equations gives the following system:

$$\begin{bmatrix} 1875 & -1250 & 0 & 0 & 0 & 0\\ -1250 & 6250 & -5000 & 0 & 0 & 0\\ 0 & -5000 & 10000 & -5000 & 0 & 0\\ 0 & 0 & -5000 & 8400 & -3400 & 0\\ 0 & 0 & 0 & -3400 & 4400 & -1000\\ 0 & 0 & 0 & 0 & -1000 & 1000 \end{bmatrix}\begin{bmatrix} x_1\\ x_2\\ x_3\\ x_4\\ x_5\\ x_6\end{bmatrix} = \begin{bmatrix} 500\\ -875\\ -1000\\ 3200\\ -2150\\ 450\end{bmatrix}$$

The augmented matrix is

$$\begin{bmatrix} 1875 & -1250 & 0 & 0 & 0 & 0 & 500\\ -1250 & 6250 & -5000 & 0 & 0 & 0 & -875\\ 0 & -5000 & 10000 & -5000 & 0 & 0 & -1000\\ 0 & 0 & -5000 & 8400 & -3400 & 0 & 3200\\ 0 & 0 & 0 & -3400 & 4400 & -1000 & -2150\\ 0 & 0 & 0 & 0 & -1000 & 1000 & 450 \end{bmatrix}$$

Gauss-Elimination follows

$$\begin{bmatrix} 1875 & -1250 & 0 & 0 & 0 & 0 & 500\\ 0 & 5416.67 & -5000 & 0 & 0 & 0 & -541.667\\ 0 & -5000 & 10000 & -5000 & 0 & 0 & -1000\\ 0 & 0 & -5000 & 8400 & -3400 & 0 & 3200\\ 0 & 0 & 0 & -3400 & 4400 & -1000 & -2150\\ 0 & 0 & 0 & 0 & -1000 & 1000 & 450 \end{bmatrix}\begin{array}{l} \\ \leftarrow r_2 + \frac{1250}{1875}r_1\\ \\ \\ \\ \\ \end{array}$$

Applying sequentially the following row operations eliminates all below diagonal elements: $r_3 \leftarrow r_3 + (5000/5416.67)r_2$, $r_4 \leftarrow r_4 + (5000/5384.62)r_3$, $r_5 \leftarrow r_5 + (3400/3757.14)r_4$ and $r_6 \leftarrow r_6 + (3400/3757.14)r_5$.

The resulting upper triangular system is solved with back-substitution as follows:

$$244.253x_6 = 61.0632 \quad \Rightarrow \quad x_6 = 0.25\,\mathrm{m}$$

and

$$\begin{aligned}
1323.19x_5 - 1000x_6 = -514.639 \quad &\Rightarrow \quad x_5 = -0.2\,\mathrm{m} \\
3757.14x_4 - 3400x_5 = 1807.14 \quad &\Rightarrow \quad x_4 = 0.3\,\mathrm{m} \\
5384.62x_3 - 5000x_4 = -1500 \quad &\Rightarrow \quad x_3 = 0\,\mathrm{m} \\
5416.67x_2 - 5000x_3 = -541.667 \quad &\Rightarrow \quad x_2 = -0.1\,\mathrm{m} \\
1875\,x_1 - 1250x_2 = 500 \quad &\Rightarrow \quad x_1 = 0.2\,\mathrm{m}
\end{aligned}$$

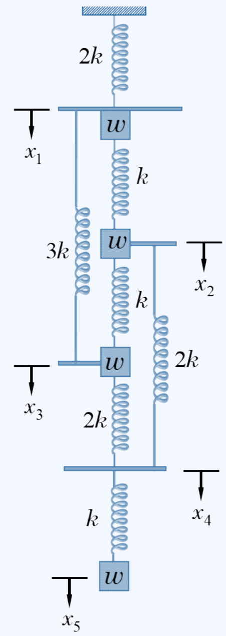
Discussion: Thomas's algorithm, also known as the *Tridiagonal Matrix Algorithm* (TDMA), is commonly used for solving ordinary differential equations, especially in applications involving heat conduction, fluid dynamics, and other engineering fields where tridiagonal systems frequently arise.

Thomas's algorithm is very efficient, with a time complexity of $\mathcal{O}(n)$, as it only requires a single forward and backward pass through the data. This is a significant improvement over general methods like Gaussian elimination, which have a time complexity of $\mathcal{O}(n^3)$.

EXAMPLE 2.10: Solving Banded System of Linear Equations

Linear springs are connected to masses of equal weights, along with rigid massless plates, as illustrated in the figure. To find the deflection of the linear system, assume the plates move vertically up and down with no rotation. The equilibrium equations can then be expressed as

$$
\begin{aligned}
&w + 2k(0-x_1) + k(x_2-x_1) + 3k(x_3-x_1) = 0\\
&w + \ \ k(x_1-x_2) + k(x_3-x_2) + 2k(x_4-x_2) = 0\\
&w + \ 3k(x_1-x_3) + k(x_2-x_3) + 2k(x_4-x_3) = 0\\
&0 \ + \ \ 2k(x_3-x_4) + 2k(x_2-x_4) + k(x_5-x_4) = 0\\
&w + k(x_4-x_5) = 0
\end{aligned}
$$



where x and k denote displacement and spring constant, respectively. (a) Express the equilibrium equations in **Ax=b** form; (b) use the *Gauss-Elimination* method as well as the data given to find the displacements. Given: $k/w = 15$.

SOLUTION:

The linear system can be rearranged into matrix form as follows:

$$
\left\{
\begin{aligned}
w &= 6k\,x_1 - k\,x_2 - 3k\,x_3\\
w &= -k\,x_1 + 4k\,x_2 - k\,x_3 - 2k\,x_4\\
w &= -3k\,x_1 - k\,x_2 + 6k\,x_3 + 2k\,x_4\\
0 &= -2k\,x_2 - 2k\,x_3 + 5k\,x_4 - k\,x_5\\
w &= -k\,x_4 + k\,x_5
\end{aligned}
\right\}
\Rightarrow
\begin{bmatrix}
6 & -1 & -3 & & \\
-1 & 4 & -1 & -2 & \\
-3 & -1 & 6 & 2 & 0\\
 & -2 & -2 & 5 & -1\\
 & & 0 & -1 & 1
\end{bmatrix}
\begin{bmatrix} x_1\\ x_2\\ x_3\\ x_4\\ x_5\end{bmatrix}
= \frac{w}{k}
\begin{bmatrix} 1\\ 1\\ 1\\ 0\\ 1\end{bmatrix}
$$

The augmented matrix is

$$
\begin{bmatrix}
6 & -1 & -3 & 0 & 0 & \frac{1}{15}\\
-1 & 4 & -1 & -2 & 0 & \frac{1}{15}\\
-3 & -1 & 6 & 2 & 0 & \frac{1}{15}\\
0 & -2 & -2 & 5 & -1 & 0\\
0 & 0 & 0 & -1 & 1 & \frac{1}{15}
\end{bmatrix}
$$

The row reductions are carried out as follows:

$$
\sim
\begin{bmatrix}
6 & -1 & -3 & 0 & 0 & \frac{1}{15}\\
0 & \frac{23}{6} & -\frac{3}{2} & -2 & 0 & \frac{7}{90}\\
0 & -\frac{3}{2} & \frac{9}{2} & 2 & 0 & \frac{1}{10}\\
0 & -2 & -2 & 5 & -1 & 0\\
0 & 0 & 0 & -1 & 1 & \frac{1}{15}
\end{bmatrix}
\begin{matrix} \\ \leftarrow r_2 + \frac{1}{6}r_1\\ \leftarrow r_3 + \frac{1}{2}r_1\\ \\ \\ \end{matrix}
\sim
\begin{bmatrix}
6 & -1 & -3 & 0 & 0 & \frac{1}{15}\\
0 & \frac{23}{6} & -\frac{3}{2} & -2 & 0 & \frac{7}{90}\\
0 & 0 & \frac{90}{23} & \frac{28}{23} & 0 & \frac{3}{23}\\
0 & 0 & -\frac{64}{23} & \frac{91}{23} & -1 & \frac{14}{345}\\
0 & 0 & 0 & -1 & 1 & \frac{1}{15}
\end{bmatrix}
\begin{matrix} \\ \\ \leftarrow r_3 + \frac{9}{23}r_2\\ \leftarrow r_4 + \frac{12}{23}r_2\\ \\ \end{matrix}
$$

$$
\sim
\begin{bmatrix}
6 & -1 & -3 & 0 & 0 & \frac{1}{15}\\
0 & \frac{23}{6} & -\frac{3}{2} & -2 & 0 & \frac{7}{90}\\
0 & 0 & \frac{90}{23} & \frac{28}{23} & 0 & \frac{3}{23}\\
0 & 0 & 0 & \frac{217}{45} & -1 & \frac{2}{15}\\
0 & 0 & 0 & -1 & 1 & \frac{1}{15}
\end{bmatrix}
\begin{matrix} \\ \\ \\ \leftarrow r_4 + \frac{32}{45}r_3\\ \\ \end{matrix}
\sim
\begin{bmatrix}
6 & -1 & -3 & 0 & 0 & \frac{1}{15}\\
0 & \frac{23}{6} & -\frac{3}{2} & -2 & 0 & \frac{7}{90}\\
0 & 0 & \frac{90}{23} & \frac{28}{23} & 0 & \frac{3}{23}\\
0 & 0 & 0 & \frac{217}{45} & -1 & \frac{2}{15}\\
0 & 0 & 0 & 0 & \frac{172}{217} & \frac{307}{3255}
\end{bmatrix}
\begin{matrix} \\ \\ \\ \\ \leftarrow r_5 + \frac{45}{217}r_4\end{matrix}
$$

This leads to an upper-banded triangular system of linear equations, which can also be expressed in equation form as follows:

$$
\begin{aligned}
&2580x_5 = 307,\\
&217x_4 - 45x_5 = 6,\\
&90x_3 + 28x_4 = 3,\\
&345x_2 - 135x_3 - 180x_4 = 7.\\
&90x_1 - 15x_2 - 45x_3 = 1.
\end{aligned}
$$

Now the system of equations is suitable for the back-substitution process. Note that the last equation, which clearly involves only x_5, can be solved for it directly. The rest of the equations are solved from the bottom to the top, that is, in the order in which the equations above are arranged. This process is repeated until x_1 is reached. The solution (converted to cm unit) is found as $x_1 = 2.87\,\text{cm},\ x_2 = 5.43\,\text{cm},\ x_3 = 1.70\,\text{cm},\ x_4 = 5.23\,\text{cm},\ x_5 = 11.9\,\text{cm}$.

Discussion: As we have already learned, the Gauss elimination method gives an exact solution (*within the limits of floating-point precision*) of a linear system in a finite number of steps, making it preferable for applications requiring a precise answer over iterative methods, which may not converge to the exact solution.

In manual calculations, Gaussian elimination in banded systems is applied in the same way as in full matrix structures. However, in computer applications, specialized algorithms are tailored to leverage the structure of the matrix (*see* Section 2.10). In banded systems, it is sufficient to store only the banded structure of the matrix, which can lead to large reductions in memory usage due to storing only the non-zero bands instead of storing the entire matrix. Furthermore, using specialized algorithms for banded systems offers other advantages such as efficiency, stability, and scalability.

When to employ banded solvers?

- When the bandwidths of the coefficient matrix are relatively small compared to the matrix size.
- When dealing with large-scale systems where computational and memory efficiency are critical.
- When the system arises from discretizing PDEs or has structured, repetitive patterns.
- When the numerical stability with minimal fill-in is desired.
- In applications demanding real-time or high-performance computations, such as control systems, fluid dynamics simulations, and signal processing.

EXAMPLE 2.11: Implementation of Cholesky Decomposition

In structural analysis, the *Slope-Deflection method* may be used to solve the moments in various parts of the frame. For $L_1 = L_2 = 0.6$ and $L_3 = 1.5$ m, applying the method to the three-span beam shown in the figure below leads to the following system of linear equations:

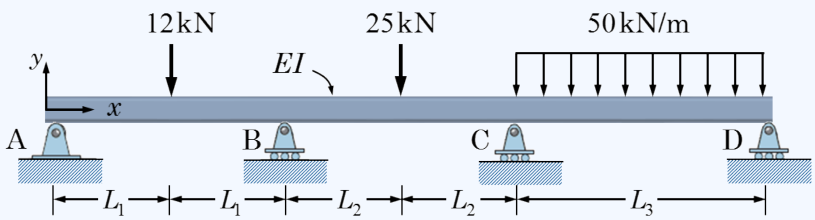
$$2\,\theta_A + \theta_B = -\frac{1.08}{EI}$$

$$\theta_A + 4\,\theta_B + \theta_C = -\frac{1.17}{EI}$$

$$\theta_B + \frac{18}{5}\,\theta_C + \frac{4}{5}\,\theta_D = -\frac{3.375}{EI}$$

$$\frac{4}{5}\,\theta_C + \frac{8}{5}\,\theta_D = \frac{5.625}{EI}$$

where θ_A, θ_B, θ_C, and θ_D are the unknown slopes. Note that the flexural rigidity (EI) changes with the beam material and cross-section. Therefore, to be able to find a more general solution, (a) express the system in matrix form and (b) employ LU-decomposition to solve the given system.



SOLUTION:

(a) The linear, 4×4 symmetric tridiagonal, system can be expressed in matrix form as follows:

$$\begin{bmatrix} 2 & 1 & 0 & 0 \\ 1 & 4 & 1 & 0 \\ 0 & 1 & \frac{18}{5} & \frac{4}{5} \\ 0 & 0 & \frac{4}{5} & \frac{8}{5} \end{bmatrix} \begin{bmatrix} \theta_A \\ \theta_B \\ \theta_C \\ \theta_D \end{bmatrix} = \frac{1}{EI} \begin{bmatrix} -1.080 \\ -1.170 \\ -3.375 \\ 5.625 \end{bmatrix}$$

Using the tridiagonal matrix notations in Section 2.9.2, the matrix for the Cholesky decomposition is setup as

$$d_1 = 2, \;\; d_2 = 4, \;\; d_3 = \frac{18}{5}, \;\; d_4 = \frac{8}{5}, \quad b_2 = b_3 = 1, \;\; b_4 = \frac{4}{5}$$

where the Cholesky algorithm is given as

$$\ell_1 = \sqrt{d_1}, \quad e_k = \frac{b_k}{\ell_{k-1}}, \quad \ell_k = \sqrt{d_k - e_k^2}, \quad \text{for } k = 2, 3, 4$$

Starting with $k = 1$ and employing the algorithm given above, we obtain

$$\ell_1 = \sqrt{d_1} = \sqrt{2}, \quad e_2 = \frac{b_2}{\ell_1} = \frac{1}{\sqrt{2}},$$

$$\ell_2 = \sqrt{d_2 - e_2^2} = \sqrt{4 - \frac{1}{2}} = \sqrt{\frac{7}{2}}, \quad e_3 = \frac{b_3}{\ell_2} = \frac{\sqrt{2}}{\sqrt{7}},$$

$$\ell_3 = \sqrt{d_3 - e_3^2} = \sqrt{\frac{18}{5} - \frac{2}{7}} = \sqrt{\frac{116}{35}}, \quad e_4 = \frac{b_4}{\ell_3} = \frac{4/5}{\sqrt{116/35}} = 2\sqrt{\frac{7}{145}},$$

$$\ell_4 = \sqrt{d_4 - e_4^2} = \sqrt{\frac{8}{5} - \frac{28}{145}} = 2\sqrt{\frac{51}{145}}$$

which gives

$$\mathbf{L} = \begin{bmatrix} \sqrt{2} & 0 & 0 & 0 \\ \frac{1}{\sqrt{2}} & \sqrt{\frac{7}{2}} & 0 & 0 \\ 0 & \sqrt{\frac{2}{7}} & \sqrt{\frac{116}{35}} & 0 \\ 0 & 0 & 2\sqrt{\frac{7}{145}} & 2\sqrt{\frac{51}{145}} \end{bmatrix}$$

Having decomposed coefficient matrix as $\mathbf{LL}^T$, the linear system becomes $\mathbf{LL}^T\boldsymbol{\theta} = \mathbf{b}$. Setting $\mathbf{L}^T\boldsymbol{\theta} = \mathbf{y}$. The system can be written as $\mathbf{Ly} = \mathbf{b}$.

Employing the forward substitution to $\mathbf{Ly} = \mathbf{b}$ yields

$$\begin{bmatrix} \sqrt{2} & 0 & 0 & 0 \\ \frac{1}{\sqrt{2}} & \sqrt{\frac{7}{2}} & 0 & 0 \\ 0 & \sqrt{\frac{2}{7}} & \sqrt{\frac{116}{35}} & 0 \\ 0 & 0 & 2\sqrt{\frac{7}{145}} & 2\sqrt{\frac{51}{145}} \end{bmatrix} \begin{bmatrix} y_A \\ y_B \\ y_C \\ y_D \end{bmatrix} = \frac{1}{EI} \begin{bmatrix} -1.080 \\ -1.170 \\ -3.375 \\ 5.625 \end{bmatrix} \rightarrow \begin{bmatrix} y_A \\ y_B \\ y_C \\ y_D \end{bmatrix} = \frac{1}{EI} \begin{bmatrix} -0.763675 \\ -0.336749 \\ -1.754995 \\ 5.392515 \end{bmatrix}$$

and applying back-substitution to $\mathbf{L}^T\boldsymbol{\theta} = \mathbf{y}$ gives

$$\begin{bmatrix} \sqrt{2} & \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & \sqrt{\frac{7}{2}} & \sqrt{\frac{2}{7}} & 0 \\ 0 & 0 & 2\sqrt{\frac{29}{35}} & 2\sqrt{\frac{7}{145}} \\ 0 & 0 & 0 & 2\sqrt{\frac{51}{145}} \end{bmatrix} \begin{bmatrix} \theta_A \\ \theta_B \\ \theta_C \\ \theta_D \end{bmatrix} = \begin{bmatrix} y_A \\ y_B \\ y_C \\ y_D \end{bmatrix} \rightarrow \begin{bmatrix} \theta_A \\ \theta_B \\ \theta_C \\ \theta_D \end{bmatrix} = \frac{1}{EI} \begin{bmatrix} -0.744485 \\ 0.408971 \\ -2.061397 \\ 4.546323 \end{bmatrix}$$

Discussion: The decomposition methods are favored in systems of linear equations that need to be solved repeatedly (i.e., with multiple right-hand sides). Decomposition methods allow an analyst to decompose the coefficient matrix just once and then reuse **L** and **U** to solve for each new **b** with forward and backward substitution. In this example, the Cholesky method was not required; however, it was applied to this example to illustrate the implementation of the algorithm.

Cholesky decomposition is often used in finite difference and finite element methods to solve systems derived from discretized differential equations. The tridiagonal structure arises frequently in applications like heat conduction, fluid flow, and other simulations. Using Cholesky decomposition in these cases allows for rapid and stable solutions.

The Cholesky decomposition of a tridiagonal symmetric positive definite matrix is also tridiagonal. This reduces storage requirements to just two vectors (the main diagonal and below-diagonals), resulting in $\mathcal{O}(n)$ memory usage. Cholesky decomposition is inherently stable for symmetric positive definite matrices, avoiding the need for pivoting, which simplifies the decomposition. Stability is particularly beneficial when solving linear systems or performing iterative computations where small numerical errors could propagate.