Chapter 25: Solutions of Linear Systems: Existence, Uniqueness, General Form

Introduction

Systems of linear equations arise frequently in engineering, particularly in structural analysis, finite element methods, and geotechnical modeling. Understanding when a system has a solution, whether that solution is unique, and how to express all possible solutions is foundational for computational methods and modeling physical phenomena. This chapter explores the theoretical framework behind the solutions of linear systems, using concepts from matrix algebra and linear transformations.

25.1 System of Linear Equations

A **system of linear equations** is a collection of equations involving the same set of variables. The general form of a system with m equations and n unknowns is:

$$\begin{gathered} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \wedge ¿ b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \wedge ¿ b_2 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \wedge ¿ b_m \\ ¿ \\ ¿ \end{gathered}$$

This can be compactly written in **matrix form** as:

$$Ax = b$$

Where:

- A is an $m \times n$ matrix of coefficients.
- $x \in R^n$ is the vector of unknowns.
- $b \in R^m$ is the right-hand side vector.

25.2 Types of Solutions

A system of linear equations may have:

- **No solution**: The system is inconsistent.
- **Exactly one solution**: The system is consistent and independent.
- **Infinitely many solutions**: The system is consistent but dependent.

The nature of the solution depends on:

- The **rank of the matrix** A,
- The **rank of the augmented matrix** $[A \vee b]$,
- The number of variables n.

25.3 Conditions for Existence of a Solution

A system $Ax=b$ has at least one solution (i.e., it is **consistent**) if and only if:

$$\text{Rank}(A)=\text{Rank}([A \vee b])$$

If this condition is not satisfied, the system has **no solution**.

25.4 Conditions for Uniqueness of Solution

If a solution exists and the rank of A equals the number of unknowns n (i.e., the matrix is of **full column rank**), then the solution is **unique**.

$$\text{If Rank}(A)=n\text{, then the system has a unique solution.}$$

In other words:

- For square systems ($m=n$): If $det(A)\neq 0$, the system has a unique solution.
- For rectangular systems ($m\neq n$): Use rank conditions.

25.5 General Form of Solutions

Homogeneous Systems

If $b=0$, the system is **homogeneous**:

$$Ax=0$$

- The trivial solution $x=0$ always exists.
- If $\text{Rank}(A)<n$, then there are infinitely many **non-trivial solutions**.
- The solution set forms a **vector subspace** of R^n called the **null space** or **kernel** of A.

Non-Homogeneous Systems

For $b\neq 0$, if a particular solution x_p exists, then the **general solution** is given by:

$$x=x_p+x_h$$

Where:

- x_p: A particular solution.
- x_h: General solution of the homogeneous system $Ax=0$.

This means every solution of the non-homogeneous system is a sum of one particular solution and all solutions of the associated homogeneous system.

25.6 Row Reduction and Echelon Forms

To analyze the system, we often apply **Gaussian elimination** or **Gauss–Jordan elimination** to convert the augmented matrix into **row echelon form (REF)** or **reduced row echelon form (RREF)**.

The process helps:

- Determine rank.
- Identify pivot positions.
- Distinguish between leading and free variables.

A general approach:

1. Form the augmented matrix $[A \vee b]$.
2. Perform row operations to get REF or RREF.
3. Interpret the result to determine solution type.

25.7 Geometric Interpretation

- **2 variables**: Each equation represents a line in R^2.
- **3 variables**: Each equation is a plane in R^3.

- Solutions are intersections:
  - o Point → unique solution.
  - o Line/plane → infinite solutions.
  - o No intersection → no solution.

25.8 Rank and Nullity Theorem

For an $m \times n$ matrix A, the **Rank–Nullity Theorem** states:

$$\text{Rank}(A) + \text{Nullity}(A) = n$$

Where:

- Rank ¿ $dim(\text{Image of } A)$
- Nullity ¿ $dim(\text{Null space of } A)$

This provides insight into the dimension of the solution space of $Ax = 0$.

25.9 Application in Civil Engineering

Civil engineers use systems of linear equations to:

- Solve equilibrium equations in structural analysis.
- Model flow networks (e.g., water supply, drainage).
- Perform least squares fitting in surveying and design.
- Implement finite element methods (FEM) for analyzing stresses and strains.

Understanding the solution behavior ensures that models are solvable, stable, and physically realistic.

25.10 Solution Techniques for Linear Systems

While Gaussian elimination is the fundamental technique for solving linear systems, in practical civil engineering computations, several numerical methods are used for larger or sparse systems.

25.10.1 Gaussian Elimination

This method transforms the matrix into an upper triangular form using row operations, and then solves the resulting equations via **back substitution**.

Steps:

1. Forward elimination to convert to upper triangular form.
2. Backward substitution to compute unknowns.

25.10.2 Gauss–Jordan Elimination

This is a modified version of Gaussian elimination that converts the matrix to **reduced row echelon form** (RREF), allowing the solution to be read directly without back-substitution.

25.10.3 Cramer's Rule

For small square systems where $det(A) \neq 0$, Cramer's Rule provides a direct formula:

$$x_i = \frac{det(A_i)}{det(A)} \text{for } i = 1, 2, \ldots, n$$

Where A_i is the matrix formed by replacing the i-th column of A with b.

Limitations: Not suitable for large systems due to computational cost and numerical instability.

25.11 LU Decomposition

LU Decomposition is a matrix factorization technique used to solve linear systems more efficiently, especially when multiple right-hand sides b are involved.

$$A = LU$$

Where:

- L is a lower triangular matrix.
- U is an upper triangular matrix.

To solve $Ax = b$, proceed in two steps:

1. Solve $Ly = b$ (forward substitution).
2. Solve $Ux = y$ (back substitution).

Used in:

- Structural finite element analysis.
- Geotechnical modeling.
- Hydrological simulation.

25.12 Singular and Ill-Conditioned Systems

25.12.1 Singular Matrix

A matrix A is **singular** if $det(A)=0$. In such cases:

- The system may have no solution or infinitely many solutions.
- The matrix is not invertible.

25.12.2 Ill-Conditioned System

An **ill-conditioned** system is highly sensitive to small changes in coefficients or data, leading to large errors in the solution.

This is measured by the **condition number** of matrix A:

$$\kappa(A)=\|A\|\cdot\|A^{-1}\|$$

- $\kappa(A)\gg 1$: Ill-conditioned
- $\kappa(A)\approx 1$: Well-conditioned

Such systems are common in:

- Statically indeterminate structures with close-to-parallel constraints.
- Long-span bridge modeling.

Remedies:

- Use pivoting in Gaussian elimination.
- Improve numerical precision.
- Reformulate the model if necessary.

25.13 Role of Inverse Matrices in Solving Systems

If A is a square and invertible matrix, the system can be solved as:

$$x=A^{-1}b$$

However:

- Computing the inverse is expensive for large systems.
- Not numerically stable; better alternatives include LU decomposition or iterative methods.

Used for:

- Symbolic solution in design models.
- Sensitivity analysis in stress computations.

25.14 Iterative Methods for Large Systems

For very large systems (e.g., in finite element grids), **direct methods** like Gaussian elimination become computationally intensive. Iterative methods are used instead:

25.14.1 Jacobi Method

An iterative algorithm where each variable is solved using the values from the previous iteration:

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

Converges slowly, and only if the matrix is **diagonally dominant**.

25.14.2 Gauss–Seidel Method

Improves over Jacobi by using updated values immediately in the iteration:

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

25.14.3 Successive Over-Relaxation (SOR)

A modification of Gauss–Seidel introducing a relaxation parameter ω:

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

Used extensively in:

- Structural mesh solvers.
- Hydraulic modeling with boundary constraints.

25.15 Rank Deficiency and Least Squares Approximation

25.15.1 Overdetermined Systems

When $m>n$, the system may be inconsistent. In such cases, we seek a solution that **minimizes the error** in a **least squares sense**:

$$\min_x \| A x - b \|_2^2$$

The solution is given by:

$$A^T A x = A^T b$$

Used in:

- Civil surveying.
- Curve fitting in construction data.
- Sensor network calibration.

25.15.2 Pseudo-Inverse (Moore-Penrose)

If A is not square or not invertible:

$$x = A^{+¿b¿}$$

Where $A^{+¿¿}$ is the pseudo-inverse of A.

25.16 Block Matrix Methods

In civil engineering structures (like trusses or large frames), the stiffness matrix is often sparse and **block-structured**. To reduce computational complexity:

- Use **block Gaussian elimination**.
- Exploit **sparsity** with sparse solvers.
- Partition systems using **domain decomposition**.

Used in:

- Finite element software.
- High-rise structural simulations.
- Soil-structure interaction models.
