Lecture 4: **Solutions of Linear Algebraic Equations**

A system of *n* linear algebraic equations in *n* unknowns ($x_1, x_2, \ldots, x_n$) is represented by:

$$
\begin{aligned}
a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n &= b_1 \\
a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n &= b_2 \\
&\vdots \\
a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n &= b_n
\end{aligned}
$$

where *a*'s and *b*'s are constants. In matrix notation [A] [X] = [B], where [A] is the matrix of coefficients (a_{ij}'s), and [X] and [B] are column vectors x_i and b_i respectively.

Graphical method or Cramer's rule can be used to solve equations of up to 3 unknowns. Beyond n=3, the utility of both these methods is limited due to complexity of graphs and increase in computational load. Other techniques are more suitable for solving those equations.

Gauss Elimination Methods

Naï ve Gauss Elimination: In naïve Gauss elimination methods, an expression for x_1 is obtained from the first equation in terms of constants a_{1j}'s, x_j's (j is not equal to i) and b_1. This is substituted in the rest of the equations to eliminate x_1 and obtain n-1 equations in n-1 unknowns ($x_2, \ldots, x_n$). This process is repeated and unknowns x_2, x_3, … eliminated successively to obtain 1 equation in x_n, yielding the value of root for x_n. This is back-substituted in previous equation to obtain the value of x_{n-1}, and so on. The method is called naïve elimination, as it is possible to have a division by zero in the elimination or back-substitution steps. The computational load increases significantly as the number of equations increases.

Gauss-Jordan Elimination: Gauss-Jordan elimination involves elimination of unknowns from all but 1 equation. The variable x_1 is eliminated from all the equations except for the first, x_2 from all except the second, etc., such that ultimately n distinct equations, each in 1 unknown, are obtained. Further, the coefficient of the unknown is scaled to 1 during the elimination steps, so that the modified vector column [B] contains the solution to the problem. Gauss-Jordan elimination requires 50% more computations than the naïve Gauss elimination.

Gauss-Seidel Iterative Method

The system of n equations is rearranged in the form:

$$
\begin{aligned}
x_1 &= [b_1-(a_{12}x_2+\ldots+a_{1n}x_n)]/a_{11} \\
x_2 &= [b_2-(a_{21}x_1+a_{23}x_3+\ldots+a_{2n}x_n)]/a_{22} \\
&\vdots \\
x_n &= [b_n-(a_{n1}x_1+\ldots+a_{nn-1}x_{n-1})]/a_{nn}
\end{aligned}
$$

The iterative procedure starts with the initial estimates of x_i, which are then used in above equations to obtain the modified estimates until a solution is reached. In Gauss-Seidel iteration, the modified values the variables are used immediately in subsequent equations. In Jacobi variation of the iterative methods, the modified values are replaced as a set.

Solution using Software Packages

Matrix operation functions in Excel can be used to solve a system of linear equations. Mathematically, the problem is represented as:

$[A]\,[X] = [B]$

If $[A]^{-1}$ is the inverse of the matrix [A], then

$[A]^{-1}\,[A]\,[X] = [A]^{-1}\,[B]$

Since $[A]^{-1}[A] = [I]$, identity matrix,

$[X] = [A]^{-1}[B]$

So the solution consists of two steps: 1) finding the inverse using matrix inverse function, and 2) matrix multiplication of the inverse and [B] to obtain the solution. Other programs (MATLAB, Mathematica, Polymath, etc.) also have built-in functions and subroutines that use similar technique to solve these problems.

Setting up the appropriate equations (deriving the mathematical model of the system) is the most critical step in obtaining solution to the problem.