Linear Systems

An important use of matrices is in the solution of **systems of linear equations**, or **linear systems**. Linear systems occur in numerous areas of engineering and mathematics, including electric circuits and electric systems.

A linear system might be described by the following equations:

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1$$
$$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = b_2$$
$$a_{31}x_1 + a_{32}x_2 + a_{33}x_3 = b_3$$

These equations could be written in matrix form as:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

The matrix equation could be written as: $\mathbf{Ax} = \mathbf{b}$

Several methods can be used to solve linear systems in the form **Ax = b**, including:

1) Using the inverse matrix, A^{-1}
2) Gaussian elimination
3) Gauss-Jordan reduction

Solving Linear Systems using A^{-1}:

Recall that when multiplying matrices that $AB \neq BA$.

As a result, if both sides of a matrix equation are multiplied by another matrix, there is a difference between ***pre-multiplying*** and ***post-multiplying***.

Example:

$\mathbf{B} = \mathbf{C}$ (original equation)

$\mathbf{AB} = \mathbf{AC}$ (***pre-multiplying*** the original equation by matrix A)

$\mathbf{BA} = \mathbf{CA}$ (***post-multiplying*** the original equation by matrix A)

But since $AB \neq BA$, the result of pre-multiplying and post-multiplying is clearly different. This needs to be kept in mind when solving linear equations in the form $Ax = b$.

$\mathbf{Ax} = \mathbf{b}$ (system of linear equations)

$\mathbf{A^{-1}Ax} = \mathbf{A^{-1}b}$ (pre-multiply by A^{-1})

$\mathbf{Ix} = \mathbf{A^{-1}b}$ (since $A^{-1}A = I$)

$\mathbf{x} = \mathbf{A^{-1}b}$ (since $Ix = x$)

So linear systems can be solved using

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

x = inv(A)*b in MATLAB

Example: Solve the system of equations below using $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$ with MATLAB.

$$x_1 + 2x_2 + 3x_3 = 9$$
$$x_1 + 3x_2 + 4x_3 = 11$$
$$x_1 + 4x_2 + 3x_3 = 7$$

Solution:

```
% Filename:  Linear_Inv.m
% Solve a system of equations using x = inv(A)*b
A = [1 2 3; 1 3 4; 1 4 3]
b = [9;11;7]
x = inv(A)*b
```

```
EDU>> Linear_Inv.m
A =
     1     2     3
     1     3     4
     1     4     3
b =
     9
    11
     7
x =
     2
    -1
     3
```

Note: MATLAB may give warnings about using this method to solve equations and may recommend a different method. We will discuss this later.

Gaussian Elimination

Which system of equations below is easier to solve?

$$\begin{aligned} x+y+z&=7 \\ 3x+2y+z&=11 \\ 4x-2y+2z&=8 \end{aligned} \qquad \begin{aligned} x+y+z&=7 \\ -y-2z&=-10 \\ 10z&=40 \end{aligned}$$

The system on the right is easier because we can easily:

- Solve for z in the 3rd equation
- Substitute the value of z into the 2nd equation and solve for y
- Substitute the values of y and z into the 1st equation and solve for x

The right set of equations is easier to solve for because it is in **row-echelon form** where we can easily use **back substitution**.

It may be hard to recognize, but the two systems of equations are equivalent!

Gaussian Elimination

$$\begin{aligned} x + y + z &= 7 \\ 3x + 2y + z &= 11 \\ 4x - 2y + 2z &= 8 \end{aligned}$$

Original equations

$$\begin{aligned} x + y + z &= 7 \\ - y - 2z &= -10 \\ 10z &= 40 \end{aligned}$$

Equations in **row-echelon** form

$$\begin{bmatrix} 1 & 1 & 1 & 7 \\ 3 & 2 & 1 & 11 \\ 4 & -2 & 2 & 8 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 1 & 1 & 7 \\ 0 & -1 & -2 & -10 \\ 0 & 0 & 10 & 40 \end{bmatrix}$$

Original equations in **augmented matrix** form

Augmented matrix manipulated into **row-echelon** form

The matrix was manipulated using **elementary row operations**.

The process is called **Gaussian elimination**.

Manipulating augmented matrices is similar to how we manipulate equations.

Example:

$$2x+3y+4z=5 \xrightarrow[\text{of an equation by 2}]{} 4x+6y+8z=10$$

Multiply both sides of an equation by 2

$$\begin{bmatrix} 2 & 3 & 4 & 5 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{bmatrix} \rightarrow 2R_1 \rightarrow \begin{bmatrix} 4 & 6 & 8 & 10 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{bmatrix}$$

This was a type of ***elementary row operation***

Example:

$$\left.\begin{aligned} x+3y-2z&=4 \\ -2x+5y+8z&=1 \end{aligned}\right\} \longrightarrow 0x+11y+4z=9$$

Add 2 times Eq 1 to Eq 2 to form a new equation

$$\begin{bmatrix} 1 & 3 & -2 & 4 \\ -2 & 5 & 8 & 1 \\ a_{31} & a_{32} & a_{33} & b_3 \end{bmatrix} \rightarrow R_2+(2)R_1 \rightarrow \begin{bmatrix} 1 & 3 & -2 & 4 \\ 0 & 11 & 4 & 9 \\ a_{31} & a_{32} & a_{33} & b_3 \end{bmatrix}$$

This was another type of ***elementary row operation.*** Note that Eq 2 was ***replaced*** by the new equation.

Elementary Row Operations

There are three types of elementary row operations:

1) Multiply a row by a non-zero constant

Notation: $\rightarrow \mathbf{3R_1} \rightarrow$ (multiply row 1 by 3)

2) Interchange two rows

Notation: $\rightarrow \mathbf{R_{2,3}} \rightarrow$ (interchange row 2 and row 3)

3) Add a multiple of one row to another row

Notation: $\rightarrow \mathbf{R_2 + (3)R_1} \rightarrow$ (add 3 times row 1 to row 2)

Note: ***Elementary column operations*** cannot be used to solve systems of equations, but they could perhaps be used in other applications not covered in this course.

Example 1: Solve the following three equations using ***Gaussian elimination***.

$$x + y + z = 7$$

$$3x + 2y + z = 11$$

$$4x - 2y + 2z = 8$$

Pivot element

Pivot column

Solution: Form the ***augmented matrix***:

$$\begin{bmatrix} \boxed{1} & 1 & 1 & 7 \\ 3 & 2 & 1 & 11 \\ 4 & -2 & 2 & 8 \end{bmatrix} \begin{matrix} \\ \to R_2 + (-3)R_1 \to \\ \to R_3 + (-4)R_1 \to \end{matrix}$$

Perform ***elementary row operations*** using column 1 as the ***pivot column***:

$$\begin{bmatrix} 1 & 1 & 1 & 7 \\ 0 & \boxed{-1} & -2 & -10 \\ 0 & -6 & -2 & -20 \end{bmatrix} \begin{matrix} \\ \\ \to R_3 + (-6)R_2 \to \end{matrix}$$

Perform ***elementary row operations*** using column 2 as the ***pivot column***:

$$\begin{bmatrix} 1 & 1 & 1 & 7 \\ 0 & -1 & -2 & -10 \\ 0 & 0 & 10 & 40 \end{bmatrix}$$

The matrix is now in ***row-echelon*** form

Back substitute to solve for x, y, and z.

$\underline{\text{Row 3}}: \quad 10z = 40, \qquad \text{so } z = 4$

$\underline{\text{Row 2}}: \quad -y - 2(4) = -10, \quad \text{so } y = 2$

$\underline{\text{Row 1}}: \quad x + 2 + 4 = 7, \qquad \text{so } x = 1$

Note: Although it is not required, it is common to adjust each column so that the ***leading coefficient is 1***.

For the last example, we could continue as follows:

$$\begin{bmatrix} 1 & 1 & 1 & 7 \\ 0 & -1 & -2 & -10 \\ 0 & 0 & 10 & 40 \end{bmatrix} \begin{matrix} \\ \rightarrow (-1)R_2 \rightarrow \\ \rightarrow (0.1)R_3 \rightarrow \end{matrix} \begin{bmatrix} 1 & 1 & 1 & 7 \\ 0 & 1 & 2 & 10 \\ 0 & 0 & 1 & 4 \end{bmatrix}$$

Now the back substitution is even easier:

$\underline{\text{Row }3}: \quad z = 4$

$\underline{\text{Row }2}: \quad y + 2(4) = 10, \quad \text{so } y = 2$

$\underline{\text{Row }1}: \quad x + 2 + 4 = 7, \qquad \text{so } x = 1$

Checking results: It is a good idea to check your results by substituting the answers back into the original equations. Try this for the problem above:

$$x + y + z = 7$$

$$3x + 2y + z = 11$$

$$4x - 2y + 2z = 8$$

Rearranging rows: When performing Gaussian reduction, ***the pivot element must be non-zero***. If there is a zero in the pivot element position, it is useful to rearrange the rows (one of the three elementary row operations).

Example: Solve the following system of equations.

$$y + 3z = 4$$
$$-x + 2y = 3$$
$$2x - 3y + 4z = 1$$

$$\begin{bmatrix} 0 & 1 & 3 & 4 \\ -1 & 2 & 0 & 3 \\ 2 & -3 & 4 & 1 \end{bmatrix} \to R_{1,2} \to \begin{bmatrix} -1 & 2 & 0 & 3 \\ 0 & 1 & 3 & 4 \\ 2 & -3 & 4 & 1 \end{bmatrix} \to (-1)R_1 \to \begin{bmatrix} 1 & -2 & 0 & -3 \\ 0 & 1 & 3 & 4 \\ 2 & -3 & 4 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & -2 & 0 & -3 \\ 0 & 1 & 3 & 4 \\ 2 & -3 & 4 & 1 \end{bmatrix} \to R_3 + (-2)R_1 \to \begin{bmatrix} 1 & -2 & 0 & -3 \\ 0 & 1 & 3 & 4 \\ 0 & 1 & 4 & 7 \end{bmatrix} \to R_3 + (-1)R_2 \to \begin{bmatrix} 1 & -2 & 0 & -3 \\ 0 & 1 & 3 & 4 \\ 0 & 0 & 1 & 3 \end{bmatrix}$$

Row 3: $z = 3$

Row 2: $y + 3(3) = 4, \quad \text{so } y = -5$

Row 1: $x - 2(-5) = -3, \quad \text{so } x = -13$

(Sub into original equations to check)

<u>Gauss-Jordan Elimination</u>: In using ***Gauss-Jordan elimination*** (or ***Gauss-Jordan reduction***), we continue where Gaussian elimination left off and use additional elementary row operations until the augmented matrix is in reduced row-echelon form. This will eliminate the need for back substitution.

$$\begin{aligned} x - 2y + 3z &= 9 \\ -x + 3y \quad\;\; &= -4 \\ 2x - 5y + 5z &= 17 \end{aligned}$$

$$\begin{bmatrix} 1 & -2 & 3 & 9 \\ -1 & 3 & 0 & -4 \\ 2 & -5 & 5 & 17 \end{bmatrix} \xrightarrow[\text{Gaussian elimination}]{\text{Elementary row operations}} \underbrace{\begin{bmatrix} 1 & -2 & 3 & 9 \\ 0 & 1 & 3 & 5 \\ 0 & 0 & 1 & 2 \end{bmatrix}}_{\text{Row-echelon form}} \xrightarrow[\text{Gauss-Jordan elimination}]{\text{Elementary row operations}} \underbrace{\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 2 \end{bmatrix}}_{\text{Reduced row-echelon form}}$$

Use back substitution to find results

Read results directly: x = 1, y = -1, z = 2

See next slide for step-by-step details

Gauss-Jordan Elimination: Solve the system of equations below using ***Gauss-Jordan reduction*** (same example as on previous slide but detail added).

$$\begin{aligned} x - 2y + 3z &= 9 \\ -x + 3y \quad &= -4 \\ 2x - 5y + 5z &= 17 \end{aligned}$$

Solution:

$$\begin{bmatrix} 1 & -2 & 3 & 9 \\ -1 & 3 & 0 & -4 \\ 2 & -5 & 5 & 17 \end{bmatrix} \begin{matrix} \to R_2 + R_1 \to \\ \to R_3 + (-2)R_1 \to \end{matrix} \begin{bmatrix} 1 & -2 & 3 & 9 \\ 0 & 1 & 3 & 5 \\ 0 & -1 & -1 & -1 \end{bmatrix} \to R_3 + R_2 \to \begin{bmatrix} 1 & -2 & 3 & 9 \\ 0 & 1 & 3 & 5 \\ 0 & 0 & 2 & 4 \end{bmatrix} \to (1/2)R_3 \to \overbrace{\begin{bmatrix} 1 & -2 & 3 & 9 \\ 0 & 1 & 3 & 5 \\ 0 & 0 & 1 & 2 \end{bmatrix}}^{\text{Row-echelon form}}$$

$$\begin{bmatrix} 1 & -2 & 3 & 9 \\ 0 & 1 & 3 & 5 \\ 0 & 0 & 1 & 2 \end{bmatrix} \to R_1 + (2)R_2 \to \begin{bmatrix} 1 & 0 & 9 & 19 \\ 0 & 1 & 3 & 5 \\ 0 & 0 & 1 & 2 \end{bmatrix} \begin{matrix} \to R_1 + (-9)R_3 \to \\ \to R_2 + (-3)R_3 \to \end{matrix} \underbrace{\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 2 \end{bmatrix}}_{\text{Reduced row-echelon form}}$$

Pivot element

Pivot column

Results: ***x = 1, y = -1, z = 2***

Example: Solve the system of equations below using ***Gauss-Jordan reduction***.

$$2x + 4y = -2$$
$$x + 2y + 2z = 7$$
$$3x - 3y - z = 11$$

Results: ***x = 3, y = -2, z = 4***

rref( ) - a useful function in MATLAB for reducing an augmented matrix into **r**educed **r**ow **e**chelon **f**orm

Example: Use ***rref( )*** to solve the following systems of equations (both from earlier examples).

System 1:

$$\begin{aligned} x - 2y + 3z &= 9 \\ -x + 3y &= -4 \\ 2x - 5y + 5z &= 17 \end{aligned}$$

System 2:

$$\begin{aligned} x + y + z &= 7 \\ 3x + 2y + z &= 11 \\ 4x - 2y + 2z &= 8 \end{aligned}$$

```
Aug1 =
     1    -2     3     9
    -1     3     0    -4
     2    -5     5    17
Reduced_Aug1 =
     1     0     0     1
     0     1     0    -1
     0     0     1     2
Aug2 =
     1     1     1     7
     3     2     1    11
     4    -2     2     8
Reduced_Aug2 =
     1     0     0     1
     0     1     0     2
     0     0     1     4
```

```
% Filename:  GaussJordan1.m
% Two examples using rref() to solve systems of equations
% rref() reduces an augmented matrix into reduced row echelon form
Aug1 = [1 -2 3 9;-1 3 0 -4;2 -5 5 17]  % Enter augmented matrix
Reduced_Aug1 = rref(Aug1)
A2 = [1 1 1;3 2 1;4 -2 2];     % Enter A and b
b2 = [7;11;8];
Aug2 = [A2,b2]                 % Form augmented matrix from A and b
Reduced_Aug2 = rref(Aug2)
```

Left Division in MATLAB - It is recommended that a system of linear equations in the form

Ax = b

be solved using left division

x = A\b

instead of using

x = inv(A)*b

Advantage of using left division

In general, x = A\b is more stable and faster. Why? Some reasons include:

- inv(A) may not exist
- x = A\b uses Gaussian elimination and:
  - Scales matrix entries to minimize errors
  - Uses faster algorithms for special matrices, such as sparse, symmetrical or banded matrices.

Example: - The following represents an ill-conditioned linear system. The error in the result depends highly on the number of significant digits used unless the equations are scaled.

```
% Filename:  Ill_Conditioned_System.m
% Compare the results using x = inv(A)*b and x = A\b
A = [1.00000000000001, 4.0000;2.0000, 8.0000]  % Almost singular
b = [9.0000;18.0000]
x1 = inv(A)*b
x2 = A\b
```

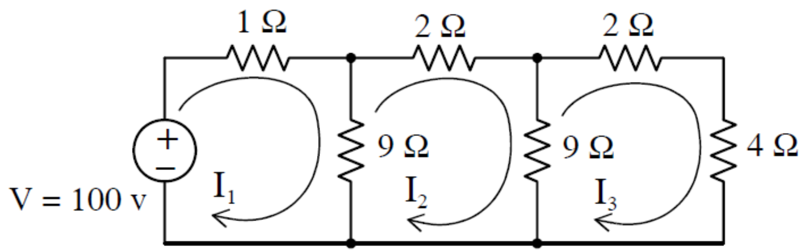
```
EDU>> Ill_Conditioned_System
A =
    1.0000    4.0000
    2.0000    8.0000
b =
     9
    18
Warning: Matrix is close to singular or badly scaled.
         Results may be inaccurate. RCOND = 7.401487e-017.
> In Ill_Conditioned_System at 6
x1 =
         0
    2.2500
x2 =
   -0.6000
    2.4000
```

- Note the warning associated with using x1 = inv(A)*b.
- Note that the results are different.

Example: A) Solve the following equations using ***Gauss-Jordan reduction***

B) Solve the equations in MATLAB using three methods:

- x = inv(A)*b
- rref( )
- x = A\b



KVL, meshes 1-3 yields:

$$10I_1 - 9I_2 = 100$$
$$-9I_1 + 20I_2 - 9I_3 = 0$$
$$-9I_2 + 15I_3 = 0$$

Answers (to check your results): $I_1 = 22.46$ A, $I_2 = 13.85$ A, $I_3 = 8.31$ A