CUFE, M. Sc., 2015-2016

Computers & Numerical Analysis (STR 681)

Lecture 2
Linear Algebraic Equations

Dr. Maha Moddather

Structural Engineering Department

Faculty of Engineering – Cairo University

mahamoddather@eng.cu.edu.eg

Spring 2016

Gauss-Jordan

- The Gauss-Jordan method is a variation of Gauss elimination.
- The major difference is that when an unknown is eliminated in the Gauss-Jordan method, it is eliminated from all other equations rather than just the subsequent ones.
- In addition, all rows are normalized by dividing them by their pivot elements. Thus, the elimination step results in an identity matrix rather than a triangular matrix .
- Have the same drawbacks of Gauss Elimination.

Gauss-Jordan

Ex.,

$$3x_1 - 0.1x_2 - 0.2x_3 = 7.85$$
$$0.1x_1 + 7x_2 - 0.3x_3 = -19.3$$
$$0.3x_1 - 0.2x_2 + 10x_3 = 71.4$$

$$\Downarrow$$

$$\left[\begin{array}{ccc|c} 3 & -0.1 & -0.2 & 7.85 \\ 0.1 & 7 & -0.3 & -19.3 \\ 0.3 & -0.2 & 10 & 71.4 \end{array}\right] \quad /3$$

$$\left[\begin{array}{ccc|c} 1 & -0.0333333 & -0.066667 & 2.61667 \\ 0.1 & 7 & -0.3 & -19.3 \\ 0.3 & -0.2 & 10 & 71.4 \end{array}\right] \quad \times 0.1 \quad \times 0.3 \quad -$$

Gauss-Jordan

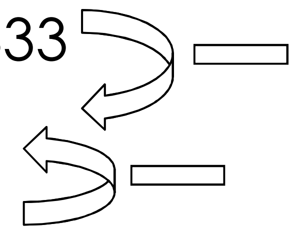
$$\left[\begin{array}{ccc|c} 1 & -0.0333333 & -0.066667 & 2.61667 \\ 0 & 7.00333 & -0.293333 & -19.5617 \\ 0 & -0.190000 & 10.0200 & 70.6150 \end{array}\right]$$

/7.00333

$$\left[\begin{array}{ccc|c} 1 & -0.0333333 & -0.066667 & 2.61667 \\ 0 & 1 & -0.0418848 & -2.79320 \\ 0 & -0.190000 & 10.0200 & 70.6150 \end{array}\right]$$

x -0.0333333

x -0.19



$$\left[\begin{array}{ccc|c} 1 & 0 & -0.0680629 & 2.52356 \\ 0 & 1 & -0.0418848 & -2.79320 \\ 0 & 0 & 10.01200 & 70.0843 \end{array}\right]$$

/10.012

Gauss-Jordan

$$\left[\begin{array}{ccc|c} 1 & 0 & -0.0680629 & 2.52356 \\ 0 & 1 & -0.0418848 & -2.79320 \\ 0 & 0 & 1 & 7.0000 \end{array}\right]$$

x -0.0680629 x -0.0418848

$$\left[\begin{array}{ccc|c} 1 & 0 & 0 & 3.0000 \\ 0 & 1 & 0 & -2.5000 \\ 0 & 0 & 1 & 7.0000 \end{array}\right]$$

LU Decomposition

- The time-consuming elimination step can be formulated so that it involves only operations on the matrix of coefficients, [*A*].
- It is well suited for those situations where many right-hand-side vectors {*B*} must be evaluated for a single value of [*A*].
- One motive for introducing *LU* decomposition is that it provides an efficient means to compute the matrix inverse. The inverse has a number of valuable applications in engineering practice.
- It also provides a means for evaluating system condition.

LU Decomposition

Upper triangular system [U]

$$\begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} = \begin{Bmatrix} d_1 \\ d_2 \\ d_3 \end{Bmatrix}$$

$$\Downarrow$$

$$[A]\{X\} - \{B\} = 0 \equiv [U]\{X\} - \{D\} = 0$$

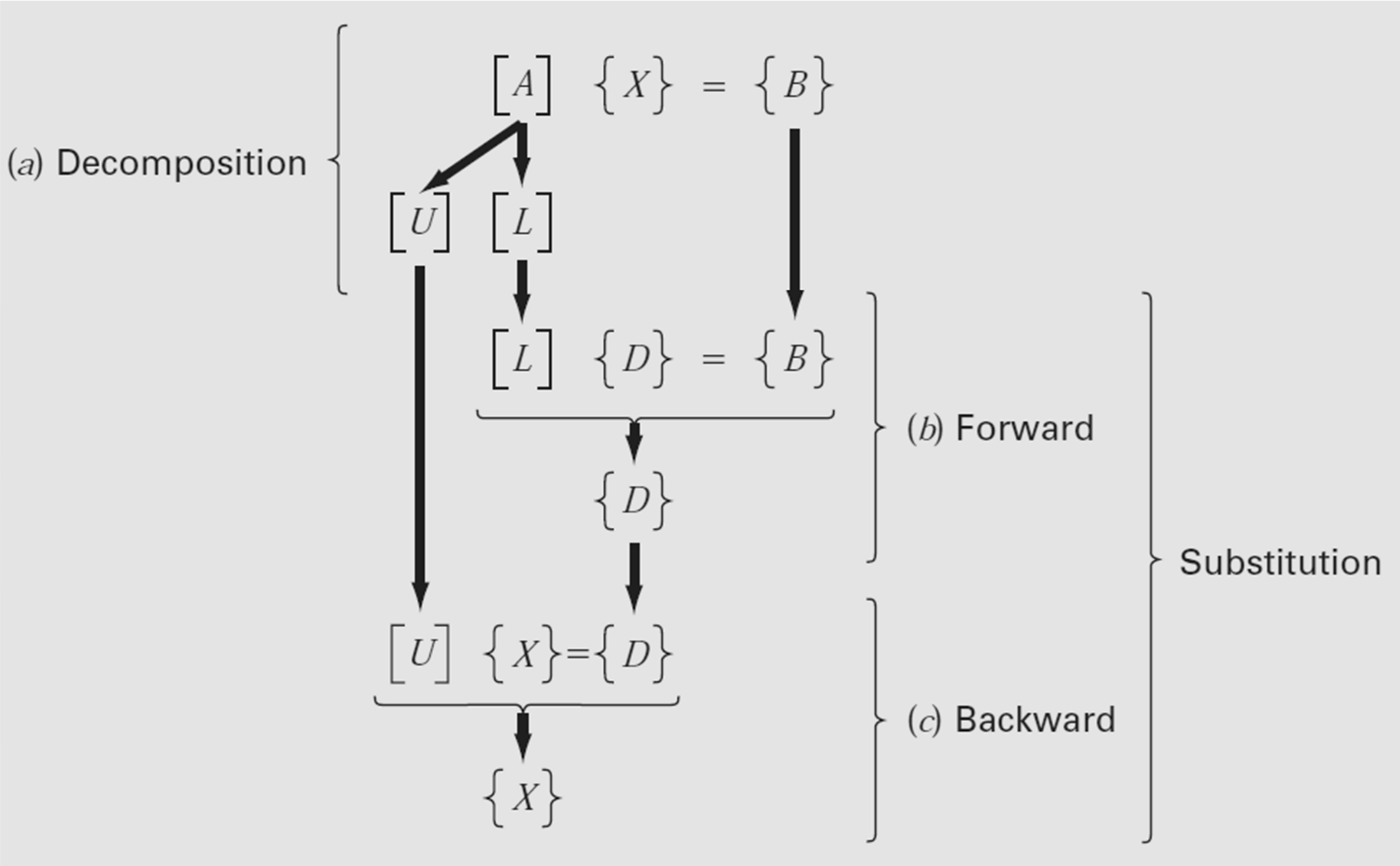
Assume there is a lower triangular system [L]

$$[L] = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix}$$ where $[L][U] = [A]$

$$\Downarrow$$

$$\boxed{[L]\{D\} = \{B\}} \Leftarrow [L]\{[U]\{X\} - \{D\}\} = [A]\{X\} - \{B\}$$

LU Decomposition



LU Decomposition

A two-step strategy for obtaining solutions:

1. **LU decomposition step. [A] is factored or "decomposed" into lower [L] and upper [U] triangular matrices.**
2. **Substitution step. [L] and [U] are used to determine a solution {X} for a right-hand side {B}. This step itself consists of two steps:**
   - ▶ **Generate an intermediate vector {D} by forward substitution.**
   - ▶ **Then, the result can be solved by back substitution for {X}.**

LU Decomposition

$$[A] = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

x (a_{21}/a_{11}) x (a_{31}/a_{11})

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a'_{22} & a'_{23} \\ 0 & a'_{32} & a'_{33} \end{bmatrix}$$

x (a'_{32}/a'_{22})

$$[U] = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a'_{22} & a'_{23} \\ 0 & 0 & a''_{33} \end{bmatrix}$$

$$f_{21} = \frac{a_{21}}{a_{11}}$$

$$f_{32} = \frac{a'_{32}}{a'_{22}}$$

$$f_{31} = \frac{a_{31}}{a_{11}}$$

LU Decomposition

$$[U] = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a'_{22} & a'_{23} \\ 0 & 0 & a''_{33} \end{bmatrix}$$

$$[L] = \begin{bmatrix} 1 & 0 & 0 \\ f_{21} & 1 & 0 \\ f_{31} & f_{32} & 1 \end{bmatrix}$$

$$[L][U] = [A]$$

Can be stores in one matrix

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ f_{21} & a'_{22} & a'_{23} \\ f_{31} & f_{32} & a''_{33} \end{bmatrix}$$

LU Decomposition

$$[L]\{D\} = \{B\}$$

$$d_i = b_i - \sum_{j=1}^{i-1} a_{ij} d_j \qquad \text{for } i = 2, 3, \ldots, n$$

Forward substitution

$$[U]\{X\} = \{D\}$$

$$x_n = d_n / a_{nn}$$

$$x_i = \frac{d_i - \sum_{j=i+1}^{n} a_{ij} x_j}{a_{ii}} \qquad \text{for } i = n-1, n-2, \ldots, 1$$

Backward substitution

LU Decomposition

$$[A] = \begin{bmatrix} 3 & -0.1 & -0.2 \\ 0.1 & 7 & -0.3 \\ 0.3 & -0.2 & 10 \end{bmatrix}$$

$$\Downarrow$$

$$[U] = \begin{bmatrix} 3 & -0.1 & -0.2 \\ 0 & 7.00333 & -0.293333 \\ 0 & 0 & 10.0120 \end{bmatrix}$$

After forward elimination

$$f_{21} = \frac{0.1}{3} = 0.03333333$$

$$f_{31} = \frac{0.3}{3} = 0.1000000$$

$$f_{32} = \frac{-0.19}{7.00333} = -0.0271300$$

LU Decomposition

$$[L] = \begin{bmatrix} 1 & 0 & 0 \\ 0.0333333 & 1 & 0 \\ 0.100000 & -0.0271300 & 1 \end{bmatrix}$$

$$[A] = [L][U] = \begin{bmatrix} 1 & 0 & 0 \\ 0.0333333 & 1 & 0 \\ 0.100000 & -0.0271300 & 1 \end{bmatrix} \begin{bmatrix} 3 & -0.1 & -0.2 \\ 0 & 7.00333 & -0.293333 \\ 0 & 0 & 10.0120 \end{bmatrix}$$

$$= \begin{bmatrix} 3 & -0.1 & -0.2 \\ 0.0999999 & 7 & -0.3 \\ 0.3 & -0.2 & 9.99996 \end{bmatrix}$$

$$[A] = \begin{bmatrix} 3 & -0.1 & -0.2 \\ 0.1 & 7 & -0.3 \\ 0.3 & -0.2 & 10 \end{bmatrix}$$

LU Decomposition

$$\begin{bmatrix} 3 & -0.1 & -0.2 \\ 0.1 & 7 & -0.3 \\ 0.3 & -0.2 & 10 \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} = \begin{Bmatrix} 7.85 \\ -19.3 \\ 71.4 \end{Bmatrix}$$

$$[L] = \begin{bmatrix} 1 & 0 & 0 \\ 0.0333333 & 1 & 0 \\ 0.100000 & -0.0271300 & 1 \end{bmatrix} \qquad \{B\} = \begin{Bmatrix} 7.85 \\ -19.3 \\ 71.4 \end{Bmatrix}$$

$$[L]\{D\} = \{B\} \Rightarrow \begin{bmatrix} 1 & 0 & 0 \\ 0.0333333 & 1 & 0 \\ 0.100000 & -0.0271300 & 1 \end{bmatrix} \begin{Bmatrix} d_1 \\ d_2 \\ d_3 \end{Bmatrix} = \begin{Bmatrix} 7.85 \\ -19.3 \\ 71.4 \end{Bmatrix}$$

get {D} by forward substitution

LU Decomposition

$$d_1 = 7.85$$

$$d_2 = -19.3 - 0.0333333(7.85) = -19.5617$$

$$d_3 = 71.4 - 0.1(7.85) + 0.02713(-19.5617) = 70.0843$$

$$\{D\} = \begin{Bmatrix} 7.85 \\ -19.5617 \\ 70.0843 \end{Bmatrix}$$

⇨ Check slide # 59

$$[U]\{X\} = \{D\} \Rightarrow \begin{bmatrix} 3 & -0.1 & -0.2 \\ 0 & 7.00333 & -0.293333 \\ 0 & 0 & 10.0120 \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} = \begin{Bmatrix} 7.85 \\ -19.5617 \\ 70.0843 \end{Bmatrix}$$

⇩

$$\{X\} = \begin{Bmatrix} 3 \\ -2.5 \\ 7.00003 \end{Bmatrix}$$

Inverse Calculation by LU Decomposition

$$[A] = \begin{bmatrix} 3 & -0.1 & -0.2 \\ 0.1 & 7 & -0.3 \\ 0.3 & -0.2 & 10 \end{bmatrix}$$

$$[U] = \begin{bmatrix} 3 & -0.1 & -0.2 \\ 0 & 7.00333 & -0.293333 \\ 0 & 0 & 10.0120 \end{bmatrix}$$

$$[L] = \begin{bmatrix} 1 & 0 & 0 \\ 0.0333333 & 1 & 0 \\ 0.100000 & -0.0271300 & 1 \end{bmatrix}$$

Inverse Calculation by LU Decomposition

$$[L]\{D\} = \{B\}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0.0333333 & 1 & 0 \\ 0.100000 & -0.0271300 & 1 \end{bmatrix} \begin{Bmatrix} d_1 \\ d_2 \\ d_3 \end{Bmatrix} = \begin{Bmatrix} 1 \\ 0 \\ 0 \end{Bmatrix}$$

get {D} by forward substitution

⇩

$$\{D\}^T = \lfloor 1 \quad -0.03333 \quad -0.1009 \rfloor$$

⇩

$$[U]\ \{X\}=\{D\} \Rightarrow \begin{bmatrix} 3 & -0.1 & -0.2 \\ 0 & 7.00333 & -0.293333 \\ 0 & 0 & 10.0120 \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} = \begin{Bmatrix} 1 \\ -0.03333 \\ -0.1009 \end{Bmatrix}$$

get {X} by back substitution ⇨

$$\{X\}^T = \lfloor 0.33249 \ \ -0.00518 \ \ -0.01008 \rfloor$$

First column in $\{A\}^{-1}$

Inverse Calculation by LU Decomposition

$$[A]^{-1} = \begin{bmatrix} 0.33249 & 0 & 0 \\ -0.00518 & 0 & 0 \\ -0.01008 & 0 & 0 \end{bmatrix}$$

To get the second column, repeat previous steps twice as follows:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0.0333333 & 1 & 0 \\ 0.100000 & -0.0271300 & 1 \end{bmatrix} \begin{Bmatrix} d_1 \\ d_2 \\ d_3 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 1 \\ 0 \end{Bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0.0333333 & 1 & 0 \\ 0.100000 & -0.0271300 & 1 \end{bmatrix} \begin{Bmatrix} d_1 \\ d_2 \\ d_3 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 1 \end{Bmatrix}$$

Inverse Calculation by LU Decomposition

$$[A]^{-1} = \begin{bmatrix} 0.33249 & 0.004944 & 0.006798 \\ -0.00518 & 0.142903 & 0.004183 \\ -0.01008 & 0.00271 & 0.09988 \end{bmatrix}$$

Check $[A][A]^{-1} = [I]$

Importance of Inverse Calculation

Find out if the system is ill-conditioned:

1. **Scale the matrix of coefficients [A] so that the largest element in each row is 1. Invert the scaled matrix and if there are elements of $[A]^{-1}$ that are several orders of magnitude greater than one, it is likely that the system is ill-conditioned.**

$$[A] = \begin{bmatrix} 1 & 1/2 & 1/3 \\ 1/2 & 1/3 & 1/4 \\ 1/3 & 1/4 & 1/5 \end{bmatrix} \Longleftrightarrow [A]^{-1} = \begin{bmatrix} 9 & -18 & 10 \\ -36 & 96 & -60 \\ 30 & -90 & 60 \end{bmatrix}$$

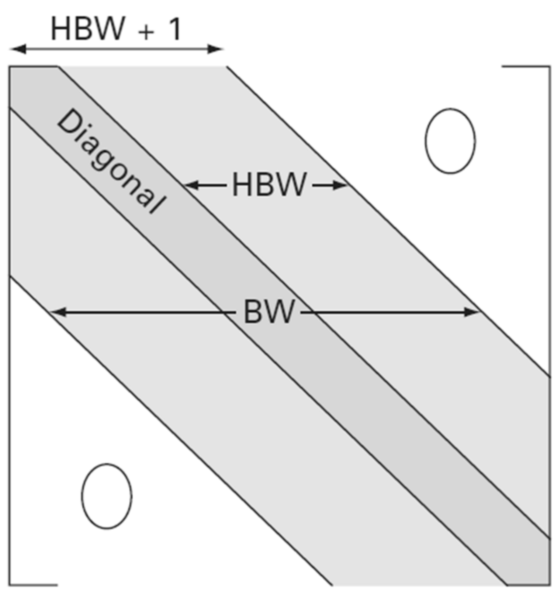
Hilbert Matrix

Importance of Inverse Calculation

Find out if the system is ill-conditioned:

2. Multiply the inverse by the original coefficient matrix and assess whether the result is close to the identity matrix. If not, it indicates ill-conditioning.

Special Matrices: Banded



- **BW: band width**
- **HBW: half band width**
- **BW = 2HBW + 1**
- **Gauss elimination or conventional *LU* decomposition can be employed to solve banded equations, but they are inefficient.**
- **None of the elements outside the band would change from their original values of zero. Thus, unnecessary space and time would be expended on the storage and manipulation of these useless zeros.**
- **An alternative: Thomas Algorithm.**

Special Matrices: Banded

$$\begin{bmatrix} f_1 & g_1 & & & & & \\ e_2 & f_2 & g_2 & & & & \\ & e_3 & f_3 & g_3 & & & \\ & & \cdot & \cdot & \cdot & & \\ & & & \cdot & \cdot & \cdot & \\ & & & & \cdot & \cdot & \cdot \\ & & & & e_{n-1} & f_{n-1} & g_{n-1} \\ & & & & & e_n & f_n \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ \cdot \\ \cdot \\ \cdot \\ x_{n-1} \\ x_n \end{Bmatrix} = \begin{Bmatrix} r_1 \\ r_2 \\ r_3 \\ \cdot \\ \cdot \\ \cdot \\ r_{n-1} \\ r_n \end{Bmatrix}$$

Special Matrices: Banded

$$\begin{bmatrix} 2.04 & -1 & & \\ -1 & 2.04 & -1 & \\ & -1 & 2.04 & -1 \\ & & -1 & 2.04 \end{bmatrix} \begin{Bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{Bmatrix} = \begin{Bmatrix} 40.8 \\ 0.8 \\ 0.8 \\ 200.8 \end{Bmatrix}$$

⇓ Decompose

$$[U] = \begin{bmatrix} 2.04 & -1 & & \\ & 1.550 & -1 & \\ & & 1.395 & -1 \\ & & & 1.323 \end{bmatrix} \quad [L] = \begin{bmatrix} 1 & & & \\ -0.49 & 1 & & \\ & -0.645 & 1 & \\ & & -0.717 & 1 \end{bmatrix}$$

⇓ One matrix

$$\begin{bmatrix} 2.04 & -1 & & \\ -0.49 & 1.550 & -1 & \\ & -0.645 & 1.395 & -1 \\ & & -0.717 & 1.323 \end{bmatrix}$$

[L]

[U]

Special Matrices: Banded

$$e_2 = -1/2.04 = -0.49$$
$$f_2 = 2.04 - (-0.49)(-1) = 1.550$$
$$e_3 = -1/1.550 = -0.645$$
$$f_3 = 2.04 - (-0.645)(-1) = 1.395$$
$$e_4 = -1/1.395 = -0.717$$
$$f_4 = 2.04 - (-0.717)(-1) = 1.323$$

Special Matrices: Banded

$$[L]\{D\} = \{B\}$$

⇓

$$\begin{bmatrix} 1 & & & \\ -0.49 & 1 & & \\ & -0.645 & 1 & \\ & & -0.717 & 1 \end{bmatrix} \{R\} = \begin{Bmatrix} 40.8 \\ 0.8 \\ 0.8 \\ 200.8 \end{Bmatrix}$$

Forward substitution ⇓

$$r_2 = 0.8 - (-0.49)40.8 = 20.8$$

$$r_3 = 0.8 - (-0.645)20.8 = 14.221$$

$$r_4 = 200.8 - (-0.717)14.221 = 210.996$$

⇒ $\{R\} = \begin{Bmatrix} 40.8 \\ 20.8 \\ 14.221 \\ 210.996 \end{Bmatrix}$

Special Matrices: Banded

$$[U]\{X\}=\{D\}$$

⇓

$$\begin{bmatrix} 2.04 & -1 & & \\ & 1.550 & -1 & \\ & & 1.395 & -1 \\ & & & 1.323 \end{bmatrix} \{T\} = \begin{Bmatrix} 40.8 \\ 20.8 \\ 14.221 \\ 210.996 \end{Bmatrix}$$

Back substitution ⇓

$$T_4 = 210.996/1.323 = 159.480$$

$$T_3 = [14.221 - (-1)159.48]/1.395 = 124.538$$

$$T_2 = [20.800 - (-1)124.538]/1.550 = 93.778$$

$$T_1 = [40.800 - (-1)93.778]/2.040 = 65.970$$

Gauss Seidel

- An Iterative or approximate methods provide an alternative to the elimination methods described to this point.

- Those approaches consists of guessing a value and then using a systematic method to obtain a refined estimate of the solution.

- The *Gauss-Seidel method* is the most commonly used iterative method.

Gauss Seidel

(1)

$$x_1 = \frac{b_1 - a_{12}x_2 - a_{13}x_3}{a_{11}}$$

(2)

$$x_2 = \frac{b_2 - a_{21}x_1 - a_{23}x_3}{a_{22}}$$

(3)

$$x_3 = \frac{b_3 - a_{31}x_1 - a_{32}x_2}{a_{33}}$$

1. **Assume all x's are zero.**
2. **Substitute in first equation to get x_1.**
3. **Substitute in the second equation the new value of x_1 and the zero value of x_3 to get x_2.**
4. **Repeat the process to calculate x_3.**
5. **Repeat the steps 1 to 4 until:**

$$|\varepsilon_{a,i}| = \left| \frac{x_i^j - x_i^{j-1}}{x_i^j} \right| 100\% < \varepsilon_s$$

Gauss Seidel

$$\begin{aligned} 3x_1 - 0.1x_2 - 0.2x_3 &= 7.85 \\ 0.1x_1 + 7x_2 - 0.3x_3 &= -19.3 \\ 0.3x_1 - 0.2x_2 + 10x_3 &= 71.4 \end{aligned}$$

$$\Downarrow$$

$$x_1 = \frac{7.85 + 0.1x_2 + 0.2x_3}{3}$$

$$x_2 = \frac{-19.3 - 0.1x_1 + 0.3x_3}{7}$$

$$x_3 = \frac{71.4 - 0.3x_1 + 0.2x_2}{10}$$

True solution

$x_1 = 3$, $x_2 = -2.5$, and $x_3 = 7$

Gauss Seidel

$$x_1 = \frac{7.85 + 0.1x_2 + 0.2x_3}{3}$$

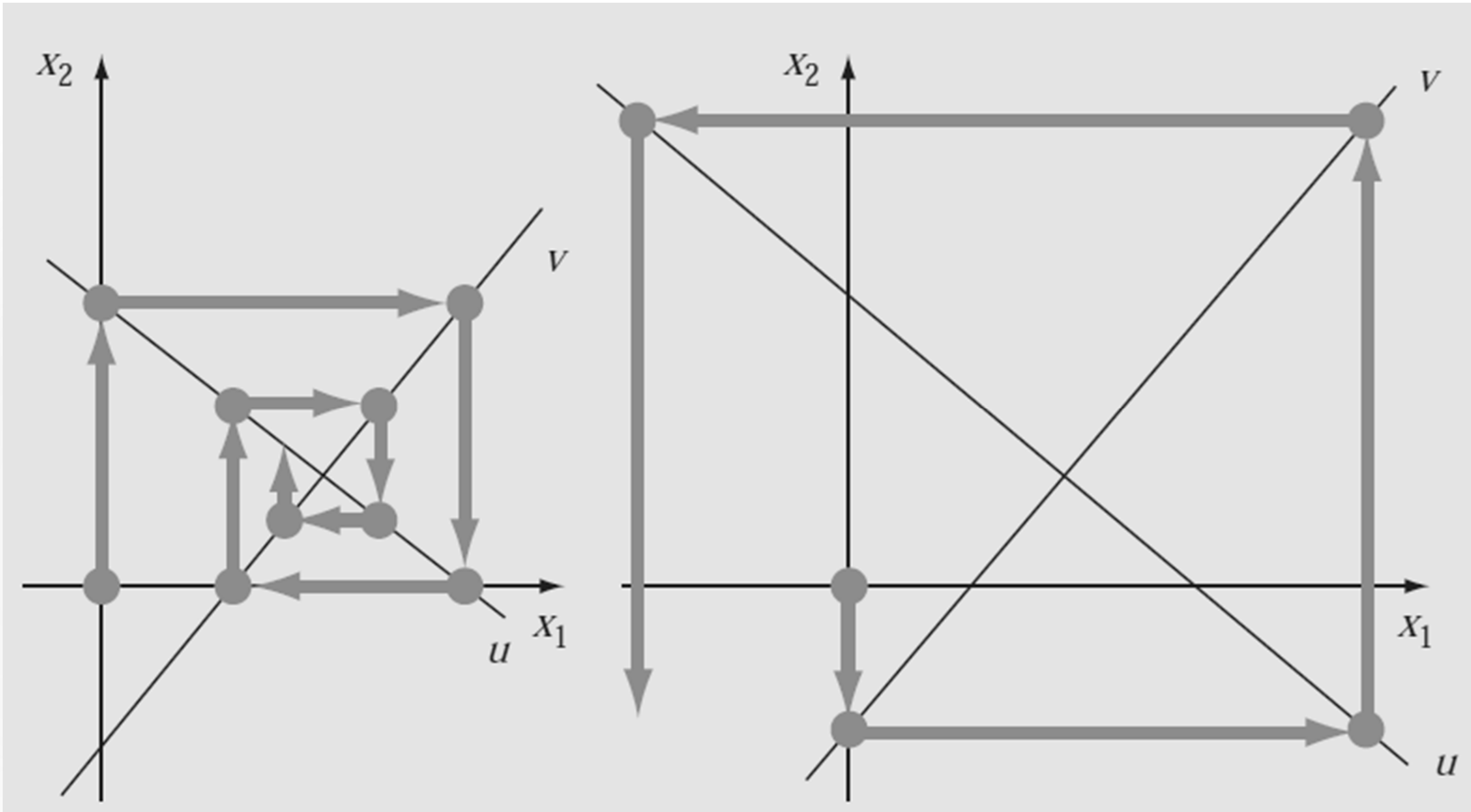
$$x_2 = \frac{-19.3 - 0.1x_1 + 0.3x_3}{7}$$

$$x_3 = \frac{71.4 - 0.3x_1 + 0.2x_2}{10}$$

Trial	x_1	x_2	x_3
1	0	0	0
2	2.616667	-2.794524	7.005610
Error	100%	100%	100%
3	2.990557	-2.499625	7.000291
error	12.5%	11.8%	0.076%

Gauss Seidel Drawbacks

Sometimes, it is nonconvergent.



Convergence　　　　Disconvergence

Gauss Seidel Drawbacks

$$|a_{ii}| > \sum_{\substack{j=1 \\ j \neq i}}^{n} |a_{ij}|$$

- The diagonal coefficient in each of the equations must be larger than the sum of the absolute values of the other coefficients in the equation.
- This criterion is sufficient but not necessary for convergence. That is, although the method may sometimes work if this equation is not met, convergence is guaranteed if the condition is satisfied.
- Systems where this equation holds are called *diagonally dominant*. Fortunately, many engineering problems of practical importance fulfill this requirement.

Gauss Seidel Drawbacks

X1=15.954

X2=8.5

Sometimes, it is nonconvergent.

$$11x_1 + 13x_2 = 286$$
$$11x_1 - 9x_2 = 99$$

$$\Rightarrow [A] = \begin{bmatrix} 11 & 13 \\ 11 & -9 \end{bmatrix}$$

(Nonconvergent)

$|11| < |13|$

$|9| < |11|$

Trial	x_1	x_2
1	0	0
2	26	20.77778
3	1.444442	-9.23457
4	36.91358	34.1166
5	-14.3196	-28.5018
6	59.68395	61.94704
7	-47.2101	-68.7013
8	107.1924	120.013

Gauss Seidel Drawbacks

X1=15.954

X2=8.5

Sometimes, it is convergent, but very slow.

$$11x_1 - 9x_2 = 99$$

$$11x_1 + 13x_2 = 286$$

$$\Rightarrow \quad [A] = \begin{bmatrix} 11 & -9 \\ 11 & 13 \end{bmatrix}$$

Convergent

$|11| > |9|$

$|13| > |11|$

Trial	x_1	x_2
1	0	0
2	9	14.38462
3	20.76923	4.426036
4	12.6213	11.32044
5	18.26218	6.54739
6	14.35696	9.851807
7	17.06057	7.564134
8	15.18884	9.147908

But very slow!

Gauss Seidel Drawbacks

2. **When it converged, it often did so very slowly.**

- ***Relaxation*** **represents a slight modification of the Gauss-Seidel method and is designed to enhance convergence.**
- **After each new value of x is computed, that value is modified by a weighted average of the results of the previous and the present iterations:** $x_i^{\text{new}} = \lambda x_i^{\text{new}} + (1 - \lambda)x_i^{\text{old}}$

where λ is a weighting factor that is assigned a value between 0 and 2.

If $\lambda = 0$, the result is unmodified.

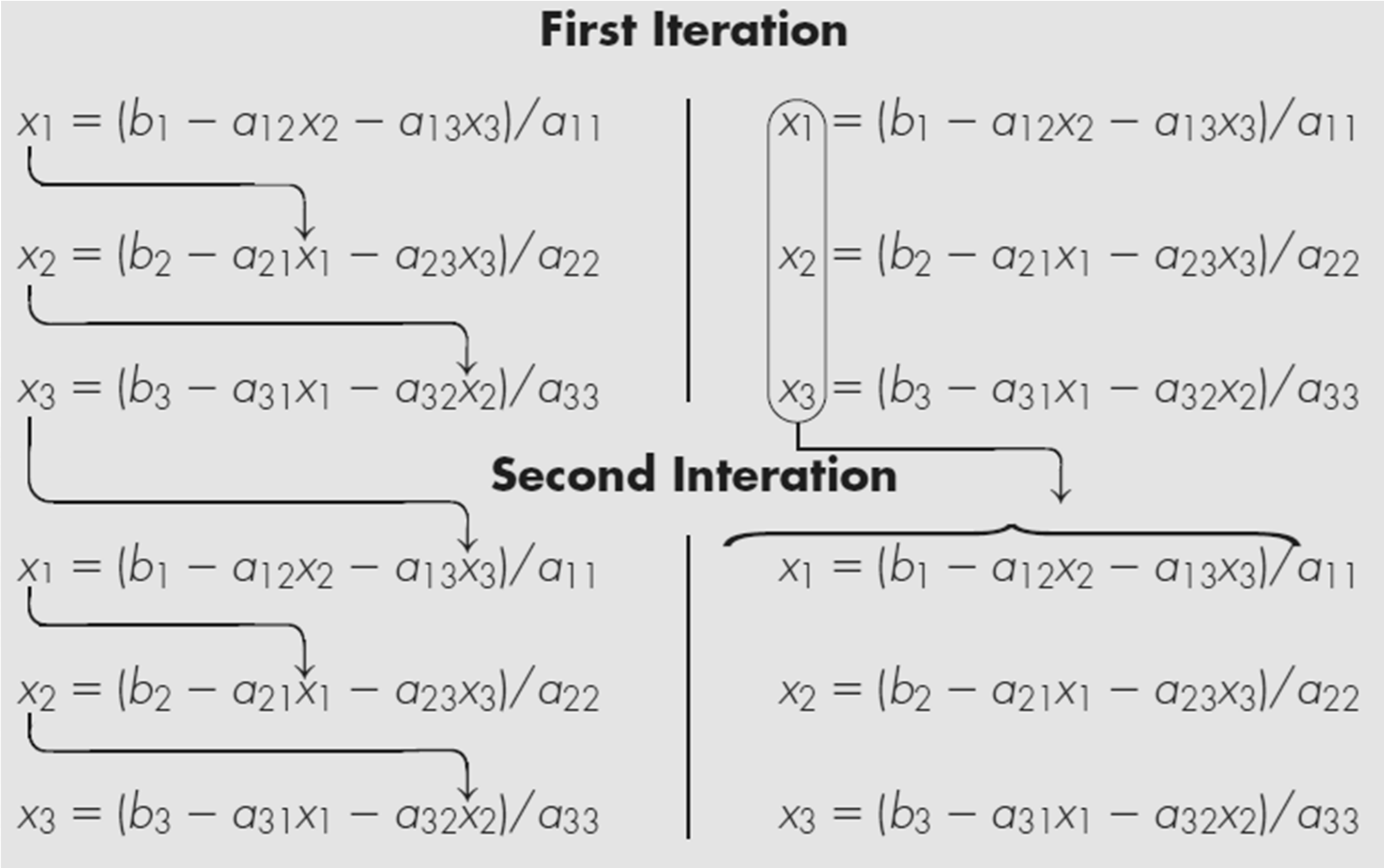
If $\lambda = 0 – 1$, "under relaxation"➔ to make nonconvergent system converge

If $\lambda = 1 – 2$, "successive overrelaxation," (SOR) ➔ to accelerate convergent system

Jacobi Iteration

- As each new x value is computed for the Gauss-Seidel method, it is immediately used in the next equation to determine another x value. Thus, if the solution is converging, the best available estimates will be employed.
- An alternative approach, called *Jacobi iteration,* utilizes a somewhat different tactic. Rather than using the latest available *x*’s, this technique computes a set of new *x*’s on the basis of a set of old *x*’s.
- Thus, as new values are generated, they are not immediately used but rather are retained for the next iteration.

Jacobi Iteration/Gauss Seidel



Gauss Seidel　　　　Jacobi Iteration

Engineering Applications

$$[A]\{X\} = \{B\}$$

$$[\text{Interactions}]\{\text{response}\} = \{\text{stimuli}\}$$

System interaction or coupling

(Geometry, stiffness,…..)

System Response

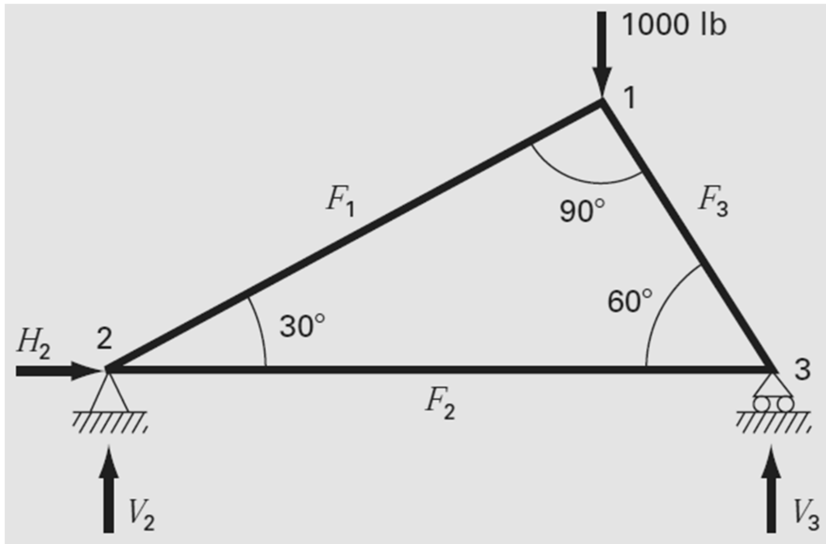
(Reactions or Internal Forces)

External Forces or stimuli that drives the system but independent of it

Case Study

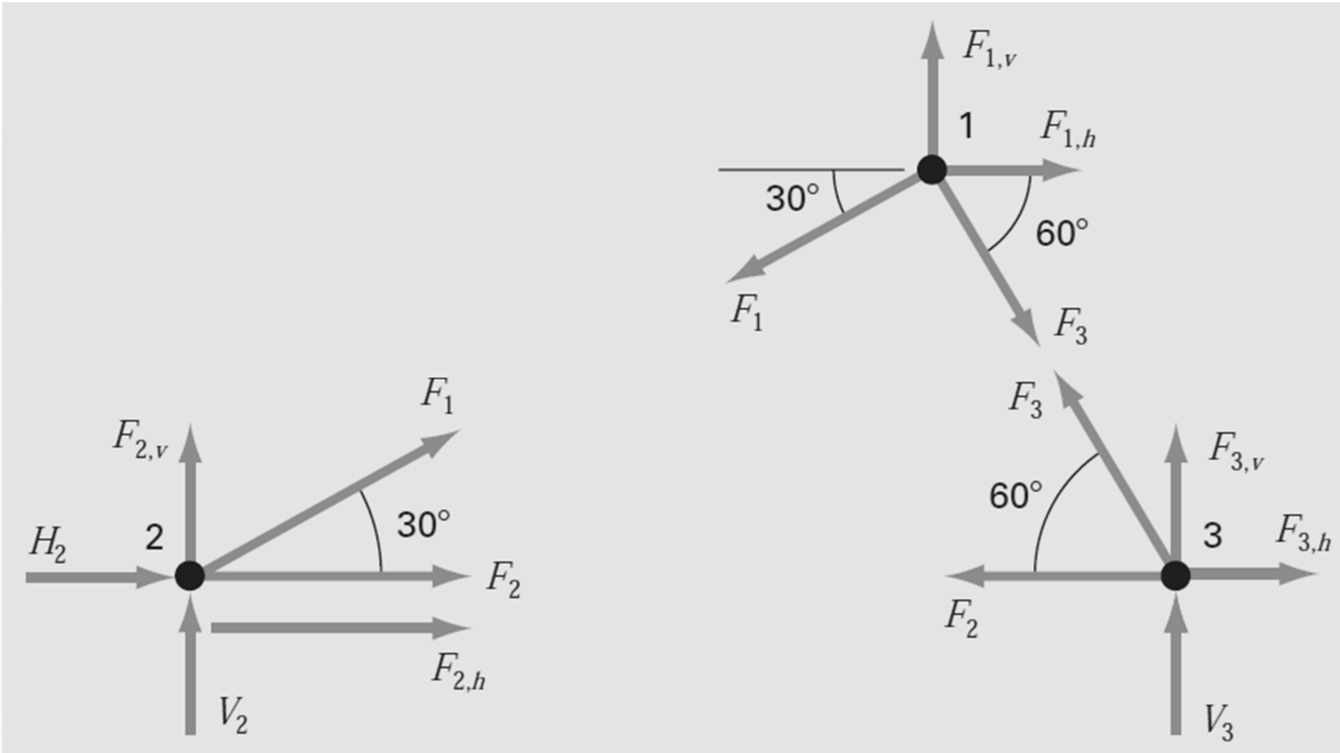
Find the forces and reactions associated with the shown truss.

- **F_1, F_2, F_3 are the tension or compression on the members of the truss.**
- **H_2, V_2, V_3 are forces that characterize how the truss interacts with the supporting surface (reaction).**
- **1000 lb is an external force ($F_{1,v}$)**



Case Study

The sum of forces in both vertical and horizontal directions must be zero at each node.



Free body diagram

Case Study

At node (1):

$$\Sigma F_H = 0 = -F_1 \cos 30^\circ + F_3 \cos 60^\circ + F_{1,h}$$
$$\Sigma F_V = 0 = -F_1 \sin 30^\circ - F_3 \sin 60^\circ + F_{1,v}$$

At node (2):

$$\Sigma F_H = 0 = F_2 + F_1 \cos 30^\circ + F_{2,h} + H_2$$
$$\Sigma F_V = 0 = F_1 \sin 30^\circ + F_{2,v} + V_2$$

At node (3):

$$\Sigma F_H = 0 = -F_2 - F_3 \cos 60^\circ + F_{3,h}$$
$$\Sigma F_V = 0 = F_3 \sin 60^\circ + F_{3,v} + V_3$$

Case Study

$$\begin{bmatrix} 0.866 & 0 & -0.5 & 0 & 0 & 0 \\ 0.5 & 0 & 0.866 & 0 & 0 & 0 \\ -0.866 & -1 & 0 & -1 & 0 & 0 \\ -0.5 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & -0.866 & 0 & 0 & -1 \end{bmatrix} \begin{Bmatrix} F_1 \\ F_2 \\ F_3 \\ H_2 \\ V_2 \\ V_3 \end{Bmatrix} = \begin{Bmatrix} 0 \\ -1000 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix}$$

$$\begin{bmatrix} F_{1,h} \\ F_{1,v} \\ F_{2,h} \\ F_{2,v} \\ F_{3,h} \\ F_{3,v} \end{bmatrix}$$

Using any of previous techniques

(LU is preferred for this problem, why?)

(Partial pivoting may be required)

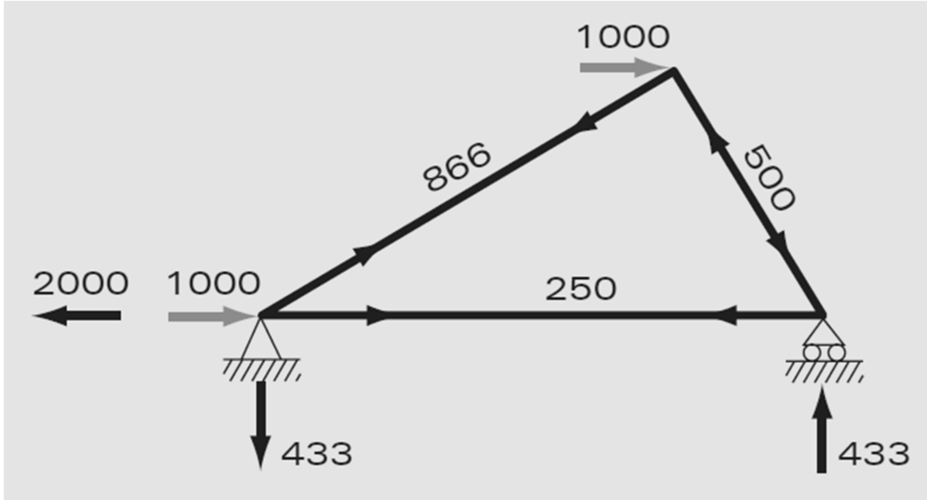
$F_1 = -500 \quad F_2 = 433 \quad F_3 = -866$

$H_2 = 0 \quad V_2 = 250 \quad V_3 = 750$

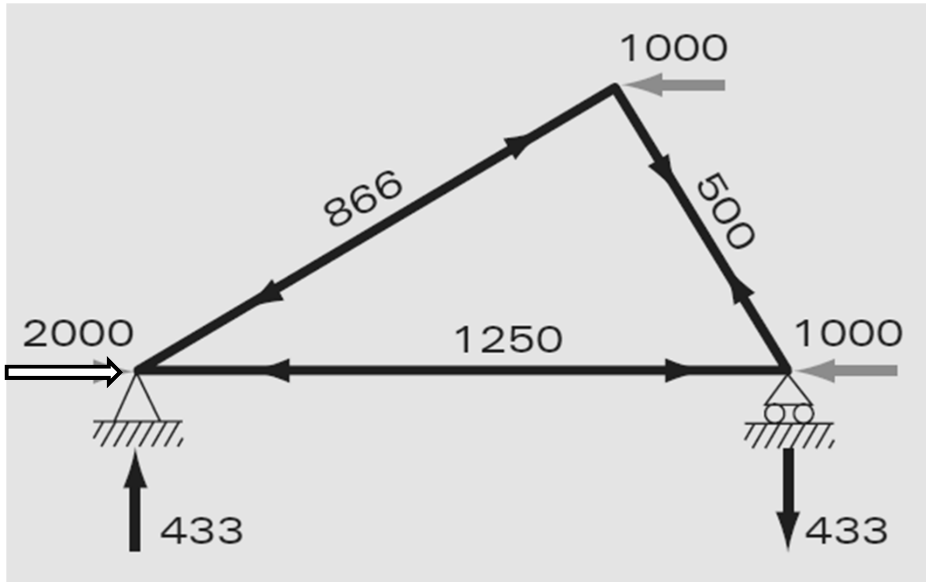
Case Study

$$\{B\} = \begin{bmatrix} F_{1,h} \\ F_{1,v} \\ F_{2,h} \\ F_{2,v} \\ F_{3,h} \\ F_{3,v} \end{bmatrix}$$

$$\{B\} = \begin{bmatrix} 1000 \\ 0 \\ 1000 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$



$$\{B\} = \begin{bmatrix} -1000 \\ 0 \\ 0 \\ 0 \\ -1000 \\ 0 \end{bmatrix}$$



Case Study

$$[A]^{-1} = \begin{bmatrix} 0.866 & 0.5 & 0 & 0 & 0 & 0 \\ 0.25 & -0.433 & 0 & 0 & 1 & 0 \\ -0.5 & 0.866 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 & -1 & 0 \\ -0.433 & -0.25 & 0 & -1 & 0 & 0 \\ 0.433 & -0.75 & 0 & 0 & 0 & -1 \end{bmatrix}$$

$$\begin{Bmatrix} F_1 \\ F_2 \\ F_3 \\ H_2 \\ V_2 \\ V_3 \end{Bmatrix} = \begin{bmatrix} 0.866 & 0.5 & 0 & 0 & 0 & 0 \\ 0.25 & -0.433 & 0 & 0 & 1 & 0 \\ -0.5 & 0.866 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 & -1 & 0 \\ -0.433 & -0.25 & 0 & -1 & 0 & 0 \\ 0.433 & -0.75 & 0 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} F_{1,h} \\ F_{1,v} \\ F_{2,h} \\ F_{2,v} \\ F_{3,h} \\ F_{3,v} \end{bmatrix}$$