LU Decomposition

Ch. 10

LU Decomposition

- What is LU decomposition?
  - Another Class of Elimination methods
- Why do we want to we want to use it?
  - The time consuming elimination step need only be performed on *[A]* NOT *{b}*
  - Situations where *[A]* doesn't change and *{b}* does.
    - Trusses with varying external loads
    - Multiple spring/mass systems with varying masses

Note: Gauss Elimination can be expressed as an LU decomposition

LU Decomposition

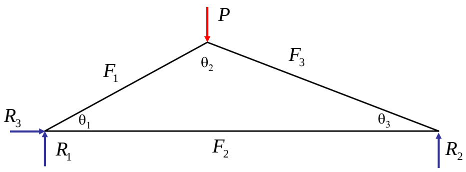
- Trusses with varying loads

$$[A]\{x\} = \{b\}$$

Coefficients defined by the Geometry of the truss (ie., $\cos\theta_1$) → $[A]$

Reaction forces & Internal forces (i.e., F's and R's) → $\{x\}$

External Forces (i.e., P) → $\{b\}$



LU Decomposition

- Trusses with varying loads

$$[A]\{x\}=\{b\}$$

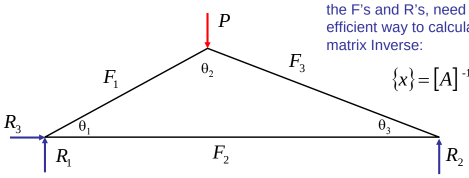
Coefficients defined by the Geometry of the truss (ie., $\cos\theta_1$)

Reaction forces & Internal forces (i.e., F's and R's)

External Forces (i.e., P)

We would like to calculate the F's and R's, need an efficient way to calculate matrix Inverse:

$$\{x\}=[A]^{-1}\{b\}$$



Overview of LU Decomposition

What is LU decomposition?

$$[A]\rightarrow[L][U]$$

Lower Triangular Matrix

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

Upper Triangular Matrix

$$[U]=\begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix}$$

LU Decomposition – A little mathematical Jugglery

Consider: $[A]\{x\}=\{b\}$ (1)

Rewrite (1) as: $[A]\{x\}-\{b\}=0$ (2)

From our Gauss Elimination Method, we know that (1) can be written as

$$\begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix}\begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix}=\begin{Bmatrix} d_1 \\ d_2 \\ d_3 \end{Bmatrix} \quad (3)$$

Rewrite (3) as: $[U]\{x\}-\{d\}=0$ (4)

LU Decomposition – A little mathematical Jugglery

Assume a Lower Tridiagonal Matrix Exists:

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

If *[L]* is assumed to have the following property:

$$[L]([U]\{x\}-\{d\})=[A]\{x\}-\{b\}$$

Then distributing *[L]* gives

$$\underbrace{[L][U]}_{[A]}\{x\}-\underbrace{[L]\{d\}}_{\{b\}}=[A]\{x\}-\{b\}$$

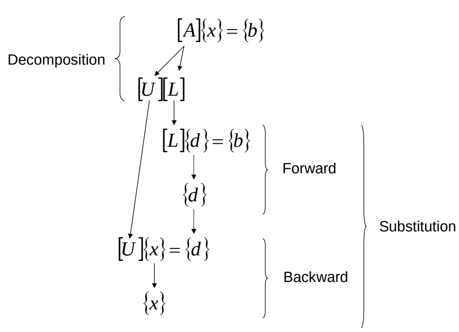
LU Decomposition

We now have the following three equations to use in a solution Strategy:

$$[U]\{x\}-\{d\}=0 \quad (i)$$
$$[L][U]=[A] \quad (ii)$$
$$[L]\{d\}=\{b\} \quad (iii)$$

1. LU decomposition
   - Factor [A] into [L] and [U] (lower & Upper triangular Matrices)
2. Substitution
   a. Use (iii) to generate {d} by forward substitution
   b. Substitute {d} into (i) and use backward substitution to solve for {x}

LU Decomposition



LU Decomposition – Gauss Elimination

Recall, that Gauss Elimination gave:

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

[L] was also produced during the forward elimination procedure, through our "factors". Consider,

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}\begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix}=\begin{Bmatrix} b_1 \\ b_2 \\ b_3 \end{Bmatrix} \quad \begin{matrix} (1) \\ (2) \\ (3) \end{matrix}$$

1. First elimination step, multiply (1) by $f_{21}=\frac{a_{21}}{a_{11}}$ and subtract the result from (2) to Eliminate a_{21}
2. Next, Multiply (1) by $f_{31}=\frac{a_{31}}{a_{11}}$ and subtract the result from (3) to eliminate a_{31}
3. Multiply (2') by $f_{32}=\frac{a'_{32}}{a'_{22}}$ and subtract the result from (3') to eliminate a'_{32}

LU Decomposition – Gauss Elimination

- Now Perform operations only on *[A]*
- Do NOT operate on *{b}*
- Save f_{ij} and operate on {b} later
- Store f_{ij} in the appropriate a_{ij} locations, i.e.,

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

Represents an efficient storage of [L] and [U]

Where,

$$[L]=\begin{bmatrix} 1 & 0 & 0 \\ f_{21} & 1 & 0 \\ f_{31} & f_{32} & 1 \end{bmatrix} \qquad [U]=\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a'_{22} & a'_{23} \\ 0 & 0 & a''_{33} \end{bmatrix}$$

LU Decomposition – Gauss Elimination

Confirm: $[A]\rightarrow[L][U]$

$$[L][U]=\begin{bmatrix} 1 & 0 & 0 \\ f_{21} & 1 & 0 \\ f_{31} & f_{32} & 1 \end{bmatrix}\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a'_{22} & a'_{23} \\ 0 & 0 & a''_{33} \end{bmatrix}=\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

LU Decomposition – Gauss Elimination

Algorithm for Forward & Backward Substitution:

1. Forward Substitution, Solve for *{d}*

$$[L]\{d\} = \{b\}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix} \begin{Bmatrix} d_1 \\ d_2 \\ d_3 \end{Bmatrix} = \begin{Bmatrix} b_1 \\ b_2 \\ b_3 \end{Bmatrix}$$

$$d_1 = b_1 \qquad \textit{for i=1}$$

$$d_i = b_i - \sum_{j=1}^{i-1} l_{ij} d_j \qquad \textit{for i=2,3, …, n}$$

LU Decomposition – Gauss Elimination

2. Backward Substitution:

$$[U]\{x\} = \{d\}$$

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

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

$$x_i = \frac{d_i - \sum_{j=i+1}^{n} u_{ij} x_j}{u_{ii}} \qquad \textit{for i=n-1,n-2, …, 1}$$

Just like in Naïve Gauss Elimination

Simple LU Decomposition Example

Solve the following system of algebraic equations using Gauss Elimination based LU decomposition

$$\begin{bmatrix} 4 & 3 & 1 \\ 3 & -1 & 5 \\ 2 & 4 & 2 \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 4 \\ 1 \end{Bmatrix}$$

Do Little LU Decomposition

- Requires all diagonal terms to be non-zero, $a_{ii} \neq 0$
- Pivoting can be added
- Decompose *[A]* using

(1) $$u_{ij} = a_{ij} - \sum_{k=1}^{i-1} l_{ik} u_{kj} \qquad \textit{for } j=1,2,3,\ldots,n$$

(2) $$l_{ij} = \frac{1}{u_{ii}} \left(a_{ij} - \sum_{k=1}^{j-1} l_{ik} u_{kj} \right) \qquad \textit{for } j=1,2,3,\ldots,i-1$$

- Computation begins by defining
  - $u_{11} = a_{11}$
  - $l_{ii} = 1$
  - $l_{i1} = a_{i1}/u_{11}$ *for* $i = 2, \ldots, n$
- Next, for each $j=2,\ldots,n$ compute u_{ij} and l_{ij} using (1) and (2) for increasing i
- Now, each coefficient is known when it is needed.
- *[U]* and *[L]* are thus found column by column.

Computational Effort

- Gauss Elimination
- Forward Elimination
- Backward Substitution

- LU Decomposition
- Forward Elimination
- Forward Substitution
- Backward Substitution
  - Less Effort during forward decomposition
  - Extra effort to do Forward Substitution

Both techniques require the same effort if only 1 set of {b}'s are use ~ n^3
Benefits from LU decomposition result if you have many {b}'s