Course Instructor
Dr. Raymond C. Rumpf
Office: A-337
Phone: (915) 747-6958
E-Mail: rcrumpf@utep.edu

Computational Methods in EE

Topic 3b – Numerical Linear Algebra

EE 4386/5301 Computational Methods in EE

Outline

Computational Methods in EE

- Direct Solution Methods
  - Naïve Gauss elimination
  - Gauss-Jordan method
  - LU decomposition
- Iterative Solution Methods
  - Jacobi method
  - Gauss-Seidel method
- Calculating matrix inverse

Naïve Gauss Elimination

What is Naïve Gauss Elimination?

Naïve Gauss elimination (GE) is the simplest method for solving a system of equations.

$$\begin{bmatrix} & A & \end{bmatrix}\begin{bmatrix} x \end{bmatrix} = \begin{bmatrix} b \end{bmatrix}$$

While simple, it can be unstable and "pivoting" is required to stabilize it. The Naïve algorithm ignores this.

In fact, we sort of already implemented this when solving our first system of linear equations, we just did not do it in matrix form.

Step 1

We start with a matrix problem…

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

Step 2

Computational
Methods in EE

Eliminate x_1 from rows 2 and 3.

$$(\text{New Row 2}) = (\text{Old Row 2}) - \frac{a_{21}}{a_{11}}(\text{Row 1})$$

$$(\text{New Row 3}) = (\text{Old Row 3}) - \frac{a_{31}}{a_{11}}(\text{Row 1})$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \rightarrow \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a'_{22} & a'_{23} \\ 0 & a'_{32} & a'_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b'_2 \\ b'_3 \end{bmatrix}$$

Step 3

Computational Methods in EE

Eliminate x_2 from row 3.

$$(\text{New Row 3}) = (\text{Old Row 3}) - \frac{a'_{32}}{a'_{22}}(\text{Row 2})$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a'_{22} & a'_{23} \\ 0 & a'_{32} & a'_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b'_2 \\ b'_3 \end{bmatrix} \rightarrow \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a'_{22} & a'_{23} \\ 0 & 0 & a''_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b'_2 \\ b''_3 \end{bmatrix}$$

Step 4

Computational Methods in EE

Now we know x_3.

$$x_3 = \frac{b''_3}{a''_{33}}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a'_{22} & a'_{23} \\ 0 & 0 & a''_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b'_2 \\ b''_3 \end{bmatrix}$$

Step 5

Computational Methods in EE

We back-substitute to find x_1 and x_2.

$$x_2 = \frac{b_2'}{a_{22}'} - \frac{a_{23}'}{a_{22}'} x_3 \qquad x_1 = \frac{b_1}{a_{11}} - \frac{a_{12}}{a_{11}} x_2 - \frac{a_{13}}{a_{11}} x_3$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22}' & a_{23}' \\ 0 & 0 & a_{33}'' \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2' \\ b_3'' \end{bmatrix}$$

Triangular Matrices

Computational Methods in EE

Notice that we get an upper triangular matrix from Gauss elimination.

Triangular matrices represent systems of equations that are "almost" solved.

It is usually a very quick and easy procedure to solve triangular matrices.

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22}' & a_{23}' \\ 0 & 0 & a_{33}'' \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2' \\ b_3'' \end{bmatrix}$$

Observation

Observe that we calculated three special factors

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

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

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

These will arise again in LU decomposition.

Computational Methods in EE

Gauss-Jordan Method

What is the Gauss-Jordan Method?

Computational Methods in EE

The Gauss-Jordan method is a technique to solve

$$\begin{bmatrix} & A & \end{bmatrix}\begin{bmatrix} x \end{bmatrix} = \begin{bmatrix} b \end{bmatrix}$$

or to calculate matrix inverses.

$$\begin{bmatrix} & A & \end{bmatrix}^{-1}$$

It is an excellent technique for solving these problems by hand!

Step 1

Computational Methods in EE

Start with a matrix problem $[A][x] = [b]$.

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

Step 2

Computational Methods in EE

Construct an *augmented matrix* from $[A]$ and $[b]$.

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

$$\begin{bmatrix} [A] & [b] \end{bmatrix} \rightarrow \begin{bmatrix} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{bmatrix}$$

Step 3

Computational Methods in EE

Normalize the first row by dividing by the diagonal element a_{11}.

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & a_{12}/a_{11} & a_{13}/a_{11} & b_1/a_{11} \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{bmatrix}$$

$$\rightarrow \begin{bmatrix} 1 & a'_{12} & a'_{13} & b'_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{bmatrix}$$

$$a'_{12} = \frac{a_{12}}{a_{11}} \qquad a'_{13} = \frac{a_{13}}{a_{11}} \qquad b'_1 = \frac{b_1}{a_{11}}$$

Step 4

Computational Methods in EE

Eliminate x_1 from all other rows.

$$(\text{New Row 2}) = (\text{Old Row 2}) - a_{21}(\text{Row 1})$$
$$(\text{New Row 3}) = (\text{Old Row 3}) - a_{31}(\text{Row 1})$$

$$\begin{bmatrix} 1 & a'_{12} & a'_{13} & b'_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & a'_{12} & a'_{13} & b'_1 \\ 0 & a'_{22} & a'_{23} & b'_2 \\ 0 & a'_{32} & a'_{33} & b'_3 \end{bmatrix}$$

New Row 2

$$a'_{22} = a_{22} - a_{21}a'_{12}$$
$$a'_{23} = a_{23} - a_{21}a'_{13}$$
$$b'_2 = b_2 - a_{21}b'_1$$

New Row 3

$$a'_{32} = a_{32} - a_{31}a'_{12}$$
$$a'_{33} = a_{33} - a_{31}a'_{13}$$
$$b'_3 = b_3 - a_{31}b'_1$$

Topic 3b -- Numerical Linear Algebra 17

Step 5

Computational Methods in EE

Normalize the second row by dividing by the diagonal element a'_{22}.

$$\begin{bmatrix} 1 & a'_{12} & a'_{13} & b'_1 \\ 0 & a'_{22} & a'_{23} & b'_2 \\ 0 & a'_{32} & a'_{33} & b'_3 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & a'_{12} & a'_{13} & b'_1 \\ 0 & 1 & a'_{23}/a'_{22} & b'_2/a'_{22} \\ 0 & a'_{32} & a'_{33} & b'_3 \end{bmatrix}$$

$$\rightarrow \begin{bmatrix} 1 & a'_{12} & a'_{13} & b'_1 \\ 0 & 1 & a''_{23} & b''_2 \\ 0 & a'_{32} & a'_{33} & b'_3 \end{bmatrix}$$

$$a''_{23} = a'_{23}/a'_{22} \qquad b''_2 = b'_2/a'_{22}$$

Topic 3b -- Numerical Linear Algebra 18

Step 6

Computational Methods in EE

Eliminate x_2 from all other rows.

$$(\text{New Row 1}) = (\text{Old Row 1}) - a'_{12}(\text{Row 2})$$
$$(\text{New Row 3}) = (\text{Old Row 3}) - a'_{32}(\text{Row 2})$$

$$\begin{bmatrix} 1 & a'_{12} & a'_{13} & b'_1 \\ 0 & 1 & a''_{23} & b''_2 \\ 0 & a'_{32} & a'_{33} & b'_3 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 0 & a''_{13} & b''_1 \\ 0 & 1 & a''_{23} & b''_2 \\ 0 & 0 & a''_{33} & b''_3 \end{bmatrix}$$

New Row 1

$$a''_{13} = a'_{13} - a'_{12}a''_{23}$$
$$b''_1 = b'_1 - a'_{12}b''_2$$

New Row 3

$$a''_{33} = a'_{33} - a'_{32}a''_{23}$$
$$b''_3 = b'_3 - a'_{32}b''_2$$

Step 7

Computational Methods in EE

Normalize the third row by dividing by the diagonal element a''_{33}.

$$\begin{bmatrix} 1 & 0 & a''_{13} & b''_1 \\ 0 & 1 & a''_{23} & b''_2 \\ 0 & 0 & a''_{33} & b''_3 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 0 & a''_{13} & b''_1 \\ 0 & 1 & a''_{23} & b''_2 \\ 0 & 0 & 1 & b''_3/a''_{33} \end{bmatrix}$$

$$\rightarrow \begin{bmatrix} 1 & 0 & a''_{13} & b''_1 \\ 0 & 1 & a''_{23} & b''_2 \\ 0 & 0 & 1 & b'''_3 \end{bmatrix}$$

$$b'''_3 = b''_3 / a''_{33}$$

Step 8

Eliminate x_3 from all other rows.

$$(\text{New Row 1}) = (\text{Old Row 1}) - a''_{13}(\text{Row 3})$$
$$(\text{New Row 2}) = (\text{Old Row 2}) - a''_{23}(\text{Row 3})$$

$$\begin{bmatrix} 1 & 0 & a''_{13} & b''_1 \\ 0 & 1 & a''_{23} & b''_2 \\ 0 & 0 & 1 & b'''_3 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 0 & 0 & b'''_1 \\ 0 & 1 & 0 & b'''_2 \\ 0 & 0 & 1 & b'''_3 \end{bmatrix}$$

New Row 1

$$b'''_1 = b''_1 - a''_{13} b'''_3$$

New Row 2

$$b'''_2 = b''_2 - a''_{23} b'''_3$$

Step 9

Extract solution from augmented matrix.

$$\begin{bmatrix} 1 & 0 & 0 & b'''_1 \\ 0 & 1 & 0 & b'''_2 \\ 0 & 0 & 1 & b'''_3 \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b'''_1 \\ b'''_2 \\ b'''_3 \end{bmatrix}$$

Example

Computational Methods in EE

$[A]=\begin{bmatrix}1&2&3\\0&4&1\\1&1&0\end{bmatrix}\quad [b]=\begin{bmatrix}16\\12\\2\end{bmatrix}$ Step 1 – Define problem

$[A\ \ b]=\begin{bmatrix}1&2&3&16\\0&4&1&12\\1&1&0&2\end{bmatrix}$ Step 2 – Form augmented matrix

$\begin{bmatrix}1&2&3&16\\0&4&1&12\\1&1&0&2\end{bmatrix}$ Step 3 – Normalize row 1

$\begin{bmatrix}1&2&3&16\\0&4&1&12\\0&-1&-3&-14\end{bmatrix}$ Step 4 – Subtract row 1 from rows 2 and 3

$\begin{bmatrix}1&2&3&16\\0&1&0.25&3\\0&-1&-3&-14\end{bmatrix}$ Step 5 – Normalize row 2

$\begin{bmatrix}1&0&2.5&10\\0&1&0.25&3\\0&0&-2.75&-11\end{bmatrix}$ Step 6 – Subtract row 2 from row 1 and 3

$\begin{bmatrix}1&0&2.5&10\\0&1&0.25&3\\0&0&1&4\end{bmatrix}$ Step 7 – Normalize row 3

$\begin{bmatrix}1&0&0&0\\0&1&0&2\\0&0&1&4\end{bmatrix}$ Step 8 – Subtract row 3 from row 1 and 2

$[x]=\begin{bmatrix}0\\2\\4\end{bmatrix}$ Step 9 – Extract answer from augmented matrix

* Be sure to extract the answer from the augmented matrix. The augmented matrix is not the final answer!

Algorithm for Any Size Matrix

Computational Methods in EE

1. Define $[A]$ and $[b]$
2. Construct augmented matrix

   $[U]=\big[[A]\ \ [b]\big]$

   ```
   U = [ A b ];
   ```
3. Iterate through all rows (m)
   a) Normalize mth row by dividing by diagonal element.

      $[U]_{\text{row } m}=[U]_{\text{row } m}\div a_{mm}$

      ```
      U(m,:) = U(m,:)/U(m,m);
      ```
   b) Iterate through all other rows (r), skipping the mth row
      i. Subtract row m from row r

         $[U]_{\text{row } r}=[U]_{\text{row } r}-a_{rm}\cdot[U]_{\text{row } m}$

         ```
         U(r,:) = U(r,:) - U(r,m)*U(m,:);
         ```
4. Extract $[x]$ from augmented matrix

   $[x]=[U]_{\text{column } M+1}$

   ```
   x = U(:,M+1);
   ```

How to Find Matrix Inverses

Computational Methods in EE

1. Define $[A]$
2. Construct augmented matrix

$$[U] = \begin{bmatrix} [A] & [I] \end{bmatrix}$$

3. Perform Gauss-Jordan method (iterate through all rows)
4. Extract $[A]^{-1}$ from augmented matrix

$$[U'] = \begin{bmatrix} [I] & [A]^{-1} \end{bmatrix}$$

Computational Methods in EE

LU Decomposition

Determining the Upper Triangular Matrix $[U]$

Computational Methods in EE

We start with

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \quad \rightarrow \quad [A][x]=[b]$$

At some point during GE, we converted this system of equations to an upper-triangular matrix $[U]$.

$$\begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} \quad \rightarrow \quad [U][x]=[d]$$

Determining the Lower Triangular Matrix $[L]$

Computational Methods in EE

There exists a lower-triangular matrix $[L]$ such that

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

This equation is why the method is called LU decomposition.

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

Recall these special terms.

We now substitute this decomposition into our original equation.

$$[A][x]=[b]$$
$$[A][x]-[b]=[0]$$
$$[L][U][x]-[b]=[0]$$
$$[L]\left([U][x]-[L]^{-1}[b]\right)=[0]$$
$$[L]\left([U][x]-[d]\right)=[0] \qquad \text{where } [d]=[L]^{-1}[b]$$

Algorithm

Computational Methods in EE

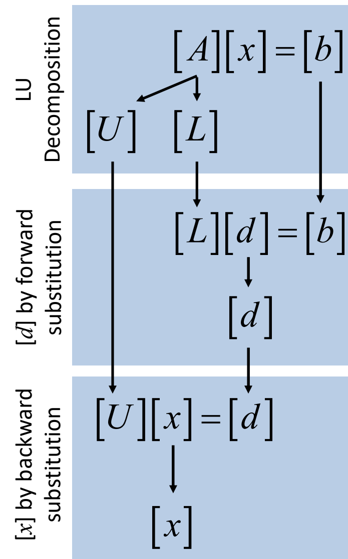
Step 1 – Decompose $[A]$ into $[L]$ and $[U]$.

a) Use GE to calculate $[U]$
b) Store l terms during GE
c) Build $[L]$ from l terms.
d) store $[L]$ and $[U]$ together as

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

Step 2 – Solve $[L][d]=[b]$ using simple forward substitution.

Step 3 – Solve $[U][x]=[d]$ using simple backward substitution.



Jacobi Method

Computational Methods in EE

What is the Jacobi Method?

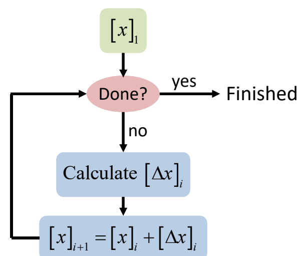
Computational Methods in EE

The Gauss-Jordan method was a direct solution of $[A][x]=[b]$.

This can be inefficient for large matrices, especially when a good initial guess $[x]$ is known.

We can create an iterative algorithm that improves the initial guess every iteration.

The method only converges for *diagonally dominant* matrices.
The algorithm is very picky about this!



Diagonally Dominant Matrices

Computational Methods in EE

A square matrix is said to be diagonally dominant if for each row, the magnitude of the diagonal element is greater than the sum of the magnitudes of all other elements in that row.

$$|a_{ii}| > \sum_{j \neq i} |a_{ij}|$$

Examples

$$[A] = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

Not diagonally dominant. ☹

$$[A] = \begin{bmatrix} 3 & 1 & 1 \\ 0 & 1 & 1 \\ 2 & 1 & 4 \end{bmatrix}$$

Not diagonally dominant. ☹

$$[A] = \begin{bmatrix} -4 & 2 & 1 \\ -1 & -3 & 1 \\ -3 & -2 & -8 \end{bmatrix}$$

Diagonally dominant! ☺

$$[A] = \begin{bmatrix} 2 & 0 & 1 \\ -1 & 2 & 2 \\ 5 & -3 & -8 \end{bmatrix}$$

Not diagonally dominant. ☹

Formulation (1 of 2)

Computational Methods in EE

Our matrix problem is

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

We expand this into its component equations.

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

Formulation (2 of 2)

Computational Methods in EE

We solve the first equation for x_1, the second equation for x_2, and the third equation for x_3.

These are the equations that we will iterate.

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

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

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

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

Implementation (1 of 2)

Computational Methods in EE

Step 1 – Define Problem

$$[A]=\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \qquad [b]=\begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

Step 2 – Come up with an initial guess for $[x]$.

$$x_1^{(1)},\ x_2^{(1)},\ \text{and}\ x_3^{(1)}$$

Implementation (2 of 2)

Computational Methods in EE

Step 3 – Iterate $[x]$ until convergence

a) Calculate new values of $[x]$

$$x_1^{(i+1)}=\frac{b_1-a_{12}x_2^{(i)}-a_{13}x_3^{(i)}}{a_{11}} \qquad x_2^{(i+1)}=\frac{b_2-a_{21}x_1^{(i)}-a_{23}x_3^{(i)}}{a_{22}} \qquad x_3^{(i+1)}=\frac{b_3-a_{31}x_1^{(i)}-a_{32}x_2^{(i)}}{a_{33}}$$

b) Check how much the values have changed

$$\varepsilon_1^{(i+1)}=\left|\frac{x_1^{(i+1)}-x_1^{(i)}}{x_1^{(i+1)}}\right| \qquad \varepsilon_2^{(i+1)}=\left|\frac{x_2^{(i+1)}-x_2^{(i)}}{x_2^{(i+1)}}\right| \qquad \varepsilon_3^{(i+1)}=\left|\frac{x_3^{(i+1)}-x_3^{(i)}}{x_3^{(i+1)}}\right|$$

c) Continue to iterate until all values of ε are sufficiently small.

Matrix Formulation (1 of 2)

Computational Methods in EE

Let's rearrange our update equations this way

$$x_1 = \frac{b_1 - a_{12}x_2 - a_{13}x_3}{a_{11}} \quad \rightarrow \quad x_1 = \frac{1}{a_{11}}\left[b_1 - \left(a_{12}x_2 + a_{13}x_3\right)\right]$$

$$x_2 = \frac{b_2 - a_{21}x_1 - a_{23}x_3}{a_{22}} \quad \rightarrow \quad x_2 = \frac{1}{a_{22}}\left[b_2 - \left(a_{21}x_1 + a_{23}x_3\right)\right]$$

$$x_3 = \frac{b_3 - a_{31}x_1 - a_{32}x_2}{a_{33}} \quad \rightarrow \quad x_3 = \frac{1}{a_{33}}\left[b_3 - \left(a_{31}x_1 + a_{32}x_2\right)\right]$$

By inspecting these equations, we can write the update equation as

$$[x]_{i+1} = \left(\text{diag}[A]\right)^{-1} \cdot \left\{[b] - \left([A] - \text{diag}[A]\right)[x]_i\right\}$$

$$\text{diag}[A] = \begin{bmatrix} a_{11} & & & \\ & a_{22} & & \\ & & \ddots & \\ & & & a_{MM} \end{bmatrix}$$

Matrix Formulation (2 of 2)

Computational Methods in EE

For simplicity, let

$$[D] = \text{diag}[A]$$

Rearrange the "update equation" so that it makes more intuitive sense.

$$\begin{aligned} [x]_{i+1} &= [D]^{-1} \cdot \left\{[b] - \left([A] - [D]\right)[x]_i\right\} \\ &= [D]^{-1}[b] - [D]^{-1}[A][x]_i + [D]^{-1}[D][x]_i \\ &= [x]_i + \underbrace{[D]^{-1}\left([b] - [A][x]_i\right)} \end{aligned}$$

Improvement on $[x]_i$

Previous solution

Matrix Implementation

Computational Methods in EE

1. Define $[A]$ and $[b]$
2. Make initial guess $[x]_1$
3. Extract diagonal of $[A]$

   $[D] = \text{diag}[A]$

```
D = diag(diag(A));
```

4. Iterate until convergence
   a) Calculate adjustment of $[x]_i$

   $$[\Delta x]_i = [D]^{-1}\left([b]-[A][x]_i\right)$$

```
dx = D\(b - A*x);
```

   b) Adjust $[x]_i$

   $$[x]_{i+1} = [x]_i + [\Delta x]_i$$

```
x = x + dx;
```

   c) Compute relative error

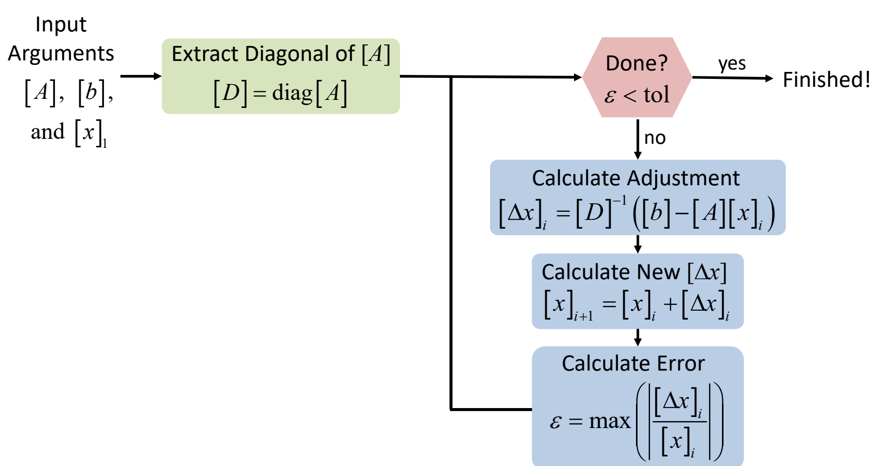
   $$\varepsilon = \max\left(\left|\frac{[\Delta x]_i}{[x]_i}\right|\right)$$

```
err = max(abs(dx./x));
```

   d) Continue iteration until ε is sufficiently small

Block Diagram of Jacobi Method

Computational Methods in EE



Computational Methods in EE

Gauss-Seidel Method

Gauss-Seidel Method

Computational Methods in EE

The Jacobi method required $[A]$ to be diagonally dominant, which restricts what the method can be used to solve.

The Gauss-Seidel method is a modification to the Jacobi method to overcome this limitation.

Formulation (1 of 4)

Computational Methods in EE

The matrix $[A]$ can be decomposed into the sum of a lower triangular matrix $[L']$ and an upper triangular matrix $[U']$.

$$[A]=[L']+[U']$$

$$[A]=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \vdots & \vdots & \ddots & \\ a_{N1} & a_{N2} & & a_{NN} \end{bmatrix} \quad [L']=\begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ a_{21} & a_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \\ a_{N1} & a_{N2} & & a_{NN} \end{bmatrix} \quad [U']=\begin{bmatrix} 0 & a_{12} & \cdots & a_{1N} \\ 0 & 0 & \cdots & a_{2N} \\ \vdots & \vdots & \ddots & \\ 0 & 0 & & 0 \end{bmatrix}$$

Note: $[L']$ and $[U']$ here are not the same $[L]$ and $[U]$ that we used in LU decomposition.

Formulation (2 of 4)

Computational Methods in EE

We are trying to solve $[A][x]=[b]$. Given that $[A] = [L'] + [U]$, we get

$$[A][x]=[b]$$

$$\left([L']+[U']\right)[x]=[b]$$

$$[L'][x]+[U'][x]=[b]$$

Formulation (3 of 4)

Computational Methods in EE

From our prior experience, we know that $[L'][x]=[b]$ is fast to solve for $[x]$ using forward-substitution.

We rearrange our matrix equation to take advantage of this.

$$[L'][x]+[U'][x]=[b]$$

$$[L'][x]=[b]-[U'][x]$$

$$[x]=\underbrace{[L']^{-1}\left([b]-[U'][x]\right)}_{\text{This can be solved very fast!}}$$

Formulation (4 of 4)

Computational Methods in EE

We derive our update equation from the last expression.

$$[x]=[L']^{-1}\left([b]-[U'][x]\right)$$

$$\downarrow$$

$$\boxed{[x]_{i+1}=[L']^{-1}\left([b]-[U'][x]_i\right)}$$

Calculating Matrix Inverse

Computational Methods in EE

Using Gauss-Jordan Method

Computational Methods in EE

Given matrix $[A]$, we form an augmented matrix

$$\big[[A]\ \ [I]\big] \rightarrow \begin{bmatrix} a_{11} & a_{12} & a_{13} & | & 1 & 0 & 0 \\ a_{21} & a_{22} & a_{23} & | & 0 & 1 & 0 \\ a_{31} & a_{32} & a_{33} & | & 0 & 0 & 1 \end{bmatrix}$$

Use the Gauss-Jordan method until the augmented matrix has the form

$$\begin{bmatrix} 1 & 0 & 0 & | & c_{11} & c_{12} & c_{13} \\ 0 & 1 & 0 & | & c_{21} & c_{22} & c_{23} \\ 0 & 0 & 1 & | & c_{31} & c_{32} & c_{33} \end{bmatrix}$$

Here is the matrix inverse.

$$[A]^{-1} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix}$$

Using LU Decomposition

Computational Methods in EE

We apply the LU decomposition method on a column-by-column basis. Each solution is a column in the inverse matrix.

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad \rightarrow \quad \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} c_{11} \\ c_{21} \\ c_{31} \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad \rightarrow \quad \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} c_{12} \\ c_{22} \\ c_{32} \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \rightarrow \quad \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} c_{13} \\ c_{23} \\ c_{33} \end{bmatrix}$$

$$[A]^{-1} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix}$$

Topic 3b -- Numerical Linear Algebra 49