Introduction to Simulation - Lecture 3

Basics of Solving Linear Systems

Jacob White

Thanks to Deepak Ramaswamy, Michal Rewienski, Karen Veroy and Jacob White

Outline

Solution Existence and Uniqueness

Gaussian Elimination Basics

- LU factorization
- Pivoting and Growth

Hard to solve problems

- Conditioning

Application Problems

$$\underbrace{\left[\ G\ \right]}_{M}\ \underbrace{V_n}_{x} = \underbrace{I_s}_{b}$$

- No voltage sources or rigid struts
- Symmetric and Diagonally Dominant
- Matrix is n x n

Systems of Linear Equations

$$\begin{bmatrix} \uparrow & \uparrow & \cdots & \uparrow \\ \vec{M}_1 & \vec{M}_2 & \cdots & \vec{M}_N \\ \downarrow & \downarrow & \cdots & \downarrow \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{bmatrix}$$

$$\Downarrow$$

$$x_1\vec{M}_1 + x_2\vec{M}_2 + \cdots + x_N\vec{M}_N = b$$

Find a set of weights, x, so that the weighted sum of the columns of the matrix M is equal to the right hand side b

Systems of Linear Equations

Key Questions

- Given Mx = b
  - Is there a solution?
  - Is the solution Unique?
- Is there a Solution?

There exists weights, $x_1, \cdots x_N,$ such that

$$x_1\vec{M}_1 + x_2\vec{M}_2 + \cdots + x_N\vec{M}_N = b$$

A solution exists when b is in the span of the columns of M

SMA-HPC ©2003 MIT

Systems of Linear Equations

Key Questions Continued

- Is the Solution Unique?

Suppose there exists weights, $y_1, \cdots y_N$, not all zero

$$y_1 \vec{M}_1 + y_2 \vec{M}_2 + \cdots + y_N \vec{M}_N = 0$$

Then if $Mx = b$, therefore $M(x+y) = b$

A solution is unique only if the columns of M are linearly independent.

Systems of Linear Equations

Key Questions

Square Matrices

- Given Mx = b, where M is square
  - If a solution exists for any b, then the solution for a specific b is unique.

For a solution to exist for any b, the columns of M must span all N-length vectors. Since there are only N columns of the matrix M to span this space, these vectors must be linearly independent.

A square matrix with linearly independent columns is said to be nonsingular.

Gaussian Elimination Basics

Important Properties

Gaussian Elimination Method for Solving $Mx = b$

- A "Direct" Method
  Finite Termination for exact result (ignoring roundoff)
- Produces accurate results for a broad range of matrices
- Computationally Expensive

Gaussian Elimination Basics

Reminder by Example

3 x 3 example

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

$$\Downarrow$$

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

Gaussian Elimination Basics

Reminder by Example

Key Idea

Use Eqn 1 to Eliminate x_1 From Eqn 2 and 3

$$M_{11}x_1 + M_{12}x_2 + M_{13}x_3 = b_1$$

$$\left(M_{22} - \frac{M_{21}}{M_{11}}M_{12}\right)x_2 + \left(M_{23} - \frac{M_{21}}{M_{11}}M_{13}\right)x_3 = b_2 - \frac{M_{21}}{M_{11}}b_1$$

$$\left(M_{32} - \frac{M_{31}}{M_{11}}M_{12}\right)x_2 + \left(M_{33} - \frac{M_{31}}{M_{11}}M_{13}\right)x_3 = b_3 - \frac{M_{31}}{M_{11}}b_1$$

Gaussian Elimination Basics

Reminder by Example

Key Idea in the Matrix

Pivot

MULTIPLIERS

$$\begin{bmatrix} M_{11} & M_{12} & M_{13} \\ 0 & \left(M_{22} - \left(\frac{M_{21}}{M_{11}}\right) M_{12}\right) & \left(M_{23} - \left(\frac{M_{21}}{M_{11}}\right) M_{12}\right) \\ 0 & \left(M_{32} - \left(\frac{M_{31}}{M_{11}}\right) M_{12}\right) & \left(M_{33} - \left(\frac{M_{31}}{M_{11}}\right) M_{12}\right) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 - \left(\frac{M_{21}}{M_{11}}\right) b_1 \\ b_3 - \left(\frac{M_{31}}{M_{11}}\right) b_1 \end{bmatrix}$$

Gaussian Elimination Basics

Reminder by Example

Remove x2 from Eqn 3

Pivot

$$\begin{bmatrix} M_{11} & M_{12} & M_{13} \\ 0 & \left(M_{22} - \left(\frac{M_{21}}{M_{11}}\right)M_{12}\right) & \left(M_{23} - \left(\frac{M_{21}}{M_{11}}\right)M_{12}\right) \\ 0 & 0 & \left(M_{33} - \left(\frac{M_{31}}{M_{11}}\right)M_{12} - \frac{\left(M_{32} - \left(\frac{M_{31}}{M_{11}}\right)M_{12}\right)}{\left(M_{22} - \left(\frac{M_{21}}{M_{11}}\right)M_{12}\right)}\left(M_{23} - \left(\frac{M_{21}}{M_{11}}\right)M_{12}\right)\right) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 - \left(\frac{M_{21}}{M_{11}}\right)b_1 \\ b_3 - \left(\frac{M_{31}}{M_{11}}\right)b_1 - \frac{\left(M_{32} - \left(\frac{M_{31}}{M_{11}}\right)M_{12}\right)}{\left(M_{22} - \left(\frac{M_{21}}{M_{11}}\right)M_{12}\right)}\left(b_2 - \left(\frac{M_{21}}{M_{11}}\right)b_1\right) \end{bmatrix}$$

Multiplier

Gaussian Elimination Basics

Reminder by Example

Simplify the Notation

$$\begin{bmatrix} M_{11} & M_{12} & M_{13} \\ 0 & \tilde{M}_{22} & \tilde{M}_{23} \\ 0 & \tilde{M}_{32} & \tilde{M}_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ \tilde{b}_2 \\ \tilde{b}_3 \end{bmatrix}$$

Gaussian Elimination Basics

Reminder by Example

Remove x2 from Eqn 3

Pivot

$$\begin{bmatrix} M_{11} & M_{12} & M_{13} \\ 0 & \tilde{M}_{22} & \tilde{M}_{23} \\ 0 & 0 & \tilde{M}_{33} - \frac{\tilde{M}_{32}}{\tilde{M}_{22}} \tilde{M}_{23} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ \tilde{b}_2 \\ \tilde{b}_3 - \frac{\tilde{M}_{32}}{\tilde{M}_{22}} \tilde{b}_2 \end{bmatrix}$$

Multiplier

Gaussian Elimination Basics

Reminder by Example

GE Yields Triangular System

$$\begin{bmatrix} U_{11} & U_{12} & U_{13} \\ 0 & U_{22} & U_{23} \\ 0 & 0 & U_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

$$x_3 = \frac{y_3}{U_{33}}$$

$$x_2 = \frac{y_2 - U_{23} x_3}{U_{22}}$$

$$x_1 = \frac{y_1 - U_{12} x_2 - U_{13} x_3}{U_{11}}$$

Gaussian Elimination Basics

Reminder by Example

The right-hand side updates

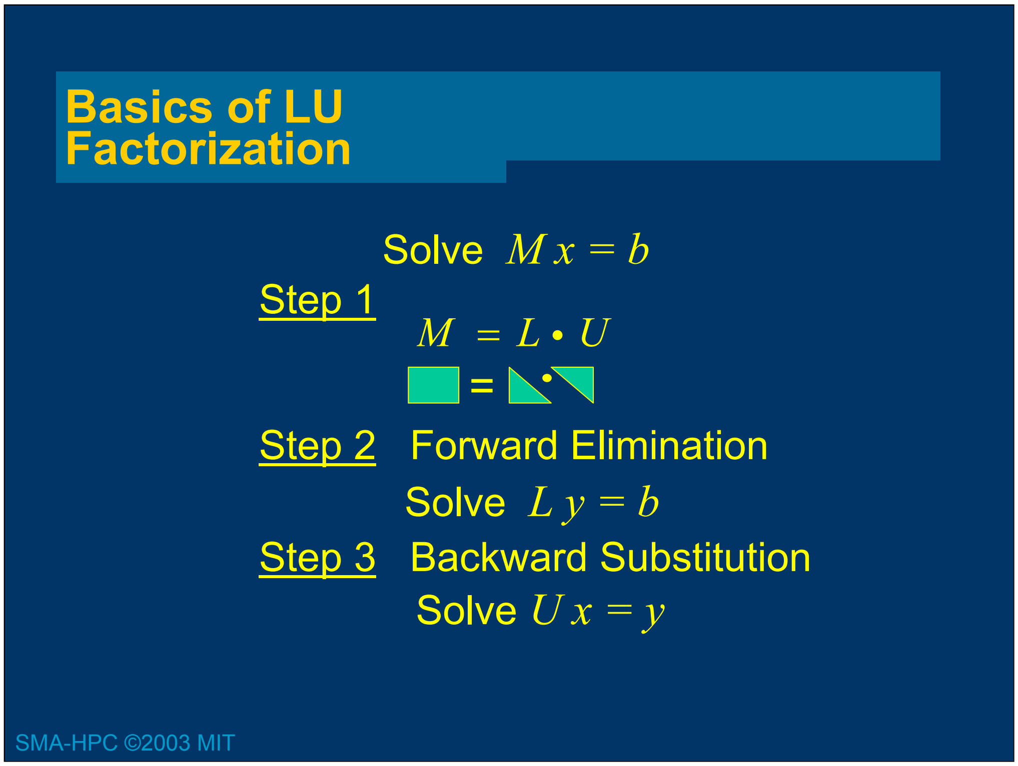
$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 - \left(\dfrac{M_{21}}{M_{11}}\right) b_1 \\ b_3 - \left(\dfrac{M_{31}}{M_{11}}\right) b_1 - \dfrac{\tilde{M}_{32}}{\tilde{M}_{22}} \tilde{b}_2 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 0 & 0 \\ \dfrac{M_{21}}{M_{11}} & 1 & 0 \\ \dfrac{M_{31}}{M_{11}} & \dfrac{\tilde{M}_{32}}{\tilde{M}_{22}} & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

Gaussian Elimination Basics

Reminder by Example

Fitting the pieces together

$$\begin{bmatrix} M_{11} & M_{12} & M_{13} \\ \dfrac{M_{21}}{M_{11}} & \tilde{M}_{22} & \tilde{M}_{23} \\ \dfrac{M_{31}}{M_{11}} & \dfrac{\tilde{M}_{32}}{\tilde{M}_{22}} & \tilde{M}_{33} \end{bmatrix}$$



Recall from basic linear algebra that a matrix A can be factored into the product of a lower and an upper triangular matrix using Gaussian Elimination. The basic idea of Gaussian elimination is to use equation one to eliminate x1 from all but the first equation. Then equation two is used to eliminate x2 from all but the second equation. This procedure continues, reducing the system to upper triangular form as well as modifying the right-hand side. More is

needed here on the basics of Gaussian elimination.

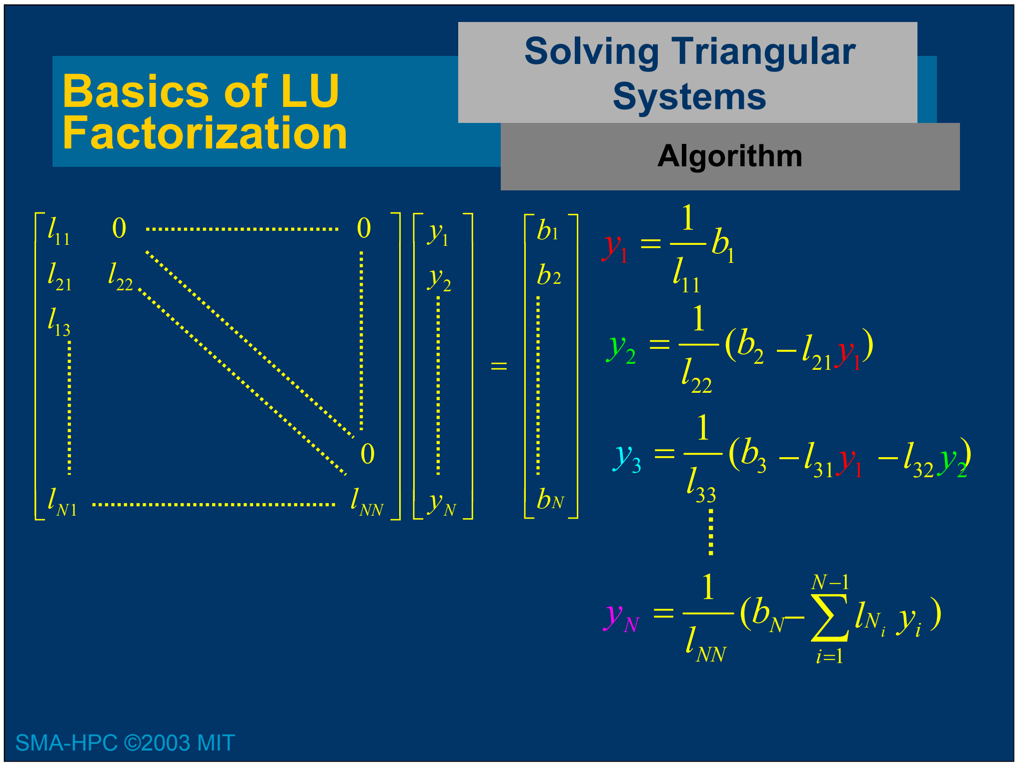
Basics of LU Factorization

Solving Triangular Systems

Matrix

$$\begin{bmatrix} l_{11} & 0 & \cdots & 0 \\ l_{21} & l_{22} & \ddots & \vdots \\ l_{13} & & \ddots & 0 \\ \vdots & & & \\ l_{N1} & \cdots & & l_{NN} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{bmatrix}$$

The First Equation has only y_1 as an unknown.
The Second Equation has only y_1 and y_2 as unknowns.



Solving a triangular system of equations is straightforward but expensive. y_1 can be computed with one divide, y_2 can be computed with a multiply, a subtraction and a divide. Once y_{k-1} has been computed, y_k can be computed with $k-1$ multiplies, $k-2$ adds, a subtraction and a divide. Roughly the number of arithmetic operations is

N divides + 0 add /subs + 1 add /subs + ……. N- 1 add /sub

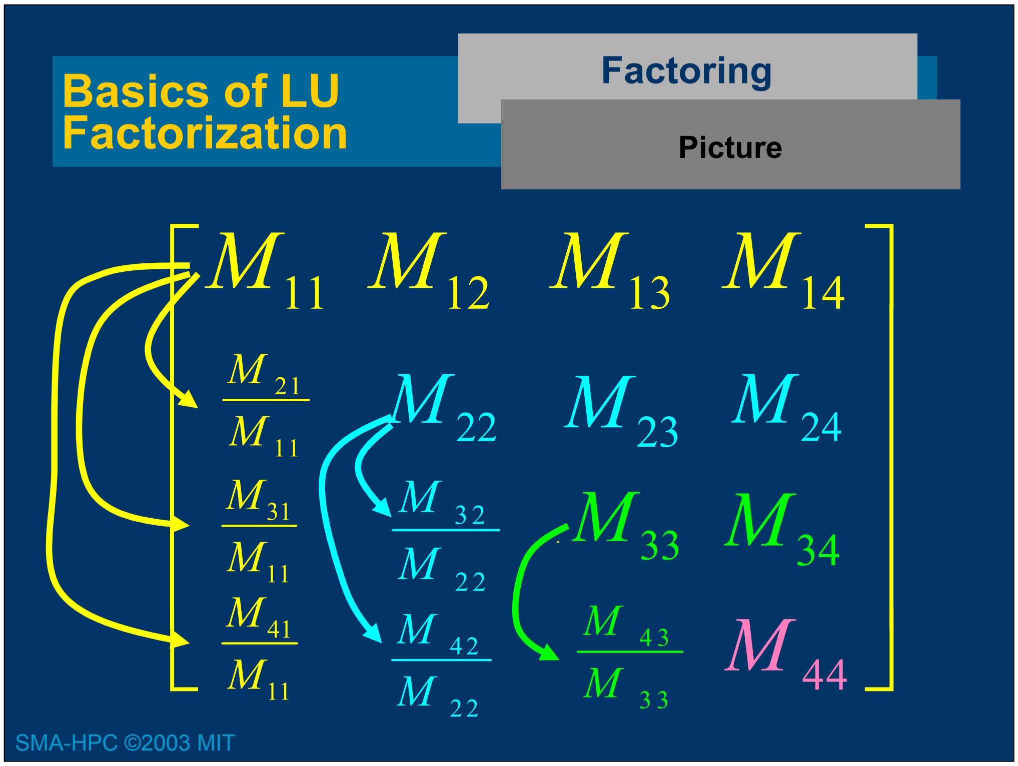
for y_1 ↑ for y_2 ↑ for y_N ↑

+ 0 mults + 1 mult + …….. + N- 1 mults

for y_1 ↑ for y_2 ↑ for y_N ↑

= $(N-1)(N-2)/2$ add/subs + $(N-1)(N-2)/2$ mults + N divides

= Order N^2 operations



The above is an animation of LU factorization. In the first step, the first equation is used to eliminate x1 from the 2nd through 4th equation. This involves multiplying row 1 by a multiplier and then subtracting the scaled row 1 from each of the target rows. Since such an operation would zero out the a21, a31 and a41 entries, we can replace those zero'd entries with the scaling factors, also called the multipliers. For row 2, the scale factor is a21/a11 because if one multiplies row 1 by a21/a11 and then subtracts the result from row 2, the resulting a21 entry would be zero. Entries a22, a23 and a24 would also be modified during the subtraction and this is noted by changing the color of these matrix entries to blue. As row 1 is used to zero a31 and a41, a31 and a41 are replaced by multipliers. The remaining entries in rows 3 and 4 will be modified during this process, so they are recolored blue.

This factorization process continues with row 2. Multipliers are generated so that row 2 can be used to eliminate x2 from rows 3 and 4, and these multipliers are stored in the zero'd locations. Note that as entries in rows 3 and 4 are modified during this process, they are converted to gr een. The final step is to used row 3 to eliminate x3 from row 4, modifying row 4's entry, which is denoted by converting a44 to pink.

It is interesting to note that as the multipliers are standing in for zero'd matrix entries, they are not modified during the factorization.

LU Basics

Factoring

Algorithm

For i = 1 to n-1 {   "For each Row"

For j = i+1 to n {   "For each target Row below the source"

$$M_{ji} = \frac{M_{ji}}{M_{ii}}$$ Pivot

For k = i+1 to n { "For each Row element beyond Pivot"

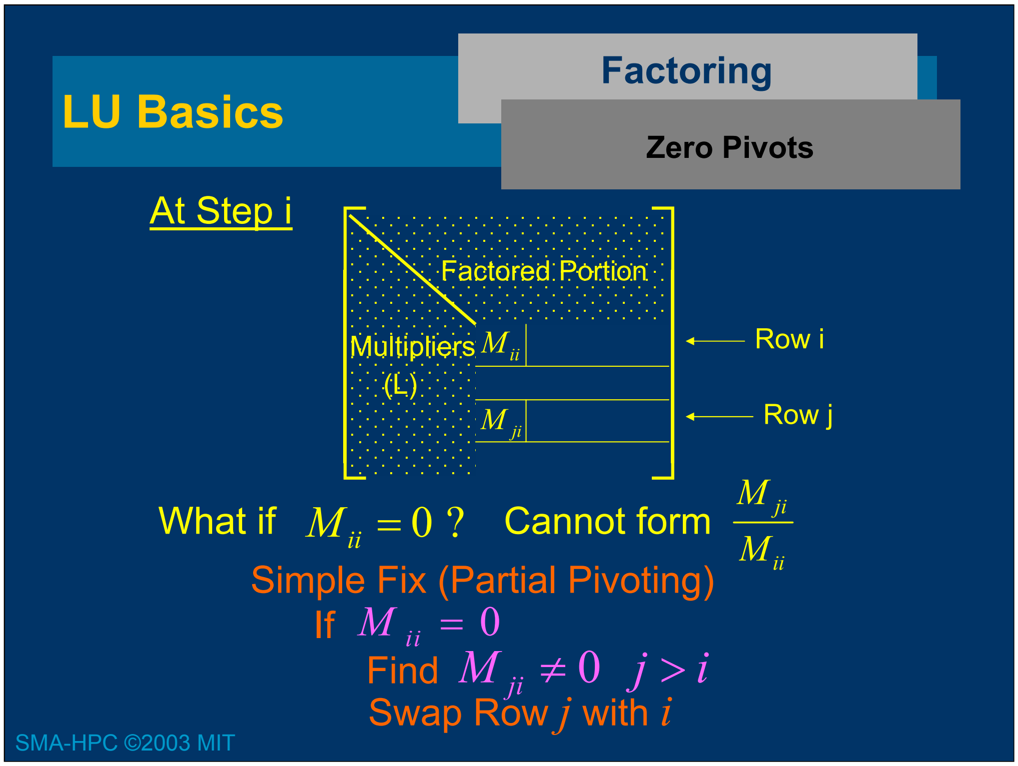
$$M_{jk} \leftarrow M_{jk} - M_{ji} M_{ik}$$

Multiplier

}

}

}

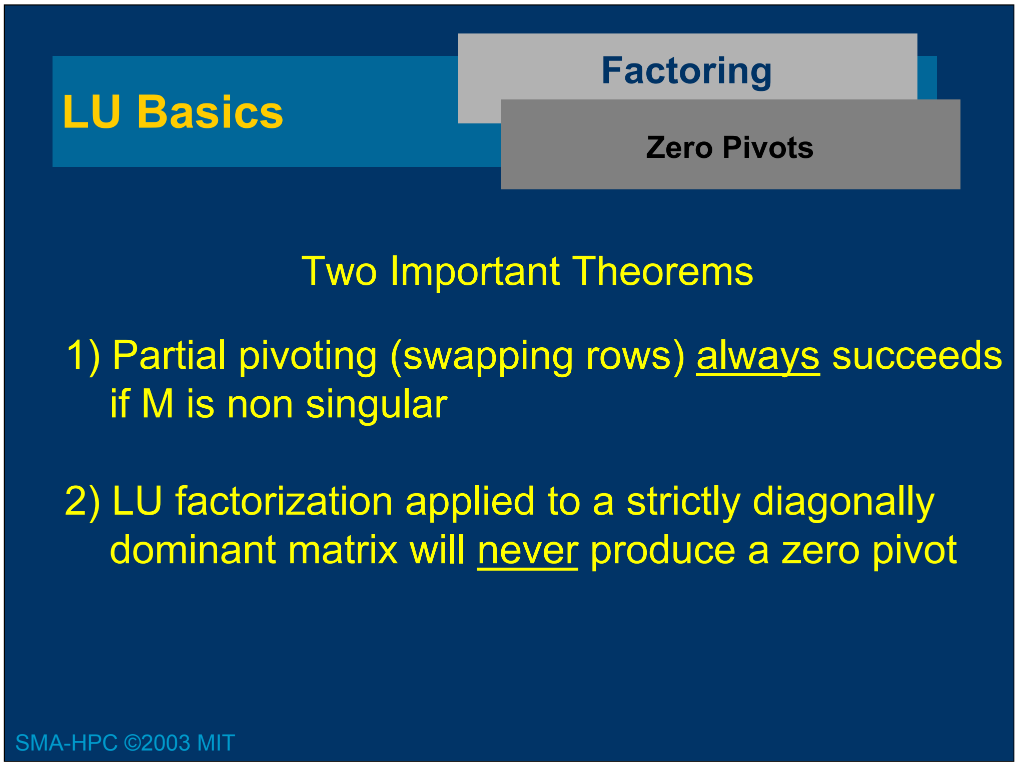


Swapping row j with row i at step i of LU factorization is identical to applying LU to a matrix with its rows swapped "a priori".

To see this consider swapping rows before beginning LU.

$$
\begin{bmatrix}
- - - - - - - - - \; r_1 \; - - - - - - - - - \\
- - - - - - - - - \; r_2 \; - - - - - - - - - \\
\\
- - - - - - - - - \; r_j \; - - - - - - - - - \\
\\
- - - - - - - - - \; r_i \; - - - - - - - - - \\
\\
- - - - - - - - - \; r_N \; - - - - - - - - -
\end{bmatrix}
\begin{bmatrix} x_1 \\ x_2 \\ | \\ x_i \\ | \\ x_j \\ | \\ x_N \end{bmatrix}
=
\begin{bmatrix} b_1 \\ b_2 \\ | \\ b_j \\ | \\ b_i \\ | \\ b_N \end{bmatrix}
$$

Swapping rows corresponds to reordering only the equations. Notice that the vector of unknowns is NOT reordered.

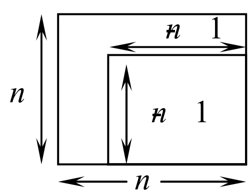


Theorem Gaussian Elimination applied to strictly diagonally dominant matrices will never produce a zero pivot.

Proof

1) Show the first step succeeds.

2) Show the $(n-1) \times (n-1)$ sub matrix



is still strictly diagonally dominant.

First Step $a_{11} \neq 0$ as $|a_{11}| > \sum_{j=2}^{n} |a_{ij}|$

Second row after first step

$$0,\ a_{22} - \frac{a_{21}}{a_{11}} a_{12},\ a_{23} - \frac{a_{21}}{a_{11}} a_{13}, \ldots,\ a_{2n} - \frac{a_{21}}{a_{11}} a_{1n}$$

Is

$$\left| a_{22} - \frac{a_{21}}{a_{11}} a_{12} \right| > \left| a_{2j} - \frac{a_{21}}{a_{11}} a_{1j} \right| ?$$

LU Basics

Numerical Problems

Small Pivots

Contrived Example

$$\begin{bmatrix} 10^{-17} & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \end{bmatrix}$$

$$\mathrm{L} = \begin{bmatrix} 1 & 0 \\ 10^{17} & 1 \end{bmatrix} \qquad \mathrm{U} = \begin{bmatrix} 10^{-17} & 1 \\ 0 & \underbrace{-10^{17}+2}_{\text{Can we represent this ?}} \end{bmatrix}$$

In order to compute the exact solution first forward eliminate

$$\begin{bmatrix} 1 & 0 \\ 10^{17} & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \end{bmatrix}$$

and therefore $y_1 = 1,\ y_2 = 3 - 10^{17}$.

Backward substitution yields

$$\begin{bmatrix} 10^{-17} & 1 \\ 0 & 2-10^{17} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 3-10^{17} \end{bmatrix}$$

and therefore $x_2 = \dfrac{3-10^{17}}{2-10^{17}} \simeq +1$

and $x_1 = 10^{17}(1 - \dfrac{3-10^{17}}{2-10^{17}}) = 10^{17}(\dfrac{2-10^{17}-(3-10^{17})}{2-10^{17}}) \simeq +1$

In the rounded case

$$\begin{bmatrix} 1 & 0 \\ 10^{17} & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \end{bmatrix} \qquad y_1 = 1 \quad y_2 = -10^{17}$$

$$\begin{bmatrix} 10^{-17} & 1 \\ 0 & 10^{17} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ -10^{17} \end{bmatrix} \qquad x_1 = 1 \quad x_2 = 0$$

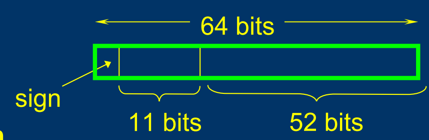
LU Basics

Numerical Problems

Small Pivots

An Aside on Floating Point Arithmetic

Double precision number $X.XXXX\ldots X \bullet 10^{\text{exponent}}$



Basic Problem

$$1.0 - 0.000000000000001 = 1.0$$

or

$$1.0000001 - 1 + (\pi \cdot 10^{-7}) \ \text{ right 8 digits}$$

Key Issue

Avoid small differences between large numbers !

LU Basics

Numerical Problems

Small Pivots

Back to the contrived example

$$LU_{\text{Exact}} = \begin{bmatrix} 1 & 0 \\ 10^{17} & 1 \end{bmatrix} \begin{bmatrix} 10^{-17} & 1 \\ 0 & 2-10^{17} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \end{bmatrix}$$

$$LU_{\text{Rounded}} = \begin{bmatrix} 1 & 0 \\ 10^{17} & 1 \end{bmatrix} \begin{bmatrix} 10^{-17} & 1 \\ 0 & -10^{17} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \end{bmatrix}$$

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{\text{Exact}} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \qquad \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{\text{Rounded}} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

LU Basics

Numerical Problems

Small Pivots

Partial Pivoting for Roundoff reduction

$$\text{If } |M_{ii}| < \max_{j>i} |M_{ji}|$$

$$\text{Swap row } i \text{ with } \arg\left(\max_{j>i} |M_{ij}|\right)$$

$$LU_{\text{reordered}} = \begin{bmatrix} 1 & 0 \\ 10^{-17} & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 0 & \underbrace{1+2\cdot 10^{-17}}_{\text{This term gets rounded}} \end{bmatrix}$$

This multiplier is small

This term gets rounded

SMA-HPC ©2003 MIT

To see why pivoting helped notice that

$$\begin{bmatrix} 1 & 0 \\ 10^{-17} & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}$$

yields $y_1 = 3,\ y_2 = 1 - 10^{-17} \simeq 1$

Notice that without partial pivoting y_2 was $3\text{-}10^{17}$ or $\simeq \text{-}10^{17}$ with rounding. The right hand side value 3 in the unpivoted case was rounded away, where as now it is preserved. Continuing with the back substitution.

$$\begin{bmatrix} 1 & 2 \\ 0 & 1+2\cdot 10^{-17} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}$$

$x_2 \simeq 1 \quad x_1 \simeq 1$

LU Basics

Numerical Problems

Small Pivots

If the matrix is diagonally dominant or partial pivoting for round-off reduction is during LU Factorization:

1) The multipliers will always be smaller than one in magnitude.

2) The maximum magnitude entry in the LU factors will never be larger than $2^{(n-1)}$ times the maximum magnitude entry in the original matrix.

SMA-HPC ©2003 MIT

To see why pivoting helped notice that

$$\begin{bmatrix} 1 & 0 \\ 10^{-17} & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}$$

yields $\mathrm{y}_1 = 3,\ \mathrm{y}_2 = 1 - 10^{-17} \simeq 1$

Notice that without partial pivoting y_2 was $3\text{-}10^{17}$ or $\simeq \text{-}10^{17}$ with rounding. The right hand side value 3 in the unpivoted case was rounded away, where as now it is preserved. Continuing with the back substitution.

$$\begin{bmatrix} 1 & 2 \\ 0 & 1 + 2 \cdot 10^{-17} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}$$

$x_2 \simeq 1 \quad x_1 \simeq 1$

Hard to Solve Systems

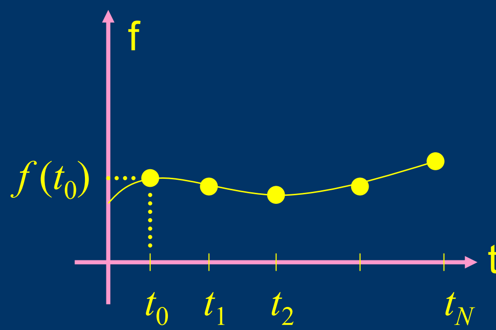
Fitting Example

Polynomial Interpolation

Table of Data

t_0	$f(t_0)$
t_1	$f(t_1)$
$\vdots$	$\vdots$
t_N	$f(t_N)$

$\Rightarrow$



Problem fit data with an N^{th} order polynomial

$$f(t) = \alpha_0 + \alpha_1 t + \alpha_2 t^2 + \ldots + \alpha_N t^N$$

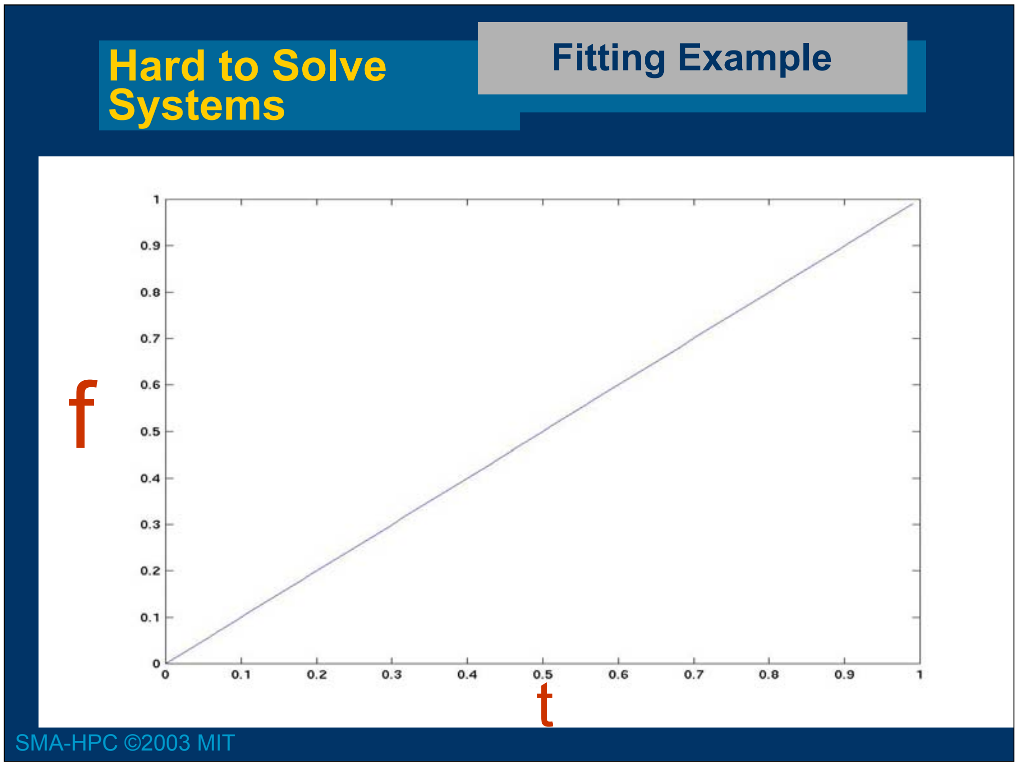
Hard to Solve Systems

Example Problem

Matrix Form

$$\underbrace{\begin{bmatrix} 1 & t_0 & t_0^{\,2} & & t_0^{\,N} \\ 1 & t_1 & t_1^{\,2} & & t_1^{\,N} \\ & & & & \\ & & & & \\ 1 & t_N & t_N^{\,2} & & t_N^{\,N} \end{bmatrix}}_{M_{\text{interp}}} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \\ \\ \alpha_N \end{bmatrix} = \begin{bmatrix} f(t_0) \\ f(t_1) \\ \\ \\ f(t_N) \end{bmatrix}$$

The kth row in the system of equations on the slide corresponds to insisting that the Nth order polynomial match the data exactly at point tk. Notice that we selected the order of the polynomial to match the number of data points so that a square system is generated. This would not generally be the best approach to fitting data, as we will see in the next slides.



Notice what happens when we try to fit a high order polynomial to a function that is nearly t. Instead of getting only one coefficient to be one and the rest zero, instead when 100th order polynomial is fit to the data, extremely large coefficients are generated for the higher order terms. This is due to the extreme sensitivity of the problem, as we shall see shortly.

Hard to Solve Systems

Perturbation Analysis

Measuring Error Norms

Vector Norms

L_2 (Euclidean) norm :

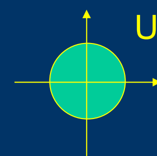
$$\|x\|_2 = \sqrt{\sum_{i=1}^{n} |x_i|^2}$$

L_1 norm :

$$\|x\|_1 = \sum_{i=1}^{n} |x_i|$$

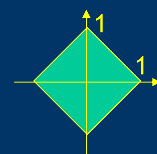
L_∞ norm :

$$\|x\|_\infty = \max_i |x_i|$$

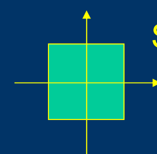


Unit circle

$\|x\|_2 < 1$



$\|x\|_1 < 1$



Square

$\|x\|_\infty < 1$

Hard to Solve Systems

Perturbation Analysis

Measuring Error Norms

Matrix Norms

Vector induced norm : $\|A\| = \max_x \dfrac{\|Ax\|}{\|x\|} = \max_{\|x\|=1} \|Ax\|$

Induced norm of A is the maximum "magnification" of x by A

Easy norms to compute:

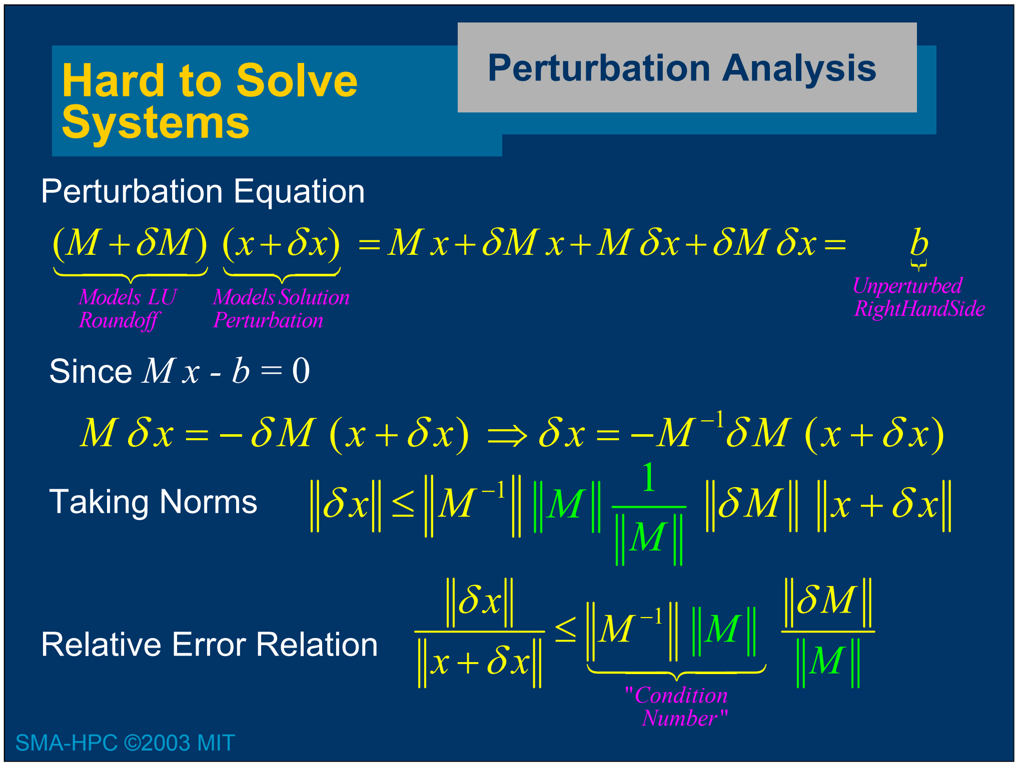
$\|A\|_1 = \max_j \sum_{i=1}^{n} |A_{ij}|$ = max abs column sum

Why? Let $x = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$

$\|A\|_\infty = \max_i \sum_{j=1}^{n} |A_{ij}|$ = max abs row sum

Why? Let $x = \begin{bmatrix} \pm 1 \\ \vdots \\ \vdots \\ \pm 1 \end{bmatrix}$

$\|A\|_2$ = not so easy to compute!!



As the algebra on the slide shows the relative changes in the solution x is bounded by an M-dependent factor times the relative changes in M. The factor

$$\| M^{-1} \| \, \| M \|$$

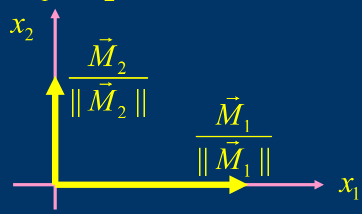
was historically referred to as the condition number of M, but that definition has been abandoned as then the condition number is norm-dependent. Instead the condition number of M is the ratio of singular values of M.

$$\operatorname{cond}(M) = \frac{\sigma_{\max}(M)}{\sigma_{\min}(M)}$$

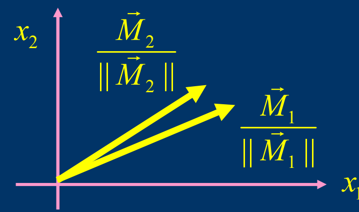
Singular values are outside the scope of this course, consider consulting Trefethen & Bau.

Geometric Approach is clearer

$M = [\vec{M}_1 \ \vec{M}_2]$, Solving $M x = b$ is finding $x_1 \vec{M}_1 + x_2 \vec{M}_2 = b$

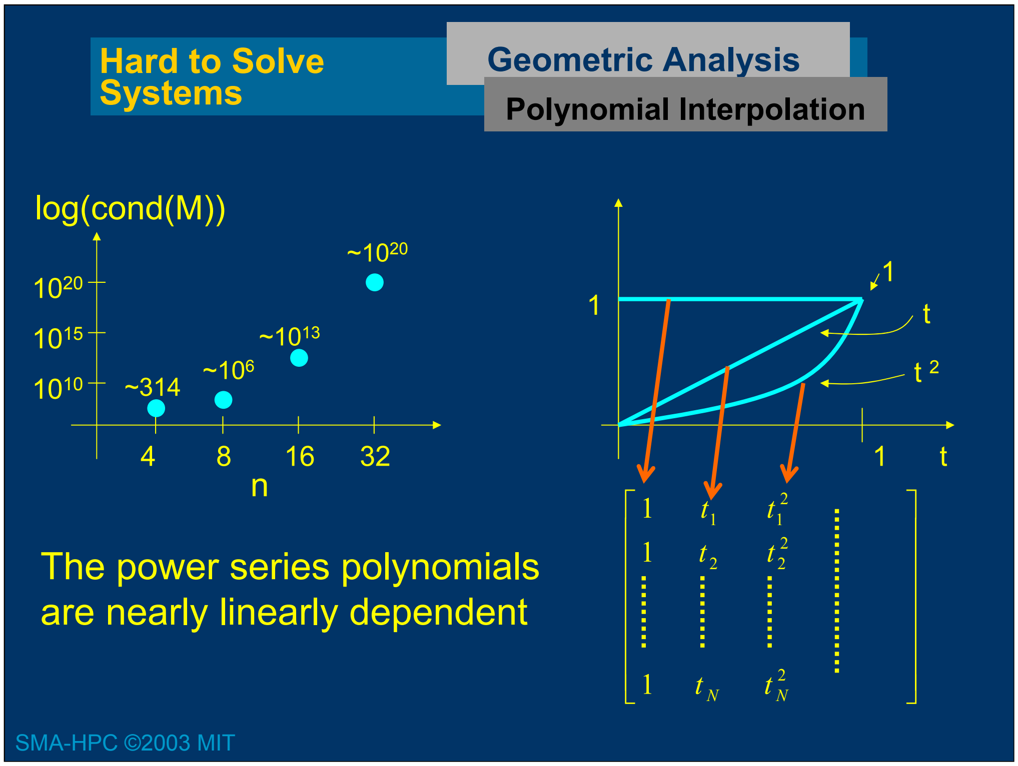


Columns orthogonal



Columns nearly aligned

When vectors are nearly aligned, difficult to determine how much of $\vec{M}_1$ versus how much of $\vec{M}_2$



Question Does row scaling reduce growth ?

$$\begin{bmatrix} d_{11} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & d_{NN} \end{bmatrix} \begin{bmatrix} a_{11} & \cdots & a_{1N} \\ \vdots & & \vdots \\ a_{N1} & \cdots & a_{NN} \end{bmatrix} = \begin{bmatrix} d_{11}a_{11} & \cdots & d_{11}a_{1N} \\ \vdots & & \vdots \\ d_{NN}a_{N1} & \cdots & d_{NN}a_{NN} \end{bmatrix}$$

Does row scaling reduce condition number ?

$\|M\| \, \|M^{-1}\| \equiv$ condition number of M

Theorem If floating point arithmetic is used, then row scaling ($D\,M\,x = D\,b$) will not reduce growth in a meaningful way.

If

$$\hat{M}_{LU}\,\hat{x} = b$$

and

$$\widehat{D\,M}_{LU}\,\hat{x}' = D\,b$$

then

$$\hat{x} = \hat{x}' \leftarrow \text{No roundoff reduction}$$

Summary

Solution Existence and Uniqueness

Gaussian Elimination Basics

- LU factorization
- Pivoting and Growth

Hard to solve problems

- Conditioning