1 Using the LU Matrix for Multiple Right-Hand Sides

- Many physical situations are modelled with a large set of linear equations.
- The equations will depend on the geometry and certain external factors that will determine the right-hand sides.
- If we want the solution for **many different values of these right-hand sides**,
- it is inefficient to solve the system from the start with each one of the right-hand-side values.
- Using the LU equivalent of the coefficient matrix is preferred.
- Suppose we have solved the system $Ax = b$ by Gaussian elimination.
- We now know the LU equivalent of A:
  $A = L * U$
- Consider now that we want to solve $Ax = b$ with some new b-vector.
- We can write

$$Ax = b$$

$$LUx = b$$

$$Ly = b$$

- The product of U and x is a vector, call it y.
- Now, we can solve for y from $Ly = b$.
- This is readily done because L is lower-triangular and we get y by forward-substitution.
- Call the solution $y = b'$.
- Going back to the original $LUx = b$, we see that, from $Ux = y = b'$, we can get x from $Ux = b'$.
- Which is again readily done by back-substitution (U is upper-triangular).

- i.e., Solve $Ax = b$, where we already have its L and U matrices:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0.66667 & 1 & 0 & 0 \\ 0.33333 & -0.45454 & 1 & 0 \\ 0.0 & -0.54545 & 0.32 & 1 \end{bmatrix} *$$

$$\begin{bmatrix} 6 & 1 & -6 & -5 \\ 0 & -3.6667 & 4 & 4.3333 \\ 0 & 0 & 6.8182 & 5.6364 \\ 0 & 0 & 0 & 1.5600 \end{bmatrix}$$

- Suppose that the b-vector is $[6\ -7\ -2\ 0]^T$.
- We first get $y = Ux$ from $Ly = b$ by forward substitution:

$$y = [6\ -11\ -9\ -3.12]^T$$

- and use it to compute x from $Ux = y$:

$$x = [-0.5\ 1\ 0.3333\ -2]^T.$$

- Now, if we want the solution with a different b-vector;

$$bb = [1\ 4\ -3\ 1]^T$$

- we just do $Ly = bb$ to get

$$y = [1\ 3.3333\ -1.8182\ 3.4]^T$$

- and then use this y in $Ux = y$ to find the new x:

$$x = [0.0128\ -0.5897\ -2.0684\ 2.1795]^T$$

2 The Inverse of a Matrix and Matrix Pathology

- <u>Division</u> by a matrix is not defined but the equivalent is obtained from the <u>*inverse*</u> of the matrix.
- If the product of two square matrices, $A * B$, equals to the *identity matrix*, I, B is said to be the inverse of A (and also A is the inverse of B).

- Matrices do not <u>commute</u> ($A*B \neq B*A$) on multiplication but inverses are an exception: $A*A^{-1} = A^{-1}*A$.
- To find the inverse of matrix A, <u>use an elimination method</u>.
- We augment the A matrix with the identity matrix of the same size and solve. The solution is A^{-1}. Example;

$$A = \begin{bmatrix} 1 & -1 & 2 \\ 3 & 0 & 1 \\ 1 & 0 & 2 \end{bmatrix},$$

$$\begin{bmatrix} 1 & -1 & 2 & 1 & 0 & 0 \\ 3 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 2 & 0 & 0 & 1 \end{bmatrix},$$

$$\begin{matrix} R_2 - (3/1)R_1 \rightarrow \\ R_3 - (1/1)R_1 \rightarrow \end{matrix}$$

$$\begin{bmatrix} 1 & -1 & 2 & 1 & 0 & 0 \\ 0 & 3 & -5 & -3 & 1 & 0 \\ 0 & 1 & 0 & -1 & 0 & 1 \end{bmatrix},$$

$$\begin{bmatrix} 1 & -1 & 2 & 1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 1 \\ 0 & 3 & -5 & -3 & 1 & 0 \end{bmatrix},$$

$$R_3 - (3/1)R_2 \rightarrow$$

- Cont.

$$\begin{bmatrix} 1 & -1 & 2 & 1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 1 \\ 0 & 0 & -5 & 0 & 1 & -3 \end{bmatrix},$$

$$R_3/(-5)$$,

$$R_1 - (2/1)R_3 \rightarrow$$,

$$\left[\begin{array}{cccccc} 1 & -1 & 0 & 1 & 2/5 & -6/5 \\ 0 & 1 & 0 & -1 & 0 & 1 \\ 0 & 0 & 1 & 0 & -1/5 & 3/5 \end{array}\right],$$

$$R_2 - (1/-1)R_1 \rightarrow ,$$

$$\left[\begin{array}{cccccc} 1 & 0 & 0 & 0 & 2/5 & -1/5 \\ 0 & 1 & 0 & -1 & 0 & 1 \\ 0 & 0 & 1 & 0 & -1/5 & 3/5 \end{array}\right]$$

- We confirm the fact that we have found the inverse by multiplication:

$$\underbrace{\left[\begin{array}{ccc} 1 & -1 & 2 \\ 3 & 0 & 1 \\ 1 & 0 & 2 \end{array}\right]}_{A} *$$

$$\underbrace{\left[\begin{array}{ccc} 0 & 2/5 & -1/5 \\ -1 & 0 & 1 \\ 0 & -1/5 & 3/5 \end{array}\right]}_{A^{-1}} =$$

$$\underbrace{\left[\begin{array}{ccc} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{array}\right]}_{I}$$

- It is more *efficient* to use Gaussian elimination. We show only the final triangular matrix; we used pivoting:

$$\left[\begin{array}{cccccc} 1 & -1 & 2 & 1 & 0 & 0 \\ 3 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 2 & 0 & 0 & 1 \end{array}\right] \rightarrow \left[\begin{array}{cccccc} 3 & 0 & 1 & 0 & 1 & 0 \\ (0.333) & -1 & 1.667 & 1 & -0.333 & 0 \\ (0.333) & (0) & 1.667 & 0 & -0.333 & 1 \end{array}\right]$$

- After doing the back-substitutions, we get

$$\left[\begin{array}{cccccc} 3 & 0 & 1 & 0 & 0.4 & -0.2 \\ (0.333) & -1 & 1.667 & -1 & 0 & 1 \\ (0.333) & (0) & 1.667 & 0 & -0.2 & 0.6 \end{array}\right]$$

- If we have the inverse of a matrix, we can *use it to solve a set of equations*, $Ax = b$,
- because multiplying by A^{-1} gives the answer (x):

$$A^{-1}Ax = A^{-1}b$$
$$x = A^{-1}b$$

2.1 Pathological Systems

- When a real physical situation is modelled by a set of linear equations, we can anticipate that the set of equations will have a solution that matches the values of the quantities in the physical problem (**the equations should truly do represent it**).
- Because of round-off errors, the solution vector that is calculated may imperfectly predict the physical quantity, but there is assurance that a solution exists.
- Here is an example of a matrix that has no inverse:

$$A = \begin{bmatrix} 1 & -2 & 3 \\ 2 & 4 & -1 \\ -1 & -14 & 11 \end{bmatrix}$$

Element A(3,3) cannot be used as a divisor in the back-substitution.

```
>> lu(A)
ans =
 2.0000      4.0000     -1.0000
-0.5000    -12.0000     10.5000
 0.5000      0.3333      0
```

That means that **we cannot solve**.

- The definition of a *singular matrix* is a matrix that *does not have an inverse.*

2.2 Redundant Systems

- Even though a matrix is singular, it may still have a solution. Consider again the same singular matrix:

$$A = \begin{bmatrix} 1 & -2 & 3 \\ 2 & 4 & -1 \\ -1 & -14 & 11 \end{bmatrix}$$

- Suppose we solve the system $Ax = b$ where the right-hand side is $b = [5, 7, 1]^T$.

```
>> Ab=[1 -2 3 5; 2 4 -1 7; -1 -14 11 1]
>> lu (Ab)
ans =
 2.0000      4.0000     -1.0000      7.0000
(-0.5000)  -12.0000     10.5000      4.5000
(0.5000)    (0.3333)     0           0
```

- The back-substitution cannot be done.
  - The output suggests that x_3 can have any value.
  - Suppose we set it equal to 0. We can solve the first two equations with that substitution, that gives $[17/4, -3/8, 0]^T$.
  - Suppose we set x_3 to 1 and repeat. This gives $[3, 1/2, 1]^T$, and this is another solution.
  - We have found a solution, actually, an infinity of them. The reason for this is that the system is **redundant**.
- What we have here is not truly three linear equations but only two independent ones.
- The system is called *redundant*.
- See Table 1 for the comparison of *singular* and *nonsingular* matrices.

3 Ill-Conditioned Systems

- A system whose coefficient matrix is singular has **no unique solution**.
- What if the matrix is *almost* singular?

$$A = \begin{bmatrix} 3.02 & -1.05 & 2.53 \\ 4.33 & 0.56 & -1.78 \\ -0.83 & -0.54 & 1.47 \end{bmatrix}$$

- The LU equivalent has a very small element in (3, 3),

$$LU = \begin{bmatrix} 4.33 & 0.56 & -1.78 \\ (0.6975) & -1.4406 & 3.7715 \\ (-0.1917) & (0.3003) & -0.0039 \end{bmatrix},$$

Table 1: A comparison of singular and nonsingular matrices

For Singular matrix A:	For Nonsingular Matrix A:
It has no inverse, A^{-1}	It has an inverse, A^{-1}
Its determinant is zero	The determinant is nonzero
There is no unique solution to the system $Ax = b$	There is a unique solution to the system $Ax = b$
Gaussian elimination cannot avoid a zero on the diagonal	Gaussian elimination does not encounter a zero on the diagonal
The rank is less than n	The rank equals n
Rows are linearly dependent	Rows are linearly independent
Columns are linearly dependent	Columns are linearly independent

- Inverse has elements very large in comparison to A:

$$inv(A) = \begin{bmatrix} 5.6611 & -7.2732 & -18.5503 \\ 200.5046 & -268.2570 & -66669.9143 \\ 76.8511 & -102.6500 & -255.8846 \end{bmatrix}$$

- Matrix is <u>nonsingular</u> but is <u>*almost*</u> singular.
- Suppose we solve the system $Ax = b$, with b equal to $[-1.61, 7.23, -3.38]^T$.
  - The solution is $x = [1.0000, 2.0000, -1.0000]^T$.
- Now suppose that we make a **small change** in just the first element of the b-vector : $[-1.60, 7.23, -3.38]^T$.
  - We get $x = [1.0566, 4.0051, -0.2315]^T$
- if $b = [-1.61, 7.22, -3.38]^T$, the solution now is $x = [1.07271, 4.6826, 0.0265]^T$ which also differs.
- A system whose coefficient matrix is <u>nearly singular</u> is called <u>**ill-conditioned**</u>.
- When a system is <u>ill-conditioned</u>, the solution is very <u>sensitive</u>
  - to small changes in the right-hand vector,
  - to small changes in the coefficients.
- $A(1, 1)$ is changed from 3.02 to 3.00, original b-vector, a large change in the solution $x = [1.1277, 6.5221, 0.7333]^T$].
- This means that it is also very sensitive to round-off error.

```
>> A=[3.02 -1.05 2.53; 4.33 0.56 -1.78; -0.83 -0.54 1.47];
b=[-1.61 7.23 -3.38];A\b'
ans =
    1.0000
    2.0000
   -1.0000
>> A=[3.02 -1.05 2.53; 4.33 0.56 -1.78; -0.83 -0.54 1.47];
b=[-1.60 7.23 -3.38];A\b'
ans =
    1.0566
    4.0050
   -0.2315
>> A=[3.02 -1.05 2.53; 4.33 0.56 -1.78; -0.83 -0.54 1.47];
b=[-1.61 7.22 -3.38];A\b'
ans =
    1.0727
    4.6826
    0.0265
```

```
>> A=[3.00 -1.05 2.53; 4.33 0.56 -1.78;
      -0.83 -0.54 1.47];
b=[-1.61 7.23 -3.38];A\b'
ans =
    1.1277
    6.5221
    0.7333
```

3.1 Norms

- **<u>Norm</u>**, a measure of the <u>magnitude of the matrix</u>.
- The magnitude of a single number is just its distance from zero: $|-4.2| = 4.2$.
- For vectors in two- or three space, norm is called the *Euclidean norm*, and is computed by $\sqrt{x_1^2 + x_2^2 + x_3^2}$.
- We compute the Euclidean norm of vectors with more than three components by

$$||x||_e = \sqrt{x_1^2 + x_2^2 + \ldots + x_n^2} = \left(\sum_{i=1}^{n} x_i^2\right)^{1/2}$$

- Defining the p-norm as

$$||x||_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{1/p}$$

- Using $||A||$ to represent the norm of matrix A, some properties

  1. $||A|| \geq 0$ and $||A|| = 0$ if and only if $A = 0$
  2. $||kA|| = ||k||||A||$
  3. $||A + B|| \leq ||A|| + ||B||$
  4. $||AB|| \leq ||A||||B||$

- 1-, 2-, and ∞-norms;

$$\begin{array}{l} ||x||_1 = \sum_{i=1}^{n} |x_i| = sum\ of\ magnitudes \\ ||x||_2 = \left(\sum_{i=1}^{n} |x_i|^2\right)^{1/2} = Euclidean\ norm \\ ||x||_\infty = max_{1\leq i\leq n}|x_i| = maximum - magnitude\ norm \end{array}$$

- i.e., Compute the 1-, 2-, and ∞-norms of the vector $x = (1.25, 0.02, -5.15, 0)$

$$\begin{array}{l} ||x||_1 = |1.25| + |0.02| + |-5.15| + |0| = 6.42 \\ ||x||_2 = 5.2996 \\ ||x||_\infty = 5.15 \end{array}$$

3.1.1 Matrix Norms

- The <u>norms</u> of a <u>matrix</u> are similar to the norms of a vector.

$$\begin{array}{l} ||A||_1 = max_{1\leq j\leq n} \sum_{i=1}^{n} |a_{ij}| = maximum\ column\ sum \\ ||A||_\infty = max_{1\leq i\leq n} \sum_{j=1}^{n} |a_{ij}| = maximum\ row\ sum \end{array}$$

- For an $m \times n$ matrix, the Frobenius norm is defined as

$$||A||_f = \left(\sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}^2\right)^{1/2}$$

  The Frobenius norm is a good measure of the magnitude of a matrix.

- Suppose r is the largest eigenvalue of $A^T * A$. Then $||A||_2 = r^{1/2}$.

- This is called the <u>*spectral norm*</u> of A, and $||A||_2$ is always less than (or equal to) $||A||_1$ and $||A||_\infty$.

- The spectral norm is usually the most *expensive*.

- Which norm is best? In most instances, we want the norm that puts the smallest upper bound on the magnitude of the matrix.

- In this sense, the spectral norm is usually the "best".

```
A =
  5 -5 -7
 -4  2 -4
 -7 -4  5
>> norm(A,'fro')
ans =
15
>> norm(A,inf)
ans =
17
>> norm(A,1)
ans=
16
>> norm(A)
ans =
12.0301
>> norm (A,2)
ans =
12.0301
```

we observe that the 2-norm, the spectral norm, is the norm we get if we just ask for the norm. The smallest norm of the matrix is the spectral norm, it is the tightest measure.

4 Iterative Methods

- Gaussian elimination and its variants are called ***direct methods***.

- An entirely different way to solve many systems is through ***iteration***.

- In this way, we start with an initial estimate of the solution vector and proceed to refine this estimate.

- The two methods for solving $Ax = b$ are

  1. the *Jacobi Method* ,

2. the *Gauss-Seidel Method*.

- An $n \times n$ matrix A is *diagonally dominant* if and only if;

$$|a_{ii}| > \sum_{j=1, j\neq i}^{n} |a_{ij}|, i = 1, 2, \ldots, n$$

- Although this may seem like a very restrictive condition, it turns out that there are very many applied problems that have this property.

- i.e.,

$$\begin{aligned} &6x_1 - 2x_2 + x_3 = 11 \\ &x_1 + 2x_2 - 5x_3 = -1 \\ &-2x_1 + 7x_2 + 2x_3 = 5 \end{aligned}$$

- The solution is $x_1 = 2, x_2 = 1, x_3 = 1$.

- However, before we begin our iterative scheme we must first *reorder the equations* so that the coefficient matrix is diagonally dominant.

4.1 Jacobi Method

- After reordering;

$$\begin{aligned} &6x_1 - 2x_2 + x_3 = 11 \\ &-2x_1 + 7x_2 + 2x_3 = 5 \\ &x_1 + 2x_2 - 5x_3 = -1 \end{aligned}$$

  Is the solution same? Check it out as an exercise.

- The iterative methods depend on the rearrangement of the equations in this manner:

$$x_i = \frac{b_i}{a_{ii}} - \sum_{j=1, j\neq i}^{n} \frac{a_{ij}}{a_{ii}} x_j, i = l, 2, \ldots, n, \mapsto x_1 = \frac{11}{6} - \left(\frac{-2}{6}x_2 + \frac{1}{6}x_3\right)$$

- Each equation now solved for the variables in succession:

$$\begin{aligned} x_1 &= 1.8333 + 0.3333x_2 - 0.1667x_3 \\ x_2 &= 0.7143 + 0.2857x_1 - 0.2857x_3 \\ x_3 &= 0.2000 + 0.2000x_1 + 0.4000x_2 \end{aligned} \tag{1}$$

- We begin with some initial approximation to the value of the variables.

- Say initial values are; $x_1 = 0, x_2 = 0, x_3 = 0$.

- Each component might be taken equal to *zero if no better initial estimates* are at hand.
- Note that this method is exactly the same as the method of *fixed-point iteration* for a single equation that was discussed in Section **??**.
- But it is now applied to a set of equations; we see this if we write Eqn. 1 in the form of
$$x^{(n+1)} = G(x^{(n)}) = b' - Bx^n$$
which is identical to $x_{n+1} = g(x_n)$ as used in Section **??**.
- The new values are substituted in the right-hand sides to generate a second approximation,
- and the process is repeated until successive values of each of the variables are sufficiently alike.
- Now, general form

$$\begin{aligned} x_1^{(n+1)} &= 1.8333 + 0.3333x_2^{(n)} - 0.1667x_3^{(n)} \\ x_2^{(n+1)} &= 0.7143 + 0.2857x_1^{(n)} - 0.2857x_3^{(n)} \\ x_3^{(n+1)} &= 0.2000 + 0.2000x_1^{(n)} + 0.4000x_2^{(n)} \end{aligned} \tag{2}$$

- Starting with an initial vector of $x^{(0)} = (0, 0, 0,)$, we obtain Table 2

	First	Second	Third	Fourth	Fifth	Sixth	...	Ninth
x_1	0	1.833	2.038	2.085	2.004	1.994	...	2.000
x_2	0	0.714	1.181	1.053	1.001	0.990	...	1.000
x_3	0	0.200	0.852	1.080	1.038	1.001	...	1.000

Table 2: Successive estimates of solution (Jacobi method)

- In the present context, $x^{(n)}$ and $x^{(n+1)}$ refer to the n^{th} and $(n+1)^{st}$ iterates of a vector rather than a simple variable, and g is a linear transformation rather than a nonlinear function.
- Rewrite in matrix notation; let $A = L + D + U$,

$$Ax = b, \begin{bmatrix} 6 & -2 & 1 \\ -2 & 7 & 2 \\ 1 & 2 & -5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 11 \\ 5 \\ -1 \end{bmatrix}$$

$$L = \begin{bmatrix} 0 & 0 & 0 \\ -2 & 0 & 0 \\ 1 & 2 & 0 \end{bmatrix}, D = \begin{bmatrix} 6 & 0 & 0 \\ 0 & 7 & 0 \\ 0 & 0 & -5 \end{bmatrix}, U = \begin{bmatrix} 0 & -2 & 1 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\begin{aligned} &Ax = (L + D + U)x = b \\ &Dx = -(L + U)x + b \\ &x = -D^{-1}(L + U)x + D^{-1}b \end{aligned}$$

- From this we have, identifying x on the left as the new iterate,

$$x^{(n+1)} = -D^{-1}(L + U)x^{(n)} + D^{-1}b$$

In Eqn. 2,

$$b' = D^{-1}b = \begin{bmatrix} 1.8333 \\ 0.7143 \\ 0.2000 \end{bmatrix}$$

$$D^{-1}(L + U) = \begin{bmatrix} 0 & -0.3333 & 0.1667 \\ -0.2857 & 0 & 0.2857 \\ -0.2000 & -0.4000 & 0 \end{bmatrix}$$

- This procedure is known as the Jacobi method, also called "*the method of simultaneous displacements*",
- because each of the equations is simultaneously changed by using the most recent set of x-values (see Table 2).

4.2 Gauss-Seidel Iteration

- Even though we have $newx^1$ available, we do not use it to compute $newx^2$.
- In nearly all cases the new values are better than the old and ought to be used instead.
- When this done, the procedure known as *Gauss-Seidel* iteration.
- We proceed to improve each x-value in turn, using always the most recent approximations of the other variables.
- These values were computed by using this iterative scheme:

$$\begin{aligned} x_1^{(n+1)} &= 1.8333 + 0.3333x_2^{(n)} - 0.1667x_3^{(n)} \\ x_2^{(n+1)} &= 0.7143 + 0.2857x_1^{(n+1)} - 0.2857x_3^{(n)} \\ x_3^{(n+1)} &= 0.2000 + 0.2000x_1^{(n+1)} + 0.4000x_2^{(n+1)} \end{aligned}$$

beginning with $x^{(1)} = (0, 0, 0)^T$

	First	Second	Third	Fourth	Fifth	Sixth
x_1	0	1.833	2.069	1.998	1.999	2.000
x_2	0	1.238	1.002	0.995	1.000	1.000
x_3	0	1.062	1.015	0.998	1.000	1.000

Table 3: Successive estimates of solution (Gauss-Seidel method)

- The rate of convergence is more rapid than for the Jacobi method (see Table 3).