LECTURE 9

Review of Matrix Algebra

We shall now look at algorithms for solving **systems of linear equations**. A system of linear equations is a set of equations of the form

$$
\left\{
\begin{array}{lcl}
a_{11}x_1 + a_{12}x_2 + \cdots + a_{1m}x_m & = & b_1 \\
a_{21}x_1 + a_{22}x_2 + \cdots + a_{2m}x_m & = & b_2 \\
\vdots & \vdots & \vdots \\
a_{n-1,1}x_1 + a_{n-1,2}x_2 + \cdots + a_{n-1,m}x_m & = & b_{n-1} \\
a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nm}x_m & = & b_n
\end{array}
\right. \tag{9.1}
$$

In the example above, we have a total of n equations and m unknowns. However, for the most part we shall be considering systems of equations where the total number of equations equals the total number of unknowns; for in this case, we generally have a unique solution. (If we have fewer equations than unknowns we don't have enough information to get a unique solution; and if we have more (independent) equations than unknowns that we won't have any solutions.)

The first step in casting the problem of solving systems of linear equations into the realm on numerical analysis, is to represent such a systems as a matrix equation. Thus, if we then denote by $\mathbf{A}$, $\mathbf{x}$, and $\mathbf{b}$ the matrices

$$
\mathbf{A} = \begin{bmatrix}
a_{11} & a_{12} & \cdots & a_{1,n-1} & a_{1n} \\
a_{21} & a_{22} & \cdots & a_{2,n-1} & a_{2n} \\
\vdots & \vdots & & \vdots & \vdots \\
a_{n-1,1} & a_{n-1,2} & \cdots & a_{n-1,n-1} & a_{n-1,n} \\
a_{n1} & a_{n,2} & \cdots & a_{n,n-1} & a_{nn}
\end{bmatrix}
$$

$$
\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} \quad , \quad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_{n-1} \\ b_n \end{bmatrix}
$$

then (9.1) (with $m = n$) can be written

$$\mathbf{Ax} = \mathbf{b}$$

Now what makes this formulation accessible to numerical analysis is the fact that most (if not all) numerical programming languages have a numerical array data type. In Maple, for example, a matrix like

$$
\mathbf{A} = \begin{pmatrix} 1 & 5 & 7 \\ 2 & 3 & 4 \\ 4 & 1 & 2 \end{pmatrix}
$$

can be identified with a array constructed as follows:

```
A := array(1..3,1..3);
A[1,1] := 1;
```

```
A[1,2] := 5;
A[1,3] := 7;
A[2,1] := 2;
A[2,2] := 3;
A[2,3] := 4;
A[3,1] := 4;
A[3,2] := 1;
A[3,3] := 2;
```

or more succinctly

```
A := array(1..3,1..3);
A := [[1,5,7],[2,3,4],[4.1,2]];
```

1. Three Simple Cases

1.1. Diagonal Matrices. A matrix equation $\mathbf{Ax} = \mathbf{b}$ is trivial to solve if the matrix $\mathbf{A}$ is purely diagonal. For if

$$\mathbf{A} = \begin{bmatrix} a_{11} & 0 & \cdots & 0 & 0 \\ 0 & a_{22} & \cdots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & a_{n-1,n-1} & 0 \\ 0 & 0 & \cdots & 0 & a_{nn} \end{bmatrix}$$

the corresponding system of equations reduces to

$$\begin{aligned} a_{11}x_1 &= b_1 \quad \Rightarrow \quad x_1 = \frac{b_1}{a_{11}} \\ a_{22}x_2 &= b_2 \quad \Rightarrow \quad x_2 = \frac{b_2}{a_{22}} \\ &\vdots \\ a_{nn}x_n &= b_n \quad \Rightarrow \quad x_n = \frac{b_n}{a_{nn}} \end{aligned}$$

1.2. Lower Triangular Matrices. If the matrix $\mathbf{A}$ has the form

$$\mathbf{A} = \begin{bmatrix} a_{11} & 0 & \cdots & 0 & 0 \\ a_{21} & a_{22} & \cdots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ a_{n-1,1} & a_{n-1,2} & \cdots & a_{n-1,n-1} & 0 \\ a_{n1} & a_{n,2} & \cdots & a_{n,n-1} & a_{nn} \end{bmatrix}$$

(with zeros everywhere above the diagonal from a_{11} to a_{nn}), then it is called **lower triangular**. A matrix equation $\mathbf{Ax} = \mathbf{b}$ in which $\mathbf{A}$ is lower triangular is also fairly easy to solve. For it is equivalent ot a system of equations of the form

$$\begin{aligned} a_{11}x_1 &= b_1 \\ a_{21}x_1 + a_{22}x_2 &= b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 &= b_3 \\ &\vdots \\ a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \cdots + a_{nn}x_n &= b_n \end{aligned}$$

To find the solution of such a system one solves the first equation for x_1 and then substitutes its solution b_1/a_{11} for the variable x_1 in the second equation

$$a_{21}\left(\frac{b_1}{a_{11}}\right) + a_{22}x_2 = b_2 \quad \Rightarrow \quad x_2 = \frac{1}{a_{22}}\left(b_2 - \frac{a_{21}b_1}{a_{11}}\right)$$

One can now substitute the numerical expressions for x_1, and x_2 into the third equation and get a numerical expression for x_3. Continuing in this manner we can solve the system completely.

At this point, it might be worthwhile to develop an explicit algorithm for carrying out the solution of a lower triangular linear system.

Let n denote the number of equations (and the number of unknowns). We shall assume the matrix $\mathbf{A}$ is lower triangular so that

$$a_{ij} = 0 \quad , \quad \text{if } j > i$$

According to the discusion above, we have

$$x_1 = \frac{b_1}{a_{11}}$$

and if we know $x_1, x_2, \ldots, x_{k-1}$ we can compute x_k from the k^{th} equation

$$\begin{aligned} x_k &= \frac{1}{a_{kk}}\left(b_k - a_{k1}x_1 - a_{k2}x_{k2} - \cdots - a_{k,k-1}x_k\right) \\ &= \frac{1}{a_{kk}}b_k - \frac{1}{a_{kk}}\sum_{i=1}^{k-1} a_{ki}x_i \end{aligned}$$

1.3. Upper Triangular Matrices. We can do a similar thing for systems of equations characterized by an upper triangular matrices $\mathbf{A}$. For if

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1,n-1} & a_{1n} \\ 0 & a_{22} & \cdots & a_{2,n-1} & a_{2n} \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & a_{n-1,n} \\ 0 & 0 & \cdots & 0 & a_{nn} \end{bmatrix}$$

then the corresponding system of equations will be

$$\begin{aligned} a_{11}x_1 + a_{12}x2 + \cdots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + \cdots + a_{2n}x_n &= b_2 \\ &\vdots \\ a_{n-1,n-1}x_{n-1} + a_{n-1,n}x_n &= b_{n-1} \\ a_{nn}x_n &= b_n \end{aligned}$$

which can be solved by substituting the solution of the last equation

$$x_n = \frac{b_n}{a_{nn}}$$

into the penultimate equation and solving for x_{n-1}

$$x_{n-1} = \frac{1}{a_{n-1,n-1}}\left(b_{n-1} - a_{n-1,n}\left(\frac{b_n}{a_{nn}}\right)\right)$$

and then substituting this result into the third from last equation, etc. The formula for a particular x_i then becomes

$$x_{n-i} = \frac{1}{a_{n-i,n-i}}\left(b_i - \sum_{k=0}^{i} a_{n-i,n-k}x_{n-k}\right)$$

EXAMPLE 9.1. Write a Maple program that solves the following system of equations

$$\begin{aligned} 3x_1 &= 6 \\ 2x_1 + x_2 &= 7 \\ x_1 + 2x_2 + x_3 &= 12 \\ x_1 + x_2 + 2x_3 + x_4 &= 18 \end{aligned}$$

- First note that this system corresponds to a lower triangular matrix

$$\mathbf{A} = \begin{pmatrix} 3 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 1 & 1 & 2 & 1 \end{pmatrix}$$

We can thus solve the equations $\mathbf{Ax} = \mathbf{b}$ by forward substitution: for each k from 1 to 4 we set

$$x_k = \frac{1}{a_{kk}} \left(b_k - \sum_{s=1}^{k-1} a_{ks} x_s \right)$$

In Maple this can be coded as follows:

```
n := 4;
A := array(1..n,1..n);
b := array(1..n);
x := array(1..n);
A := [[3,0,0,0],[2,1,0,0],[1,2,1,0],[1,1,2,1]];
b := [6,7,12,19];
for k from 1 to n do
    x[k] := (b[k] - sum(A[k,s]*x[s],s=1..k-1))/A[k,k];
    od;
```

□

EXAMPLE 9.2. Write a Maple program that solves the following system of equations

$$\begin{aligned} 3x_1 + 2x_2 + x_3 - x_4 &= 6 \\ 2x_2 + x_3 - x_4 &= 3 \\ -x_3 + x_4 &= 1 \\ 2x_4 &= 8 \end{aligned}$$

- First note that this system corresponds to a lower triangular matrix

$$\mathbf{A} = \begin{pmatrix} 3 & 2 & 1 & -1 \\ 0 & 2 & 1 & -1 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 2 \end{pmatrix}$$

We can thus solve the equations $\mathbf{Ax} = \mathbf{b}$ by backward substitution: for each k from 0 to 3 we set

$$x_{4-k} = \frac{1}{a_{4-k,4-k}} \left(b_{4-k} - \sum_{s=0}^{k-1} a_{4-k,4-s} x_{4-s} \right)$$

In Maple this can be coded as follows:

```
n := 4;
A := array(1..n,1..n);
b := array(1..n);
x := array(1..n);
```

```
A := [[3,2,1,-1],[0,2,1,-1],[0,0,-1,1],[0,0,0,2]];
b := [6,3,1,8];
for k from 0 to (n-1) do
    x[n-k] := (b[n-k] - sum(A[n-k,n-s]*x[n-s],s=0..k-1))/A[n-k,n-k];
    od;
```

□