Problem Set 2 Solutions

1) Show that

\[ L = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & 0 & 1 \end{bmatrix} \text{ is the inverse of } S = \begin{bmatrix} 1 & 0 & 0 \\ -l_{21} & 1 & 0 \\ -l_{31} & 0 & 1 \end{bmatrix}. \]

Multiply \( L \) and \( S \):

\[ LS = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ -l_{21} & 1 & 0 \\ -l_{31} & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} - l_{21} & 1 & 0 \\ l_{31} - l_{31} & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = I \]

Since \( LS = I \), \( L \) is the inverse of \( S \).

2)

(a) Find a 2x2 example of \( AB \neq BA \).

Let us consider the matrices

\[ A = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \]

then

\[ AB = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \neq \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} = BA. \]

(b) Find a 2x2 example of \( A^2 = -I \) with only real entries in \( A \).

If \( A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \), we must find the requirements on \( a, b, c \) and \( d \) such that \( A^2 = -I \) or \( A = A^{-1}(-I). \)

\[ A^{-1}(-I) = \frac{1}{ad-bc} \begin{bmatrix} -d & b \\ c & -a \end{bmatrix}, \]

therefore

\[ A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \frac{1}{ad-bc} \begin{bmatrix} -d & b \\ c & -a \end{bmatrix} = A^{-1}(-I). \]

This produces a set of equations:

\[ a = \frac{-d}{ad-bc}, \quad b = \frac{b}{ad-bc}, \quad c = \frac{c}{ad-bc}, \quad d = \frac{-a}{ad-bc}. \]

Through some algebra we find the constraints are \( a = -d \) and \( ad - bc = 1 \).

An example: \( A = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \).
(c) Find a 2x2 example of $B^2 = 0$ with no zeros in $B$.

$$B = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

$$B^2 = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} a^2 + bc & ab + bd \\ ac + ad & bc + d^2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

This produces the set of equations:

$$a^2 + bc = 0, \quad c(a + d) = 0, \quad b(a + d) = 0, \quad bc + d^2 = 0$$

Since none of the elements can equal zero, we are left with the constraints $a = -d$ and $bc = -a^2$.

An example: $B = \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}$.

3) Start with

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

Apply elimination to matrix $A$

$$\begin{bmatrix} 2 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 2 \end{bmatrix} \rightarrow \begin{bmatrix} 2 & 1 & 0 \\ 0 & 3/2 & 1 \\ 0 & 2 & 3 \end{bmatrix} \rightarrow \begin{bmatrix} 2 & 1 & 0 \\ 0 & 3/2 & 1 \\ 0 & 0 & 4/3 \end{bmatrix}$$

(i) (ii) (iii)

Matrix (ii) is achieved by subtracting $l_{21} = 1/2$ times row 1 from row 2.
Matrix (iii) is achieved by subtracting $l_{32} = 2/3$ times row 2 from row 3.

Hence we have

$$L = \begin{bmatrix} 1 & 0 & 0 \\ 1/2 & 1 & 0 \\ 0 & 2/3 & 1 \end{bmatrix}, \quad U = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 3/2 & 1 \\ 0 & 0 & 4/3 \end{bmatrix}$$

4) We first solve for vector $y$ such that $Ly = f$, and then solve $x$ such that $Ux = y$.

The equation $Ly = f$ is given by

$$\begin{bmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ 0 & 2 & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 3 \\ 6 \end{bmatrix}$$

Since $L$ is in lower triangular form, we use back substitution to conclude $y_1 = 0, y_2 = 3, y_3 = 0$.

Now, the equation $Ux = y$ is given by

$$\begin{bmatrix} 2 & 8 & 0 \\ 0 & 3 & 5 \\ 0 & 0 & 7 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 3 \\ 0 \end{bmatrix}$$

Since $U$ is in upper triangular form, we use back substitution to conclude $x_1 = -4, x_2 = 1, x_3 = 0$.

Thus,

$$x = \begin{bmatrix} -4 \\ 1 \\ 0 \end{bmatrix}.$$
5)

(a) The \( n \) equations relating to \( v_1, v_2, \ldots, v_n \):

Net current into any node is 0:

\[
\frac{v_{i-1} - v_i}{R} + \frac{v_{i+1} - v_i}{R} = 0
\]

for \( i = 2, 3, \ldots, n - 1 \)

Imposed boundary conditions:

\[
\begin{align*}
    v_1 &= 1 \\
    v_n &= 0
\end{align*}
\]

In the case of \( n=5 \)

The \( n \) equations in matrix form:

\[
\begin{bmatrix}
    1 & 0 & 0 & 0 & 0 \\
    -1 & 1 & 0 & 0 & 0 \\
    0 & 1 & -2 & 1 & 0 \\
    0 & 0 & 1 & -2 & 1 \\
    0 & 0 & 0 & 1 & 1 \\
\end{bmatrix}
\begin{bmatrix}
    v_1 \\
    v_2 \\
    v_3 \\
    v_4 \\
    v_5
\end{bmatrix}
= \begin{bmatrix}
    1 \\
    0 \\
    0 \\
    0 \\
    0
\end{bmatrix}
\]

We can solve this in MATLAB and see that the voltages at the nodes are evenly spaced from 0 to 1:

\[
v = \begin{bmatrix}
    1 \\
    0.75 \\
    0.5 \\
    0.25 \\
    0
\end{bmatrix}
\]
(b) MATLAB code:

```matlab
n = 10000;
A = sparse([],[],[],n,n,3*n-4);
b = zeros(n,1);
A(1,1) = 1;
A(n,n) = 1;
b(1,1) = 1;

for i = 2:n-1
    A(i,i-1) = 1;
    A(i,i) = -2;
    A(i, i+1) = 1;
end

% Determine how long it takes to solve Ax=b
tic
x = A \ b;
toc

% Print the computed value of node 5000
v5000 = sprintf("%0.6f", x(5000));
```

The computed value of $v_{5000}$ is 0.500050. Computation time is 0.002 seconds.
6)

(a) The $n^2$ equations relating to the nodes:

Net current into any node is 0:

$$\sum_{j \text{ a neighbor of } i} \frac{v_j - v_i}{R} = 0$$

for $i = 2, 3, \ldots n^2 - 1$

Imposed boundary conditions:

$$v_1 = 1$$
$$v_{n^2} = 0$$

In the case of $n=3$

The $n^2$ equations in matrix form:

$$\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & v_1 \\
1 & -3 & 1 & 0 & 1 & 0 & 0 & 0 & v_2 \\
0 & 1 & -2 & 0 & 0 & 1 & 0 & 0 & v_3 \\
1 & 0 & 0 & -3 & 1 & 0 & 1 & 0 & v_4 \\
0 & 1 & 0 & 1 & -4 & 1 & 0 & 1 & v_5 \\
0 & 0 & 1 & 0 & 1 & -3 & 0 & 0 & v_6 \\
0 & 0 & 0 & 1 & 0 & 0 & -2 & 1 & v_7 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & -3 & v_8 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & v_9 \\
\end{bmatrix} \begin{bmatrix}
v_1 \\
v_2 \\
v_3 \\
v_4 \\
v_5 \\
v_6 \\
v_7 \\
v_8 \\
v_9 \\
\end{bmatrix} = \begin{bmatrix}
1 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
\end{bmatrix}.$$
(b) MATLAB code:

```matlab
n=100; %the dimensions of the lattice

A=sparse([],[],[],n^2,n^2,5*n^2);
%Each row of A corresponds to a node in the lattice

ind = @(i,j) (i-1)*n+j;
%the function "ind" translates an (i, j) position in the lattice to a row
%and column index within the matrix A corresponding to the node at (i,j)

%for example:
%In a 3x3 lattice, the location (i, j)=(3, 2) in the lattice corresponds
%to node 8 which is represented by row and column 8 in matrix A.

%Iterate through the nodes:
for i=1:n
    for j=1:n
        c= ind(i,j);
        %We now fill in row 'c' of the matrix corresponding to node 'c'
        if(i==1 & & j==1) || (i==n & & j==n) %'c' is the first or last node
            A(c, c)=1;
            continue
        end
        if i>1 %node 'c' is not on the bottom row of the lattice
            A(c,c)=A(c,c)-1; %iterate the value of the diagonal by -1
            A(c, ind(i-1, j))= 1; %add 1 in the col of the node below 'c'
        end
        if i<n %node 'c' is not on the top row of the lattice
            A(c,c)=A(c,c)-1; %iterate the value of the diagonal by -1
            A(c, ind(i+1, j))= 1; %add 1 in the col of the node above 'c'
        end
        if j>1 %node 'c' is not in the leftmost column of the lattice
            A(c,c)=A(c,c)-1; %iterate the value of the diagonal by -1
            A(c, ind(i, j-1))= 1; %add 1 in the col of the node left of 'c'
        end
        if j<n %node 'c' is not in the rightmost column of the lattice
            A(c,c)=A(c,c)-1; %iterate the value of the diagonal by -1
            A(c, ind(i, j+1))= 1; %add 1 in the col of the node right of 'c'
        end
    end
end

b=zeros(n^2,1);
b(1)=1;
tic
y=A\b;
toc
v50=sprintf('%0.6f',y(50^2));
```
The computed value of $\nu_{50}$ is 0.488932.
Computation time is 0.07 seconds.

(c) The 2D problem is much more expensive than the 1D problem.