Orçun KÖK
50134

ELEC 418 Homework 2 Solutions

Problem 1.

a. To find conductance matrix, $where\ \frac{1}{R} \times V = I$;

➊ → $\frac{v_1 - 0}{R_1} + \frac{v_1 - v_2}{R_2} = 0$ $\qquad$ $v_1\left(\frac{1}{R_1} + \frac{1}{R_2}\right) + v_2\left(-\frac{1}{R_2}\right) = 0$

➋ → $\frac{v_2 - v_1}{R_2} + \frac{v_2 - v_3}{R_3} = 0$ $\qquad$ $v_1\left(-\frac{1}{R_2}\right) + v_2\left(\frac{1}{R_2} + \frac{1}{R_3}\right) + v_3\left(-\frac{1}{R_3}\right) = 0$

➌ → $\frac{v_3 - v_2}{R_3} + \frac{v_3 - v_2}{R_4} = 0$ $\qquad$ $v_2\left(-\frac{1}{R_3}\right) + v_3\left(\frac{1}{R_3} + \frac{1}{R_4}\right) + v_4\left(-\frac{1}{R_4}\right) = 0$

➍ → $\frac{v_4 - v_3}{R_4} + \frac{v_5 - v_4}{R_5} = 0$ $\qquad$ $v_3\left(-\frac{1}{R_4}\right) + v_4\left(\frac{1}{R_4} + \frac{1}{R_5}\right) + v_5\left(-\frac{1}{R_5}\right) = 0$

….. $\qquad$ ……

$For\ Nth \rightarrow \frac{v_{n-1} - v_{n-2}}{R_n} + \frac{v_{n-1} - 0}{R_5}$ $\qquad$ $v_n\left(\frac{1}{R_1} + \frac{1}{R_2}\right) + v_2\left(-\frac{1}{R_2}\right) = 0$

<u>NxN conductance (nodal analysis) matrix;</u>

$\frac{1}{R_1} + \frac{1}{R_2}$	$-\frac{1}{R_2}$				
$-\frac{1}{R_2}$	$\frac{1}{R_2} + \frac{1}{R_3}$	$-\frac{1}{R_3}$			
	$-\frac{1}{R_3}$	$\frac{1}{R_3} + \frac{1}{R_4}$	$-\frac{1}{R_4}$		
		$-\frac{1}{R_5}$	$\frac{1}{R_4} + \frac{1}{R_5}$	$-\frac{1}{R_5}$	
			….	…..	….
				$-\frac{1}{R_{n-1}}$	$\frac{1}{R_{n-1}} + \frac{1}{R_n}$

- There are NxN rows, hence there are N^2 entities. Disregarding the first and last rows, there are n-2 rows which contain 3 non-zero entities. Other (first and last) rows have got 2 entities. Calculating;

$$(n-2)x3 + 2x2 = \mathbf{3n - 2}$$

b. To find resistance matrix, I simply took inverse of the conductance matrix. To achieve that I created a 6x6 matrix at MatLab and to see the result. As expected, there is no zero element.

```
G =

     2    -1     0     0     0     0
    -1     2    -1     0     0     0
     0    -1     2    -1     0     0
     0     0    -1     2    -1     0
     0     0     0    -1     2    -1
     0     0     0     0    -1     2


ans =

    0.8571    0.7143    0.5714    0.4286    0.2857    0.1429
    0.7143    1.4286    1.1429    0.8571    0.5714    0.2857
    0.5714    1.1429    1.7143    1.2857    0.8571    0.4286
    0.4286    0.8571    1.2857    1.7143    1.1429    0.5714
    0.2857    0.5714    0.8571    1.1429    1.4286    0.7143
    0.1429    0.2857    0.4286    0.5714    0.7143    0.8571
```

c. To find L and U factors,

$$\left.\begin{array}{l} Mx=b \\ M=LU \end{array}\right\} \quad \begin{array}{l} Ly=b \\ Ux=y \end{array}$$

➔We know the M matrix which is the same conductance matrix in part a. rather than representing with values lets represent M matrix elements as G_{ij}.

G_{11}	G_{12}				
G_{21}	G_{22}	G_{23}			
	G_{32}	G_{33}	G_{34}		
		G_{43}	G_{44}	G_{45}	
			…	…	…
				$G_{n,n-1}$	$G_{n,n}$

Applying Gaussian Elimination for LU decomposition, ❶Finding u matrix➔

G_{11}	G_{12}				
$G_{21}-\frac{G_{21}}{G_{11}}G_{11} = 0$	$G_{22}-\frac{G_{21}}{G_{11}}G_{12} = G_{22}'$	$G_{23}-\frac{G_{21}}{G_{11}}G_{13} = G_{23}$			
	G_{32}	G_{33}	G_{34}		
		G_{43}	G_{44}	G_{45}	
			…	…	…
				$G_{n,n-1}$	$G_{n,n}$

❶Finding L matrix simultaneously➔

1					
$\frac{G_{21}}{G_{11}}$	1				
		1			
			1		
			…	…	
					1

❷Keeping on LU decomposition➔

G_{11}	G_{12}				
0	G_{22}'	G_{23}			
	$G_{32}-\frac{G_{32}}{G_{22}'}G_{22}' = 0$	$G_{33}-\frac{G_{32}}{G_{22}'}G_{32} = G_{33}'$	$G_{34}-\frac{G_{32}}{G_{22}'}G_{24} = G_{34}$		
		G_{43}	G_{44}	G_{45}	
			…	…	…
				$G_{n,n-1}$	$G_{n,n}$

❷Finding L matrix simultaneously➔

1					
$\frac{G_{21}}{G_{11}}$	1				
	$\frac{G_{32}}{G_{22}'}$	1			
			1		
			…	…	
					1

As it can be seen from the pattern keeping the decomposition on will generate;

U-matrix

G_{11}	G_{12}				
	G_{22}'	G_{23}			
		G_{33}'	G_{34}		
			G_{44}'	G_{45}	
				…	…
					$G_{n,n}'$

L-Matrix

1					
L_{21}	1				
	L_{32}	1			
		L_{43}	1		
			…	…	
				$Ln,_{n-1}$	1

Bot matrix have got NxN rows, hence there are N^2 entities for both matrix.

- *For the U matrix,* disregarding the last row, there are n-1 rows which contain 2 non-zero entities. Remaining (last) row has got 1 entity. Calculating;

$$(n-1)x2+1=\mathbf{2n-1}$$

- *For the L matrix,* disregarding the first row, there are n-1 rows which contain 2 non-zero entities. Remaining (first) row has got 1 entity. Calculating;

$$(n-1)x2+1=\mathbf{2n-1}$$

Comparing my result with MatLab,

```
L =

    1.0000         0         0         0         0         0
   -0.5000    1.0000         0         0         0         0
         0   -0.6667    1.0000         0         0         0
         0         0   -0.7500    1.0000         0         0
         0         0         0   -0.8000    1.0000         0
         0         0         0         0   -0.8333    1.0000


U =

    2.0000   -1.0000         0         0         0         0
         0    1.5000   -1.0000         0         0         0
         0         0    1.3333   -1.0000         0         0
         0         0         0    1.2500   -1.0000         0
         0         0         0         0    1.2000   -1.0000
         0         0         0         0         0    1.1667
```

MatLab script checks with my hand-made calculations.

Comparing non-zero entries of L, U and resistor matrix for N=1000,

1. **Resistor Matrix** ➔ $n^2=10^6$
2. **U Matrix** ➔ 2n-1=1999
3. **L Matrix** ➔ 2n-1=1999

U matrix and the L matrix has got the same number of entries while resistor matrix has every single entry filled in. From the computational side LU factorized matrices makes a lot of sense to use rather than the resistance matrix. Since it will take less space and need less computation

d. This is a tricky one. To evaluate that we should go as mathematically. Inverse of a matrix G is, $G^{-1} = \frac{1}{|G|} G^T$. But because of the property of our matrix G=G^T, which means that inverse of a matrix is $\frac{1}{|G|}$ scaled of the matrix itself. From calculations, which I used MatLab, one can easily see that $\frac{1}{|G|}$ for a NxN matrix is $\frac{1}{N+1}$.

```
ans =

    7.0000

>> det(G7)

ans =

    8.0000

>> det(G8)

ans =

    9.0000
```

As it is seen they minimum values of all the 6x6,7x7,8x8 matrices are salced according to their determinant such as;

0.1429*7/8=0.1250

0.1250*8/9=0.1111

```
>> min(min(inv(G6)))

ans =

    0.1429

>> min(min(inv(G7)))

ans =

    0.1250

>> min(min(inv(G8)))

ans =

    0.1111
```

Hence for the line of 1 Ohm resistances, minimum entry associated with the resistance matrix is $\frac{1}{N+1}$ where N is the size of resistance matrix.

Orçun KÖK
50134

Problem 2.

a. Let's assume we have Ax=b, then let y=Ux & Ly=b

1					-a
-a	1				
	-a	1			
		-a	1		
			-a	1	
				-a	1

1					-a
0	1				a^2
	0	1			-a^3
		-a	1		
			-a	1	
				-a	1

1					-a
0	1				a^2
	-a	1			
		-a	1		
			-a	1	
				-a	1

1					-a
0	1				a^2
	0	1			-a^3
		0	1		a^4
			0	1	-a^5
				0	1+a^6

As it can be seen that fill in occurs in the N^{th} column of the matrix, between A_{1N}, A_{NN}. From that deduction one can see that since A_{1N} and A_{NN} are already filled in. Hence, number of fill-ins that will occur will be ***N-2*** for any number N given.

b. Since we already have got the value a, we need to find the changes on the elements of the matrix to search a possible overflowing. I will handle this part separately as U matrix (Upper side) and L Matrix(lower side).

- Upper;

From the U matrix one can see that only changes are the fill-ins as well as the A. Given that a>1, for the factorized R(N-1) & R(N), where R(N) represents the N^{th} row,

(1) $R(n-1) = A_{n-1,n-1} + [(-1)^N a^N] * A_{n-1,n}$
(2) $R(n) = -a * A_{n-1,n-1} + 1 * A_{n,n}$

Calculating the new R(n) from equation 1 and 2,

(3) $R(n) \rightarrow R(n-1) \times a + R(n) = [1 + (-1)^N a^N] * A_{n,n}$

Since for the N^{th} column's last two elements one of them will be negative and the other will be positive. So looking at to$A_{n,n} = [1 + (-1)^N a^N]$ and $A_{n-1,n} = [(-1)^{N-1} a^{N-1}]$

Comparing $A_{n,n}$ & $A_{n-1,n}$ with their absolute values,

$$|1 + (-1)^N a^N| > |(-1)^{N-1} a^{N-1}|$$

Whether $A_{n,n}$ is positive or negative, it does not matter. $A_{n,n}$ is the biggest number that will be generated in the U matrix

- Lower;

For this part we will use the eq. 3, which the generalized formula fort he a matrix.

$R(n) \rightarrow R(n-1) \times a + R(n) = [1 + (-1)^N a^N] * A_{n,n}$ ➔Multiplier is "-a"

For the L matrix $L_{n-1,n}$=-a which is the only value input for the L matrix.

Now comparing the biggest value inputs as absolute values,

$$|1 + (-1)^N a^N| > |-a|$$

Where, $|1 + (-1)^N a^N|$ is the largest number input. Formulizing an equation for β;

$$\boldsymbol{\beta > \left|1 + (-1)^N a^N\right|}$$

c.

-a	1				
	-a	1			
		-a	1		
			-a	1	
				-a	1
1					-a

-a	1				
a^2	0	1			
		-a	1		
			-a	1	
				-a	1
1- a^2	a				-a

I couldn't find an algorithm to solve that problem. But for this part I worked around the statement that, *"Multipliers must always be smaller than one in magnitude"* to solve that problem. For this part I tried to reorder the matrix until the multiplier will be smaller than the one in magnitude but I couldn't implemented it.

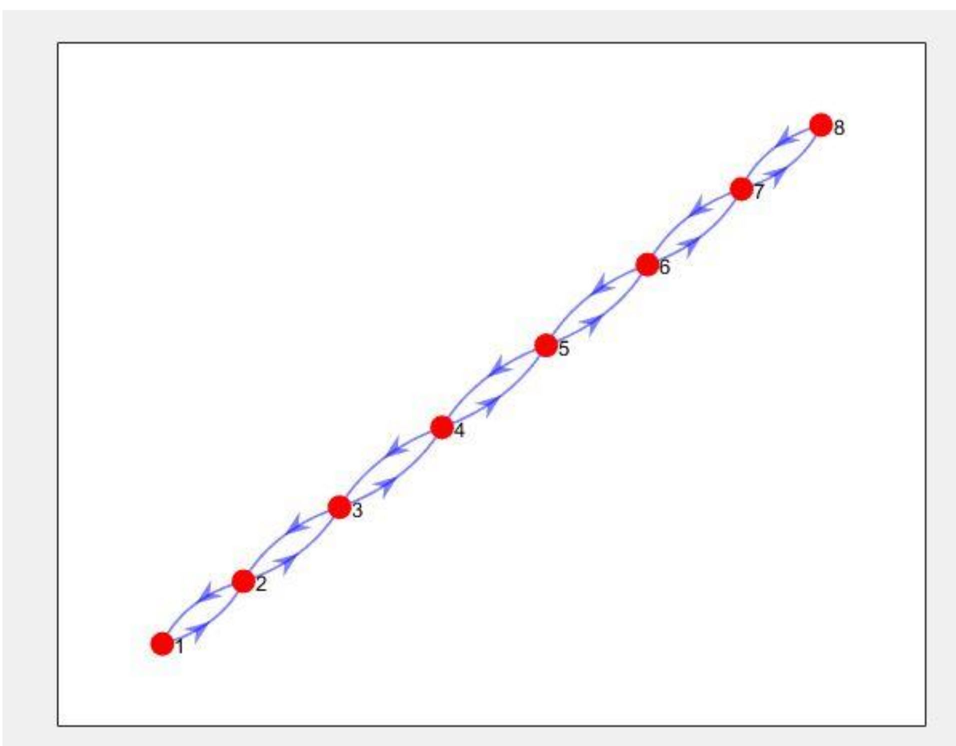
Problem 3.

Below is a MatLab created Markov Chain. MatLab code is provided. With the help of this picture we can examine if the statement *"For a Tridiagonal matrix, Gaussian elimination cannot produce more than 2n fill-ins"*. Even at a first glance you can see that. This is a 8x8 tridiagonal matrices Markov Chain but we can think it as if there are infinite number of nodes between somewhere in this chain while 8 becomes infinite. With that in mind we can say that we have N number of nodes. And as you can see that every node except the first and the last makes 2 connections, which are to their direct neighbors. This means the can produce 2 fill-ins at max. Also first and last nodes have only one connection. One can also see that that's why they can only produce 1 fill-ins.

So we have got N-2 rows that can produce 2 fill-ins at max and 2 rows that can produce 1 fill-ins at max. And also it should be kept in mind that it doesn't matter which node you are started from, the fill-ins that each node can produce will not change no matter what. Since this is a relatively an easy chain, it is more understandable. Going on with the calculation;

$$(n-2)x2+2=\mathbf{2n-2}$$

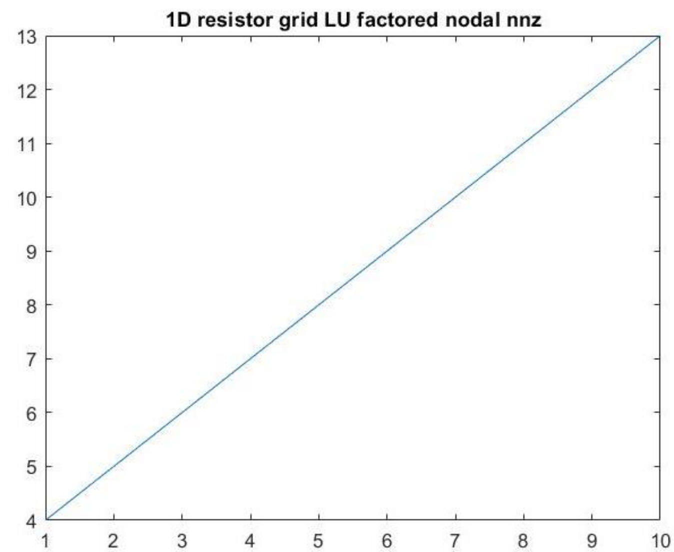
Hence maximum number of fill-ins that can occur is 2n-2.This proves the theorem.

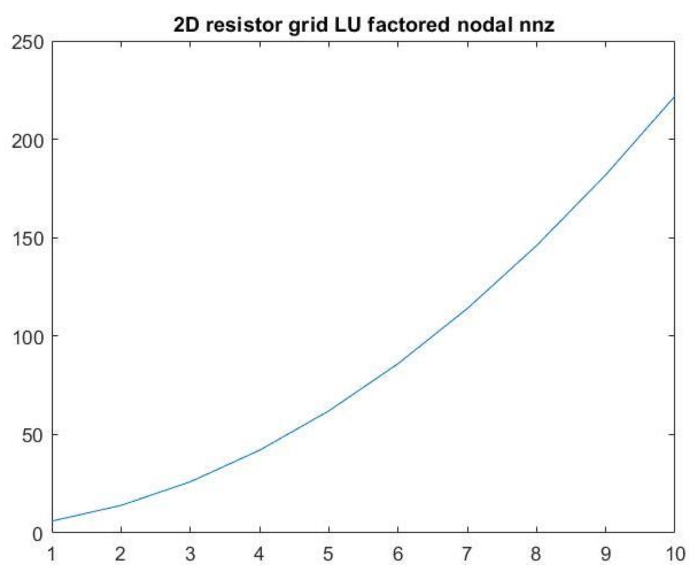


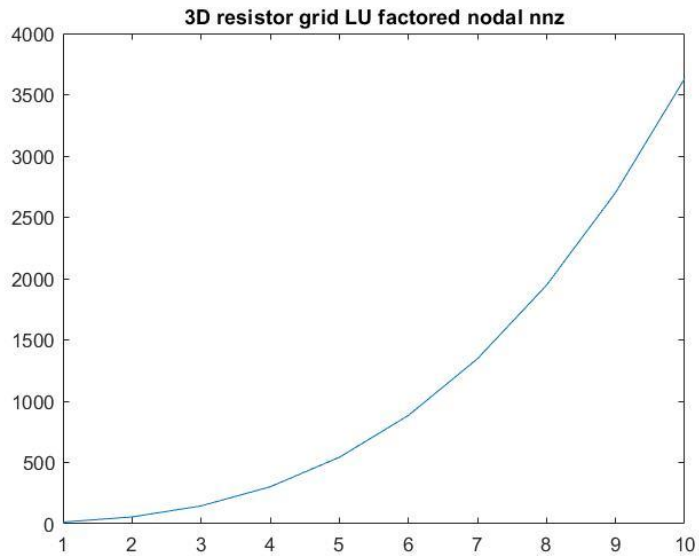
Problem 4.

a. Related code to this part is in the homework folder. My design idea fast to divide and conquer. First I created a 1D resistor array. And then I imagined two perpendicular sets of parallel 1D resistor array, which resulted in the creation of 2D resistor array. Just like the before I imagined a parallel set of 2D arrays. Afterwards, this was quite a unique since I didn't implement perpendicular sets of parallel 2D resistor arrays but Parallel 1D arrays. This is quite important or otherwise there was a possibility of creating more than one resistor for the same position.

b. I used m=10 for 1D, 2D, 3D grids. Since even m=10 requires too much memory and computational power.







It is clearly seen that with the increase in dimension, LU factored Nodal matrices non-zero elements increases more repeatedly. In other words it needs more computational power and memory.

c. For each dimension I took the largest value that my computers memory can handle.

For 1D➔m=5000 produces 5001x5001 matrix and 5003 nnz

For 2D➔m=60 produces 7320x7320 matrix and 7322 nnz

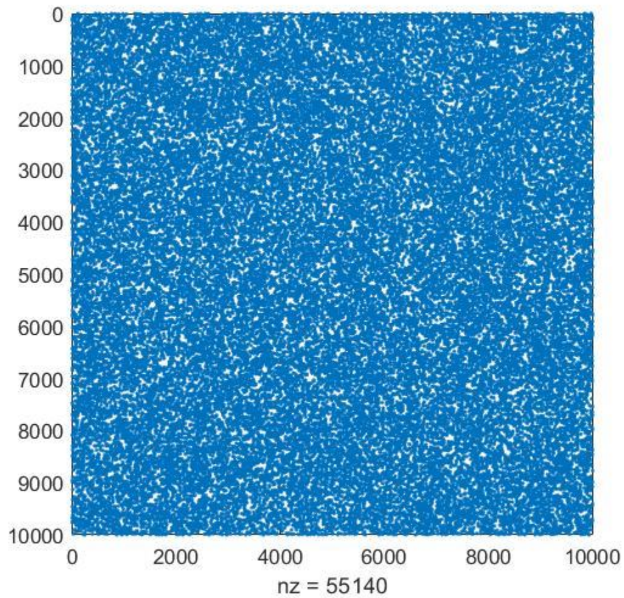
For 3D➔m=15 produces 11520x11520 matrix and 11522 nnz

Since the inverse matrix will have no zero elements size of the matrix will be the equivalent to number of nonzero elements. So comparing the inverse matrix nnz and the LU factored nnz;

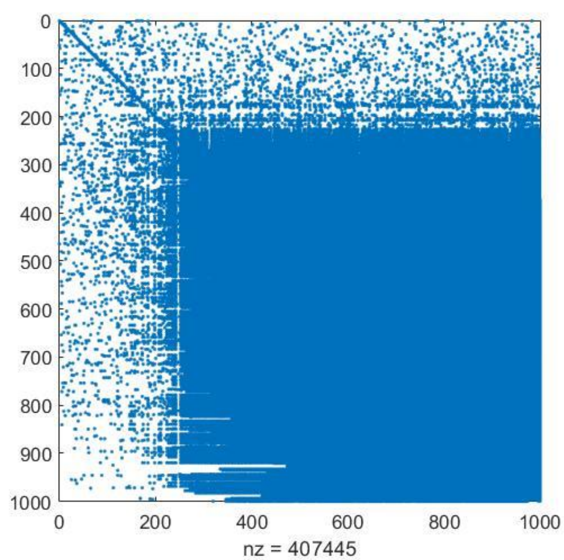
Ratio of $\frac{Lu\ factored\ nnz}{inverse\ Matrix\ nnz}$ is around $\frac{\boldsymbol{n+2}}{\boldsymbol{2n}}$

Problem 5.

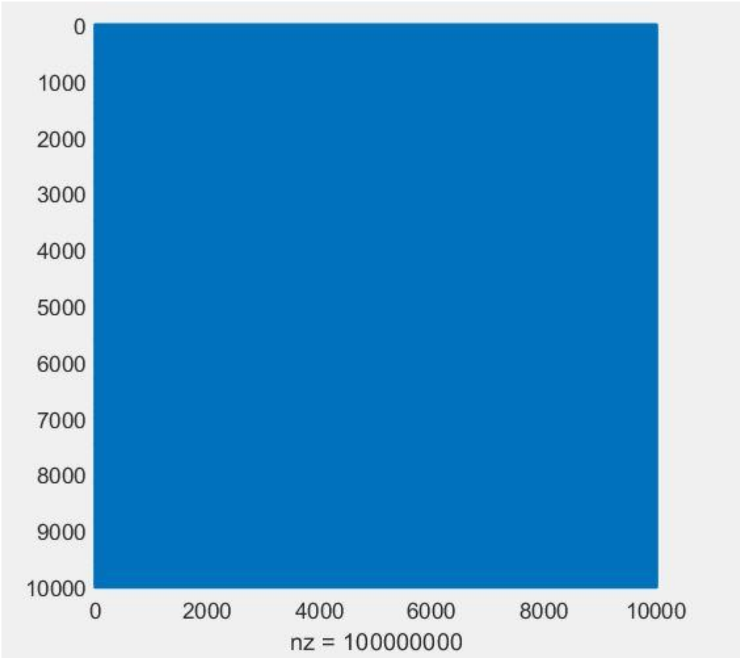
a. In the MatLab code, I compared the number of zeros with the number of non-zeros and zeros. Number of zeros are significantly higher. Since it is a sparse matrix. Below is the spy graph of the matrix.



Below is the factorization of Matrix, (because of the computational and memory limit of the computer, it is a 2000x2000 matrix)

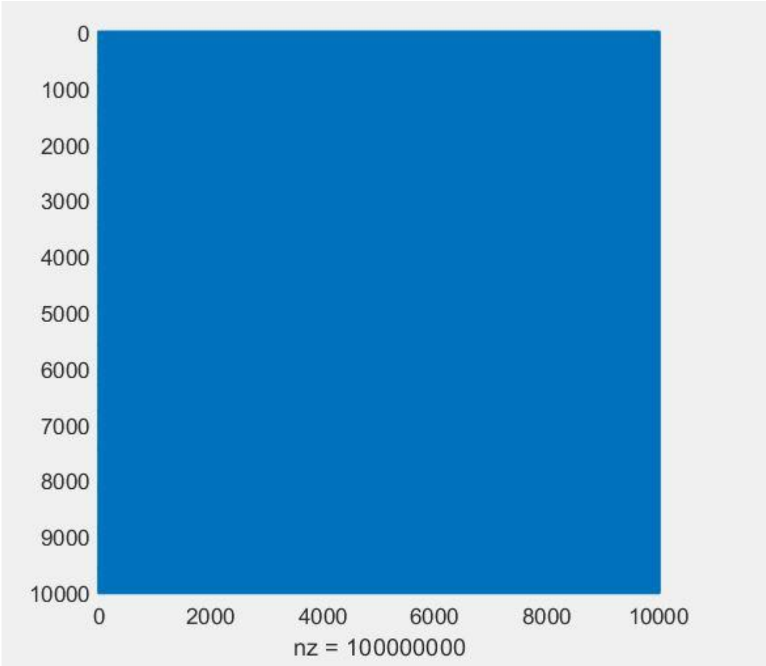


b. Below is the spy graph of as it can be seen it is not a sparse matrix. Even from the a_{ij} creating algorithm it can be seen that every entity of the matrix will be a non-zero.



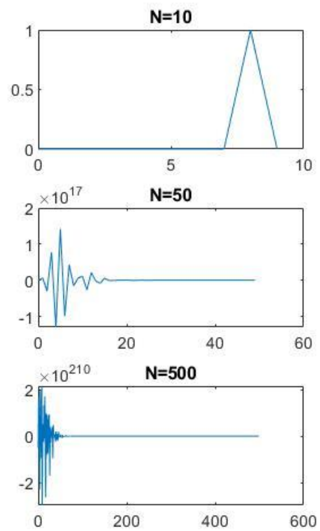
d. Computational complexity is $O(n^2)$. Power iteration applies matrix to an arbitrary starting vector and renormalizes.

e. With this method, my computation was slightly faster. Below is the spy graph of transition matrix.

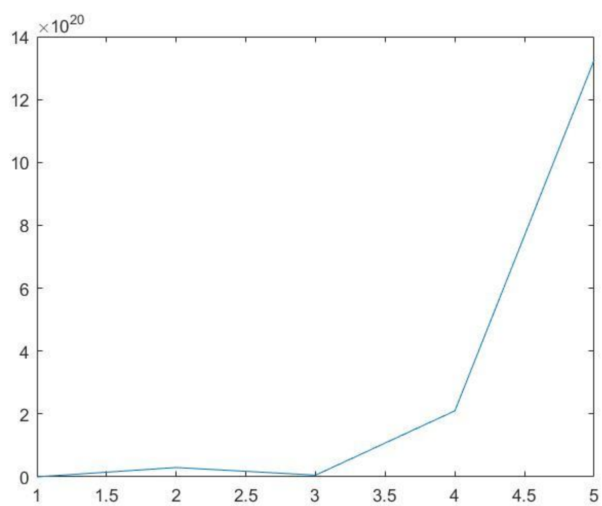


Problem 6.

a. Below graphs are the polynomial coefficients graphs. Computed polynomial coefficients does not agree with the exact solution. Because when the degree of the equation gets larger, computing of the computer gets decayed which corresponds to false coefficient results.



b. Graph of the condition numbers of matrices generated for the systems of equations plotted as a function of N,



c. What we can see is that the decaying of the coefficients has been increased. For instance without the disruption for N=100, my largest coefficient was around 10^{27} but now it is increased to around 10^{29}. Since we changed the function in such a way, we basically disrupted the equation which caused a decay on coefficients.