The University of Texas at Austin
Department of Electrical and Computer Engineering

EE362K: Introduction to Automatic Control—Fall 2009

Solutions for Problem Set Two

C. Caramanis — Due: Monday, September 21, 2009.

This problem set is intended to get us started thinking about linear algebra, and also to continue giving us some practice with Matlab.

1. The first column of A_1 is a multiple of the second column: $(1,-2)*2=(-2,4)$. Therefore, any vector of the form $(2x,-x)$ will be in the null space of A_1. Generically, the range of any matrix is the span of the columns. Since the columns of A_1 are linearly dependent, the range is just one-dimensional, and equal to $\text{Span}(1,-2)$.

   For A_2, you can check that any vector of the form: $(2x, x, -5x/2)$ is in the null space. Moreover, there are two linearly independent columns, which means that the pan of the columns is two dimensional, and that therefore the dimension of the range is 2. Since the dimension of the null space plus the dimension of the range equals n, or in this case 3, we know that indeed the null space is the one-dimensional subspace: $\text{Span}(2,1,-5/2)$. The range will be the entire (x,y)-plane, or, $\text{Span}\{(1,0,0),(0,1,0)\}$.

2. As mentioned in the book, the consensus algorithm is realizable by the update law:

$$x[k+1]=x[k]-\gamma(D-A)x[k]$$

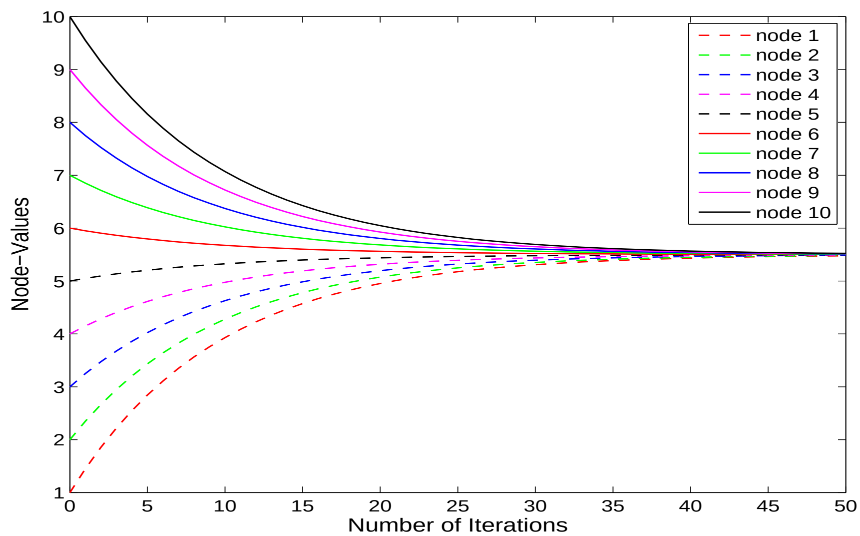
   where A is the adjacency matrix of the graph and D is a diagonal matrix with entries corresponding to the number of neighbors of each node. The constant γ describes the rate at which the estimate of the average is updated based on information from neighboring nodes.

   (a) A and D for K_{10} graph are given by:

$$A=\begin{bmatrix}0&1&1&1&1&1&1&1&1&1\\1&0&1&1&1&1&1&1&1&1\\1&1&0&1&1&1&1&1&1&1\\1&1&1&0&1&1&1&1&1&1\\1&1&1&1&0&1&1&1&1&1\\1&1&1&1&1&0&1&1&1&1\\1&1&1&1&1&1&0&1&1&1\\1&1&1&1&1&1&1&0&1&1\\1&1&1&1&1&1&1&1&0&1\\1&1&1&1&1&1&1&1&1&0\end{bmatrix},D=\begin{bmatrix}9&0&0&0&0&0&0&0&0&0\\0&9&0&0&0&0&0&0&0&0\\0&0&9&0&0&0&0&0&0&0\\0&0&0&9&0&0&0&0&0&0\\0&0&0&0&9&0&0&0&0&0\\0&0&0&0&0&9&0&0&0&0\\0&0&0&0&0&0&9&0&0&0\\0&0&0&0&0&0&0&9&0&0\\0&0&0&0&0&0&0&0&9&0\\0&0&0&0&0&0&0&0&0&9\end{bmatrix}$$

   The code and plot showing the convergence of consensus algorithm for K_{10} graph are:

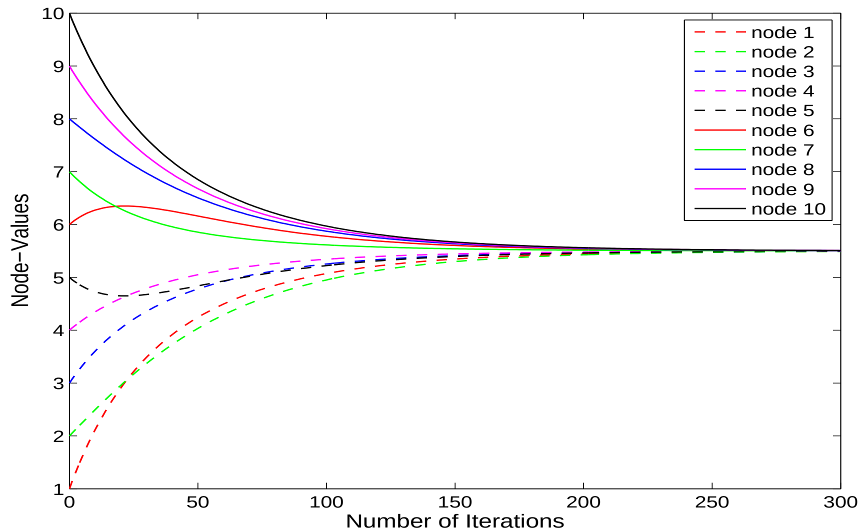
```
clear
clc

nodes=10;                          % number of nodes
iterations=500;                    % number of iterations
x=zeros(nodes,iterations);
x(:,1)=(1:nodes)';                 % initial values of the nodes

gamma=0.01;                        % value of gamma
D=diag(9*ones(1,nodes));           % diagonal matrix for K_10 graph
A=ones(nodes,nodes)-eye(nodes); % adjacency matrix for K_10 graph

for ctr=1:1:iterations-1
   x(:,ctr+1)=x(:,ctr)+gamma*(A-D)*x(:,ctr);
end

index=0:iterations-1;
plot(index,x(1,:),'r--',index,x(2,:),'g--',index,x(3,:),'b--',0:index,x(4,:),'m--',index,x(5,:),...
     'k--',index,x(6,:),'r',index,x(7,:),'g',index,x(8,:),'b',index,x(9,:),'m',index,x(10,:),'k');

legend('node 1','node 2','node 3','node 4','node 5','node 6','node 7','node 8','node 9','node 10');
xlabel('Number of Iterations')
ylabel('Node-Values')
```



(b) A and D for Petersen graph are given by:

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

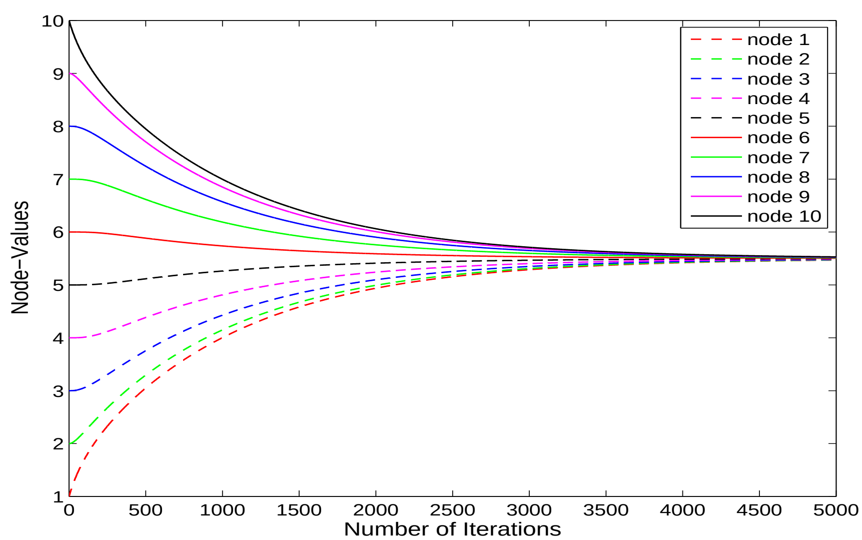
The plot for Petersen graph can be obtained by replacing values of A and D in the code for K_{10} graph in (a) and executing it:



(c) A and D for linear graph are given by:

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

The plot for linear graph can be obtained by replacing values of A and D in the code for K_{10} graph in (a) and executing it:



It can be observed that for the same value of γ, convergence rate of the consensus algorithm is highest for the K_{10} graph and lowest for the linear graph. The magnitude of γ also influences the convergence rates for each of graphs; convergence of the algorithm is possible only when γ is in some appropriate range for each graph.

3. For any v_1 and v_2 in V, $A(\alpha v_1 + \beta v_2) = \alpha A v_1 + \beta A v_2 = (\alpha + \beta) b$, for any $\alpha, \beta \in \mathbb{R}$. For V to be a vector space, we should have $(\alpha v_1 + \beta v_2) \in V$ or analogously, $A(\alpha v_1 + \beta v_2) = b$. Therefore, V is a vector space if and only if $b = (\alpha + \beta) b$ for all $\alpha, \beta \in \mathbb{R}$, or equivalently $b = 0$.

4. Since $Ax_1 = b$ and $Ax_2 = b$, $A(x_2 - x_1) = 0$. Let us take $z \triangleq x_2 - x_1 = [z_1\ z_2 \ldots z_n]^T$ and denote rows of A by A_i. Then $A(x_2 - x_1) = A_1 z_1 + A_2 z_2 + \ldots + A_n z_n = 0$. Since x_1 and x_2 are non-zero and $x_1 \neq x_2$, there is at least one $z_i \neq 0$, which means the rows of A are linearly dependent. Hence, A is not invertible.

5. One of the equivalent definitions of invertibility of a $n \times n$ matrix is that all the columns are independent. Suppose A is invertible, then it means that the columns of A, say $A_1, \ldots, A_n$, span all of $\mathbb{R}^n$. Now, since $b \in \mathbb{R}^n$, it means b is in the span of $\{A_1, \ldots, A_n\}$, which in turn means that we can write b as a linear combination of these column vectors, i.e., we can write:

$$b = A_1 x_1 + A_2 x_2 + \cdots + A_n x_n.$$

But this means $Ax = b$, which is a contradiction as there is no solution to $Ax = b$. Therefore, the columns of A cannot be independent, and therefore A cannot be invertible.

6.
- The eigenvalues of A

  In order to calculate the eigenvalues of A in MATLAB, we can use `eig()` function.

```
>> A=[-2 6 1;6 12 -6;1 -6 4]

A =

    -2     6     1
     6    12    -6
     1    -6     4

>> E=eig(A)

E =

   -5.3673
    2.8410
   16.5264
```

- The product of the eigenvalues of A

  Now that we have stored the eigenvalues of A in matrix E, we can use the `prod()` function to calculate the product of the elements of E.

```
>> prod(E)

ans =

  -252.0000
```

- The determinant of A

  We use the `det()` function to determine the determinant of A.

```
>> det(A)

ans =

  -252
```

and we see that it coincides the product of the eigenvalues.

- The determinant of $A^\top$
  In order to represent the transpose of A in MATLAB we use the single quote:

```
>> det(A')

ans =

  -252
```

and we see that it coincides `det(A)`.

- Consider the matrix

$$T = \begin{bmatrix} 5 & 5 & 4 \\ 9 & 0 & 6 \\ 8 & 8 & 8 \end{bmatrix}$$

Check that T is invertible
In order to check if T is invertible, we calculate its determinant.

```
>> det(T)

ans =

   -72
```

which is not zero. Therefore, T is invertible.

- Compute the determinant of $T^{-1}AT$

```
>> det(T^-1*A*T)

ans =

 -252.0000
```

- Try this again for some other invertible matrix, T_2. Any conjectures?
  Let us take T_2 as:

$$T_2 = \begin{bmatrix} 4 & -3 & 1 \\ -7 & 0 & 13 \\ 2 & 5 & 8 \end{bmatrix}$$

```
>> det(T2^-1*A*T2)

ans =

 -252.0000
```

As shown in the class, for any invertible matrix T we have $|T^{-1}AT| = |A|$.