EE 5329 Homeworks
Spring 2018

Updated: Thursday, February 08, 2018

DO NOT DO HOMEWORK UNTIL IT IS ASSIGNED.
THE ASSIGNMENTS MAY CHANGE UNTIL ANNOUNCED.

- For full credit, show all work.
- Some problems require hand calculations. In those cases, do not use MATLAB except to check your answers.

It is OK to talk about the homework beforehand.

BUT, once you start writing the answers, MAKE SURE YOU WORK ALONE.

The purpose of the Homework is to evaluate you individually, not to evaluate a team.

Cheating on the homework will be severely punished.

The next page must be signed and turned in at the front of ALL homeworks submitted in this course.

Homework Pledge of Honor

On all homeworks in this class - YOU MUST WORK ALONE.

Any cheating or collusion will be severely punished.

It is very easy to compare your software code and determine if you worked together It does not matter if you change the variable names.

Please sign this form and include it as the first page of all of your submitted homeworks.

....……………………………………………………………........

Typed Name: ____

Pledge of honor:

"On my honor I have neither given nor received aid on this homework."

Signature: ____

EE 5329 Homework 1

State Variable Systems, Computer Simulation

1. Simulate the van der Pol oscillator $y''+\alpha(y^2-1)y'+y=0$ using MATLAB for various ICs. In each case below, first use ICs of y(0)=0, y'(0)= 0.5. Then use ICs of y(0)=4, y'(0)= 4. Plot y(t) vs. time *t* and also the phase plane plot y'(t) vs. y(t).
   a. For α= 0.05.
   b. For α= 0.9.

2. Do MATLAB simulation of the Lorenz Attractor chaotic system. Run for 150 sec. with all initial states equal to 0.5. Plot states versus time, and also make 3-D plot of x_1, x_2, x_3 using PLOT3(x1,x2,x3).

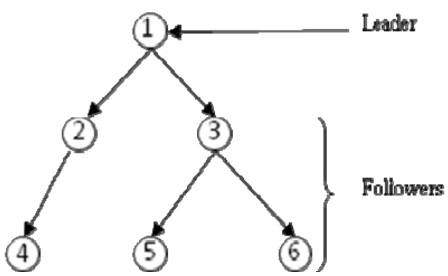
$$\dot{x}_1 = -\sigma(x_1 - x_2)$$
$$\dot{x}_2 = rx_1 - x_2 - x_1x_3$$
$$\dot{x}_3 = -bx_3 + x_1x_2$$

use σ= 10, r= 28, b= 8/3.

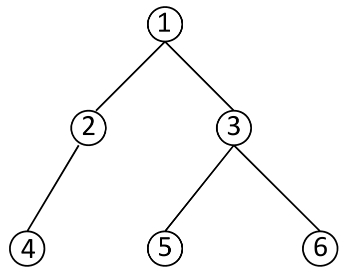
EE 5329 Homework 2

Graph Laplacian Eigenvalues

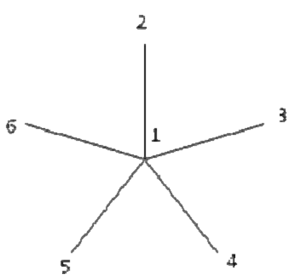
For the following graphs, take all edge weights equal to 1/2. (a) Write the adjacency matrix A and the graph Laplacian L. (b) Find the eigenvalues of L and plot in the complex s-plane. (c) Compare the Fiedler e-vals λ_2. (d) Find the left eigenvector w_1 of L for $\lambda_1 = 0$. (e) Find the Fiedler e-vector v_2.



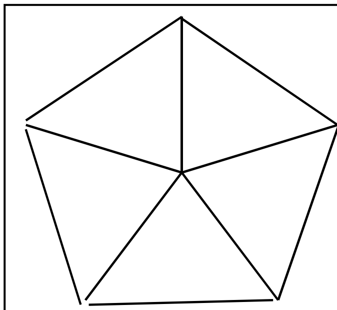
1. Directed Tree graph. (Formation graph)



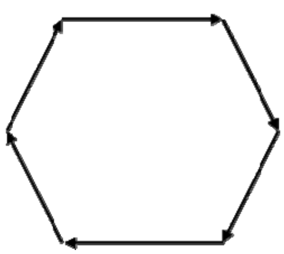
2. Undirected Tree graph.



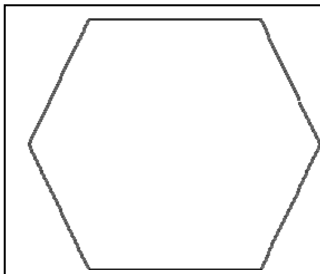
3. Star $K_{1,5}$



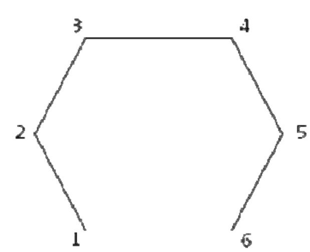
4. Star with extra edges



5. Directed 6-cycle (6-periodic graph)



6. 6-cycle C^6 (2-regular graph).



7. Path P^5.

EE 5329 Homework 3

Consensus and Graph Eigenvalues

1. Continuous-Time Consensus

Simulate the continuous-time consensus protocol

$$\dot{x}_i = u_i = \sum_{j \in N_i} a_{ij}(x_j - x_i)\,.$$

for all the graphs on Homework 2. Take all edge weights equal to 1. For each case, plot all the states versus time.

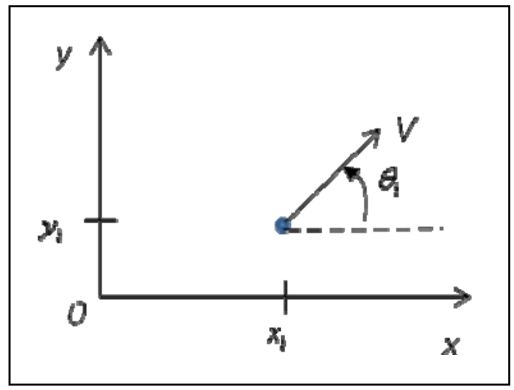
2. Consensus for Formation Control

Let each node have the vehicle dynamics given by

$$\dot{x}_i = V \sin\theta_i$$

$$\dot{y}_i = V \cos\theta_i$$

with $(x_i(t), y_i(t))$ the position and $\theta_i(t)$ the heading. This corresponds to motion in the *(x,y)* plane with velocity *V* as shown. All nodes have the same velocity *V*.

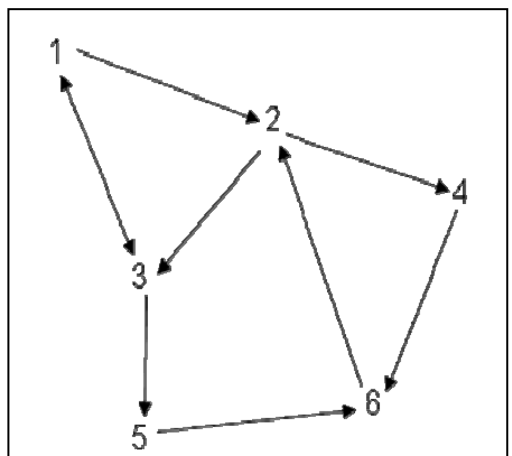


Take a flock of 6 nodes with the strongly connected communication graph structure shown. Run the continuous-time local voting protocol on the headings of the nodes as

$$\dot{\theta}_i = \sum_{j \in N_i} a_{ij}(\theta_j - \theta_i)$$

Take all the edge weights equal to 1. Set the initial headings random between $0 - 2\pi$ rads.

Now, you have 3 states running at each node: $\theta_i(t)$, $(x_i(t), y_i(t))$. Plot the six headings vs time. They should reach the weighted average consensus. Plot the six positions $(x_i(t), y_i(t))$ of the nodes in the plane vs time. They should eventually all go off in the same direction.



Graph for formation control.

EE 5329 Homework 4

Discrete-time Consensus

1. **Discrete-Time Consensus**
   Simulate the DT consensus protocol using the normalized form protocol

$$x_i(k+1) = x_i(k) + \frac{1}{d_i+1}\sum_{j\in N_i} a_{ij}(x_j(k) - x_i(k)) = \frac{1}{d_i+1}\left(x_i(k) + \sum_{j\in N_i} a_{ij}x_j(k) \right)$$

on graphs 3 and 7 from Homework 2. Take all edge weights equal to 1//2. For each case, plot all the states versus time. Start with random initial condition states between -1, +1.