



Cinvestav

Centro de Investigación y de Estudios
Avanzados del Instituto Politécnico Nacional
Unidad Guadalajara

Tarea 4. Seguimiento de consenso y topologías cambiantes

Presentado por

Jesús Alejandro Díaz Hernández

Presentado para el curso de
Tópicos avanzados de control 2

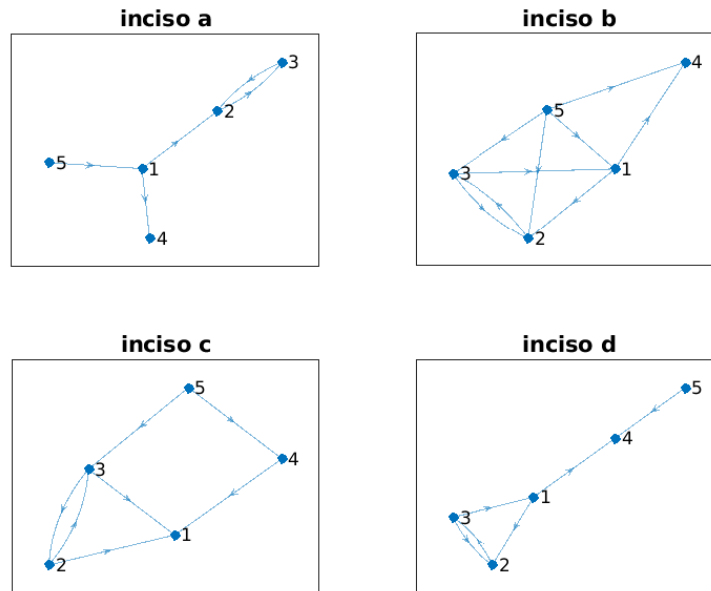
Curso impartido por: Héctor Manuel Becerra Fermín
Profesor

Guadalajara, Jalisco

5 de junio del 2024

Pregunta 1.-

Para referencia los grafos son los que se muestran a continuación



Inciso a)

Las matrices Laplacianas son: inciso a)

$$\begin{bmatrix} 1 & 0 & 0 & 0 & -1 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

inciso b)

$$\begin{bmatrix} 2 & 0 & -1 & 0 & -1 \\ -1 & 3 & -1 & 0 & -1 \\ 0 & -1 & 2 & 0 & -1 \\ -1 & 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

inciso c)

$$\begin{bmatrix} 3 & -1 & -1 & -1 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & -1 & 2 & 0 & -1 \\ 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

inciso d)

$$\begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ -1 & 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

b)

Los eigenvalores de la matriz laplaciana negativa son: inciso a)

$$\begin{bmatrix} -0.3820 \\ -2.6180 \\ -1 \\ -1 \\ 0 \end{bmatrix}$$

inciso b)

$$\begin{bmatrix} -2 \\ -1 \\ -3 \\ -3 \\ 0 \end{bmatrix}$$

inciso c)

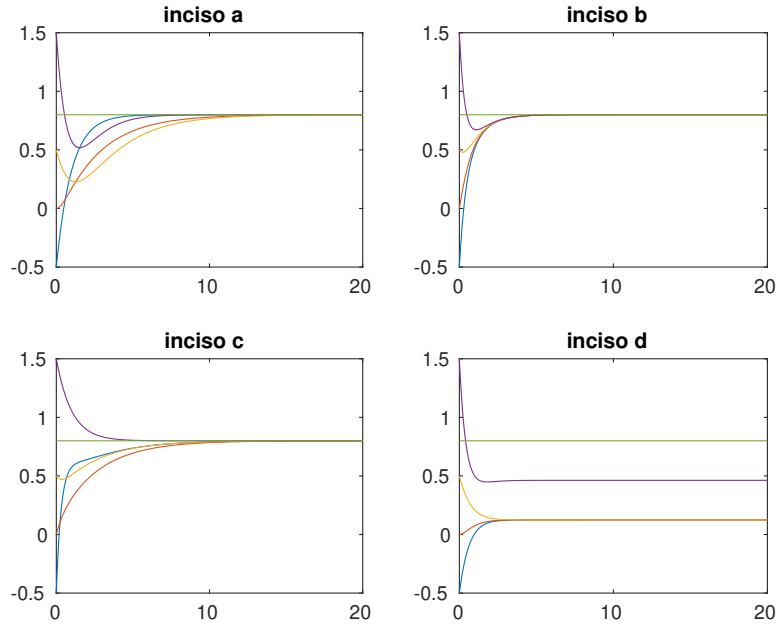
$$\begin{bmatrix} -3 \\ -0.3820 \\ -2.6180 \\ -1 \\ 0 \end{bmatrix}$$

inciso d)

$$\begin{bmatrix} -2 \\ 0 \\ -2 \\ -2 \\ 0 \end{bmatrix}$$

c)

Reproduciendo el control del algoritmo 4 podemos reproducir los resultados de la figura 2 del artículo, pero para una referencia de 0.8.

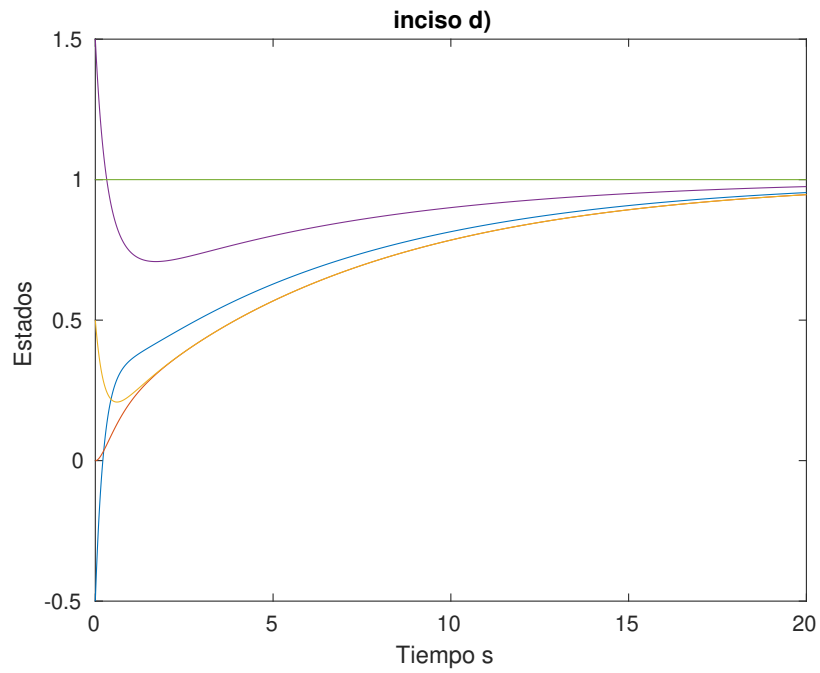


Como se puede ver a mayor conectividad algebraica, mayor rápido es la convergencia al punto de referencia. Además, la conectividad algebraica de cero en el inciso d) habla sobre la presencia del nodo aislado, por lo que no habrá consenso.

d)

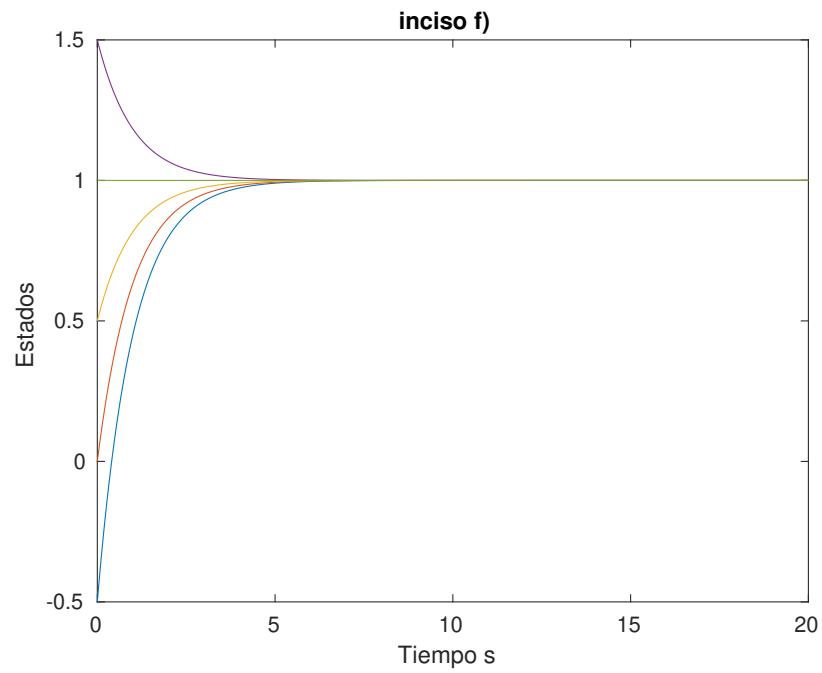
Como ahora el nodo 4 no está aislado, y existe un spanning tree, entonces se esperaría consenso. Esto es mostrado en el inciso siguiente pero para una referencia de 1.

e)



f)

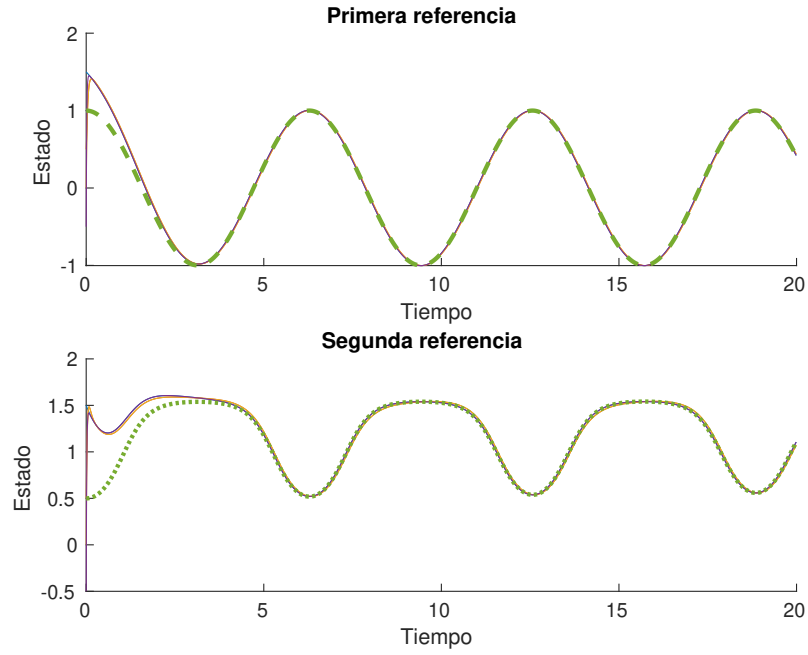
Considerando el grafo b), pero quitando la comunicación entre agentes, es decir, solo el nodo de referencia les envía información, se obtiene el siguiente resultado



Pregunta 2.-

punto 1.-

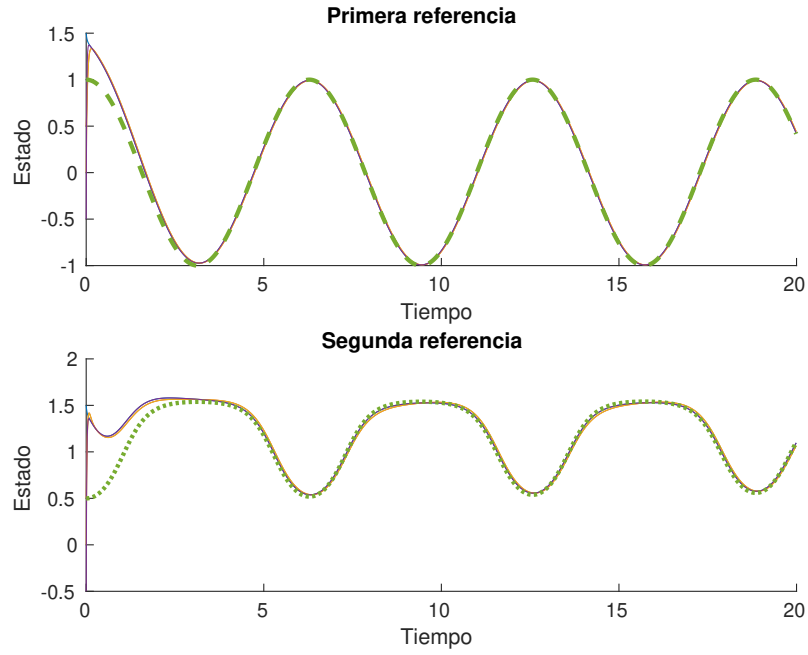
Se puede reproducir los grafos dirigidos que corresponden a la figura 6, es el mismo que el grafo de la pregunta anterior inciso a), y utilizando el algoritmo 8, se puede llegar al siguiente consenso



Sin embargo, el artículo no establece valores para las ganancias así que use matrices de unos tanto para k como para α , y matrices de 30 para las γ

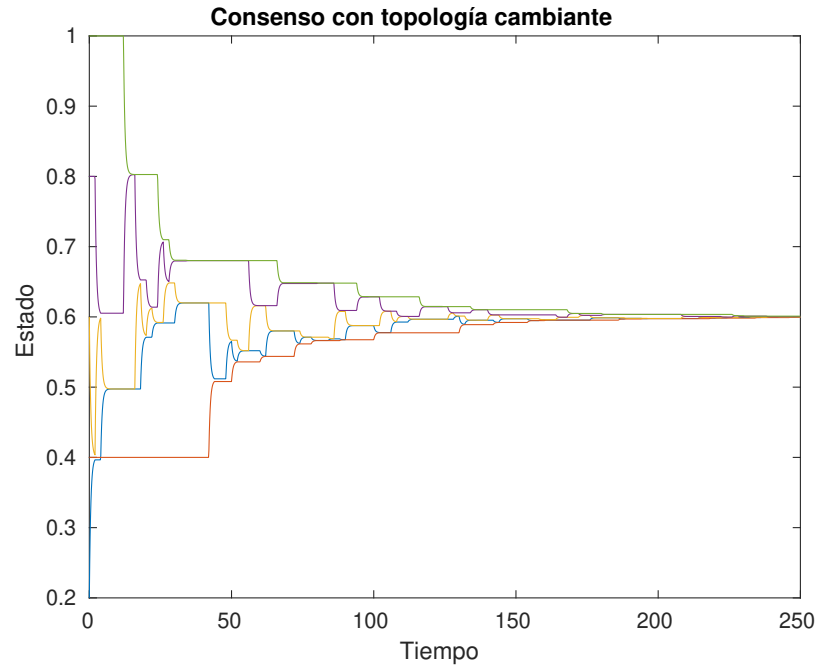
punto 2.-

Considerando el grafo como no dirigido se obtienen los siguientes resultados



Pregunta 3.-

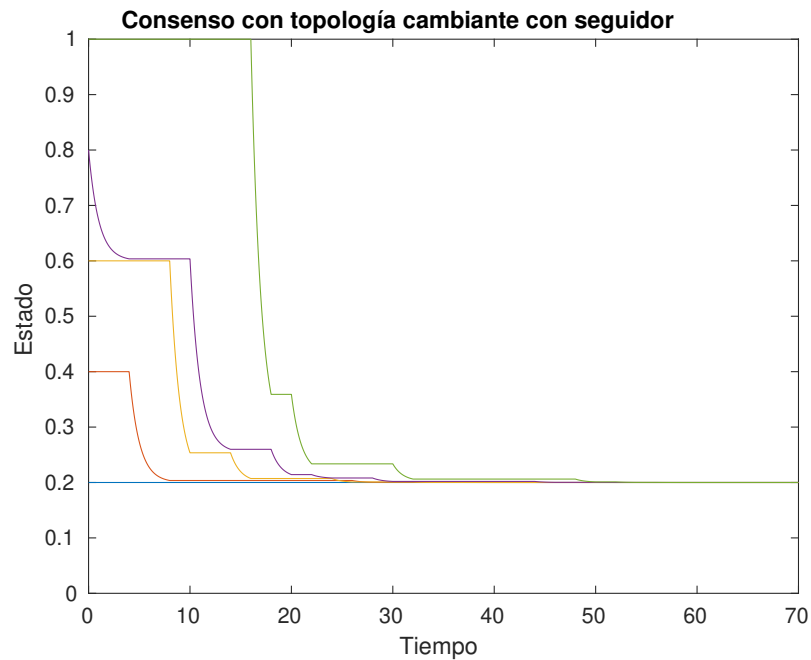
Considerando las condiciones iniciales $x_0 = [0.2; 0.4; 0.6; 0.8; 1]$ y con una ley de control con un periodo de muestreo de $0.1s$, cambiando la topología cada 2 segundos, además, cambiando los grafos de la diapositiva 9 por grafos no dirigidos, se obtienen los siguientes resultados



Los cuales convergen al promedio de las condiciones iniciales.

Pregunta 4.-

Para integradores simples con las mismas condiciones iniciales que la pregunta anterior, pero considerando el caso líder seguidor, es decir, $\mathcal{G}_5^l \triangleq \mathcal{G} \setminus \mathcal{G}_{5(1)}$, y ahora considerando los grafos dirigidos como en las diapositivas, se obtiene el siguiente resultado



Pregunta 5.-

En cuanto a los algoritmos de consenso con referencia, la implementación más sencilla se da en el caso de seguimiento de referencias constantes. Sin embargo, cuando se trata de referencias variantes en el tiempo, la correcta elección de las ganancias es crucial. El consenso se logra en ambos escenarios (referencias constantes y variantes en el tiempo), pero la selección adecuada del algoritmo es fundamental para asegurar el seguimiento eficaz de la referencia deseada.

Respecto al consenso con topologías cambiantes, se observó que el consenso puede lograrse tanto con un esquema de líder-seguidor como sin él. La definición del intervalo de tiempo para cada cambio de topología es un factor clave que influye significativamente en la velocidad de convergencia del sistema. Una correcta elección del tiempo de cambio entre topologías puede mejorar notablemente la rapidez con la que se alcanza el consenso.

Anexo (código usado)

Pregunta 1

```

1 clc
2 clearvars
3 close all
4 %Matriz de grado

```

```

5  %Primer punto
6  D1=[1 0 0 0 0;
7      0 2 0 0 0;
8      0 0 1 0 0;
9      0 0 0 1 0;
10     0 0 0 0 0];
11  D2=[2 0 0 0 0;
12     0 3 0 0 0;
13     0 0 2 0 0;
14     0 0 0 2 0;
15     0 0 0 0 0];
16  D3=[3 0 0 0 0;
17     0 1 0 0 0;
18     0 0 2 0 0;
19     0 0 0 1 0;
20     0 0 0 0 0];
21  D4=[1 0 0 0 0;
22     0 2 0 0 0;
23     0 0 1 0 0;
24     0 0 0 2 0;
25     0 0 0 0 0];
26
27
28
29  %Matriz de adyacencia
30  %Primer punto
31  A1=[0 0 0 0 1;
32      1 0 1 0 0;
33      0 1 0 0 0;
34      1 0 0 0 0;
35      0 0 0 0 0];
36  A2=[0 0 1 0 1;
37      1 0 1 0 1;
38      0 1 0 0 1;
39      1 0 0 0 1;
40      0 0 0 0 0];
41  A3=[0 1 1 1 0;
42      0 0 1 0 0;
43      0 1 0 0 1;
44      0 0 0 0 1;
45      0 0 0 0 0];
46  A4=[0 0 1 0 0;
47      1 0 1 0 0;
48      0 1 0 0 0;
49      1 0 0 0 1;
50      0 0 0 0 0];
51
52
53
54  %hago el objeto grafo, pero aun no lo ploteo
55  G1=digraph(A1');
56  G2=digraph(A2');
57  G3=digraph(A3');
58  G4=digraph(A4');
59
60  %Nodos
61  n=size(A1,1);

```

```

62 %
63 %Laplaciano
64 L1=D1-A1
65 L2=D2-A2
66 L3=D3-A3
67 L4=D4-A4
68 %
69
70 %Caso especial inciso d)
71 L=[3 -1 -1 -1 0;
72     -1 2 -1 0 0;
73     -1 -1 2 0 0;
74     -1 0 0 2 -1;
75     0 0 0 0 0];
76 %Caso especial inciso f)
77 Lf=[1 0 0 0 -1;
78     0 1 0 0 -1;
79     0 0 1 0 -1;
80     0 0 0 1 -1;
81     0 0 0 0 0];
82
83 %Calculo de los eigenvalores
84 e1=eig(-L1);
85 e2=eig(-L2);
86 e3=eig(-L3);
87 e4=eig(-L4);
88 svd(-L1)
89 svd(-L2)
90 svd(-L3)
91 svd(-L4)
92 %
93 %Hago el producto Kronecker
94 m=1; %dimensiones que estoy considerando
95 Im=eye(m);
96 Kron=kron(L,Im); %Este es para el inciso d)
97 Kronf=kron(Lf,Im); %Este es para el inciso f)
98 Kron1= kron(L1,Im);
99 Kron2= kron(L2,Im);
100 Kron3= kron(L3,Im);
101 Kron4= kron(L4,Im);
102 %
103 % %% Esta parte es para la simulacion del sistema
104
105 % %Valores iniciales
106 ref=1; %La referencia que quiero que sigan
107 xi0=[-0.5;0;.5;1.5;ref];%Guardo los valores iniciales para
    referencia
108 xi=xi0;xif=xi0;xi1=xi0;xi2=xi0;xi3=xi0;xi4=xi0;xi5=xi0;%Uso estos
    valores en la aproximacion de Euler
109 %
110 %periodo de muestreo
111 Dt=0.01;
112 tiempo=20; %segundos
113 iteraciones=tiempo/Dt;
114
115 %Simulo el sistema
116 for k=1:iteraciones

```

```

117 %Aproximacion de Euler con Gamma_kron
118 xi(:,k+1)=xi(:,k)-Dt*(Kron*xi(:,k)); %Inciso d)
119 xif(:,k+1)=xif(:,k)-Dt*(Kronf*xif(:,k)); %Inciso f)
120 xi1(:,k+1)=xi1(:,k)-Dt*(Kron1*xi1(:,k));
121 xi2(:,k+1)=xi2(:,k)-Dt*(Kron2*xi2(:,k));
122 xi3(:,k+1)=xi3(:,k)-Dt*(Kron3*xi3(:,k));
123 xi4(:,k+1)=xi4(:,k)-Dt*(Kron4*xi4(:,k));
124 end
125
126
127 %Vectores de tiempo para graficar
128 t=linspace(0,tiempo,iteraciones+1);
129
130 figure
131 subplot(2,2,1)
132 plot(t,xi1);
133 title('inciso a')
134 subplot(2,2,2)
135 plot(t,xi2);
136 title('inciso b')
137 subplot(2,2,3)
138 plot(t,xi3);
139 title('inciso c')
140 subplot(2,2,4)
141 plot(t,xi4);
142 title('inciso d')
143
144 figure
145 plot(t,xi);
146 title('inciso d')
147 xlabel('Tiempo s')
148 ylabel('Estados')
149
150 figure
151 plot(t,xif);
152 title('inciso f')
153 xlabel('Tiempo s')
154 ylabel('Estados')
155
156 figure
157 subplot(2,2,1)
158 plot(G1);
159 title('inciso a')
160 subplot(2,2,2)
161 plot(G2);
162 title('inciso b')
163 subplot(2,2,3)
164 plot(G3);
165 title('inciso c')
166 subplot(2,2,4)
167 plot(G4);
168 title('inciso d')

```

Pregunta 2

```
1 clc
```

```

2 clearvars
3 close all
4
5 % Parametros
6 num_agents = 4; % Numero de agentes
7 T = 20; % Tiempo de simulacion
8 Dt = 0.01; % Paso de tiempo
9 iteraciones = T / Dt;
10
11 % Matriz de adyacencia
12 A1 = [0 0 0 0;
13       1 0 1 0;
14       0 1 0 0;
15       1 0 0 0];
16 % Matriz de adyacencia segundo caso
17 % A1=[0 1 0 1;
18 %     1 0 1 0;
19 %     0 1 0 0;
20 %     1 0 0 0];
21
22 K = ones(num_agents); % Matriz de coeficientes k_{ij}
23 alpha = ones(num_agents, 1); % Vector de coeficientes alpha_i
24 gamma = 50*ones(num_agents, 1); % Vector de coeficientes gamma_i
25
26 % Inicializacion
27 xi = zeros(num_agents, iteraciones); % Estados de los agentes
28 xi_dot = zeros(num_agents, iteraciones); % Derivadas de los estados
29
30 % Inicializacion de los estados de los agentes
31 xi(:, 1) = [1.5;.5;0;-0.5]; % Estados iniciales estados
32
33 % Primera referencia
34 xi_r1 = cos(0:Dt:(iteraciones-1)*Dt); % Primera referencia variante
    con el tiempo
35
36 % Dinamica de la primera referencia
37 f1 = @(t, xi_r) -sin(t); % Derivada de la referencia
38
39 % Simulacion con la primera referencia
40 xi1 = xi; % Copia de los estados iniciales
41 for k = 1:iteraciones-1
42     t = (k-1) * Dt;
43
44     % Derivada de la referencia en el tiempo actual
45     xi_r1_dot = f1(t, xi_r1(k));
46
47     for i = 1:num_agents
48         if i == 1 % Vehiculo con acceso a la referencia
49             u_i = xi_r1_dot - sum(A1(i, :) .* K(i, :) .* (xi1(i, k)
                - xi1(:, k))) - alpha(i) * (xi1(i, k) - xi_r1(k))
                ;
50             else % Otros vehiculos
51                 sum_gk = sum(A1(i, :) .* K(i, :));
52                 if sum_gk == 0
53                     u_i = 0;
54                 else
55                     u_i = (1 / sum_gk) * sum(A1(i, :) .* K(i, :) .* (

```

```

        xi_dot(:, k)' - gamma(i) * (xi1(i, k) - xi1(:,
        k)'))));
56     end
57     end
58     xi_dot(i, k) = u_i;
59 end
60
61 % Actualizacion de los estados usando la aproximacion de Euler
62 xi1(:, k+1) = xi1(:, k) + Dt * xi_dot(:, k);
63 end
64
65 % Segunda referencia
66 xi_r2 = zeros(1, iteraciones);
67 xi_r2(1) = 0.5; % Estado inicial de la segunda referencia
68
69 % Dinamica de la segunda referencia
70 f2 = @(t, xi_r) sin(t) * sin(2 * xi_r); % Derivada de la segunda
    referencia
71
72 % Simulacion con la segunda referencia
73 xi2 = xi; % Copia de los estados iniciales
74 for k = 1:iteraciones-1
75     t = (k-1) * Dt;
76
77     % Derivada de la referencia en el tiempo actual
78     xi_r2_dot = f2(t, xi_r2(k));
79
80     % Actualizacion de la segunda referencia
81     xi_r2(k+1) = xi_r2(k) + Dt * xi_r2_dot;
82
83     for i = 1:num_agents
84         if i == 1 % Vehiculo con acceso a la referencia
85             u_i = xi_r2_dot - sum(A1(i, :) .* K(i, :) .* (xi2(i, k)
                - xi2(:, k)')) - alpha(i) * (xi2(i, k) - xi_r2(k))
                ;
86             else % Otros vehiculos
87                 sum_gk = sum(A1(i, :) .* K(i, :));
88                 if sum_gk == 0
89                     u_i = 0;
90                 else
91                     u_i = (1 / sum_gk) * sum(A1(i, :) .* K(i, :) .* (
                        xi_dot(:, k)' - gamma(i) * (xi2(i, k) - xi2(:,
                        k)'))));
92                 end
93             end
94             xi_dot(i, k) = u_i;
95         end
96
97         % Actualizacion de los estados usando la aproximacion de Euler
98         xi2(:, k+1) = xi2(:, k) + Dt * xi_dot(:, k);
99     end
100
101 % Ploteo de resultados
102 time = 0:Dt:(iteraciones-1)*Dt;
103
104 figure;
105

```

```

106 % Subplot para la primera referencia
107 subplot(2, 1, 1);
108 hold on;
109 for i = 1:num_agents
110     plot(time, xi1(i, :));
111 end
112 plot(time, xi_r1, '--', 'LineWidth', 2); % Primera referencia
113 xlabel('Tiempo');
114 ylabel('Estado');
115 title('Primera referencia');
116 hold off;
117
118 % Subplot para la segunda referencia
119 subplot(2, 1, 2);
120 hold on;
121 for i = 1:num_agents
122     plot(time, xi2(i, :));
123 end
124 plot(time, xi_r2, ':', 'LineWidth', 2); % Segunda referencia
125 xlabel('Tiempo');
126 ylabel('Estado');
127 title('Segunda referencia');
128 hold off;

```

Pregunta 3

```

1  clc
2  clearvars
3  close all
4
5  %Laplacianos
6  L1=[1 0 -1 0 0;
7      0 0 0 0 0;
8      -1 0 1 0 0;
9      0 0 0 0 0;
10     0 0 0 0 0];
11
12  L2=[1 -1 0 0 0;
13      -1 1 0 0 0;
14      0 0 0 0 0;
15      0 0 0 0 0;
16      0 0 0 0 0];
17
18  L3=[1 0 -1 0 0;
19      0 0 0 0 0;
20      -1 0 1 0 0;
21      0 0 0 0 0;
22      0 0 0 0 0];
23
24  L4=[0 0 0 0 0;
25      0 0 0 0 0;
26      0 0 1 -1 0;
27      0 0 -1 1 0;
28      0 0 0 0 0];
29
30  L5=[0 0 0 0 0;

```



```

31     0 0 0 0 0;
32     0 0 0 0 0;
33     0 0 0 1 -1;
34     0 0 0 -1 1];
35
36 %
37 %Hago el producto Kronecker
38 m=1; %dimensiones que estoy considerando
39 Im=eye(m);
40 Kron1= kron(L1,Im);
41 Kron2= kron(L2,Im);
42 Kron3= kron(L3,Im);
43 Kron4= kron(L4,Im);
44 Kron5= kron(L5,Im);
45 %
46 % %% Esta parte es para la simulacion del sistema
47
48 % %%Valores iniciales
49 xi0=[0.2;0.4;0.6;0.8;1];%Guardo los valores iniciales para
    referencia
50 xi=xi0;%Uso estos valores en la aproximacion de Euler
51 %
52 %periodo de muestreo
53 Dt=0.01;
54 tiempo=100; %segundos
55 iteraciones=tiempo/Dt;
56
57 %Simulo el sistema
58 for k=1:iteraciones
59     % Seleccion aleatoria de la topologia cada dos segundos
60     if mod(k, 200) == 1
61         rand_index = randi([1, 5]);
62     end
63
64     % Seleccion del producto de Kronecker correspondiente
65     switch rand_index
66     case 1
67         Kron = Kron1;
68     case 2
69         Kron = Kron2;
70     case 3
71         Kron = Kron3;
72     case 4
73         Kron = Kron4;
74     case 5
75         Kron = Kron5;
76     end
77     %Aproximacion de Euler con Gamma_kron
78     xi(:,k+1)=xi(:,k)-Dt*(Kron*xi(:,k));
79 end
80
81
82 %Vectores de tiempo para graficar
83 t=linspace(0,tiempo,iteraciones+1);
84
85
86 figure

```

```

87 plot(t,xi);
88 xlabel('Tiempo');
89 ylabel('Estado');
90 title('Consenso con topologia cambiante')

```

Pregunta 4

```

1      clc
2      clearvars
3      close all
4
5      %Laplacianos
6
7      L2=[0 0 0 0 0;
8          -1 1 0 0 0;
9          0 0 0 0 0;
10         0 0 0 0 0;
11         0 0 0 0 0];
12
13      L3=[0 0 0 0 0;
14          0 0 0 0 0;
15          -1 0 1 0 0;
16          0 0 0 0 0;
17          0 0 0 0 0];
18
19      L4=[0 0 0 0 0;
20          0 0 0 0 0;
21          0 0 0 0 0;
22          0 0 -1 1 0;
23          0 0 0 0 0];
24
25      L5=[0 0 0 0 0;
26          0 0 0 0 0;
27          0 0 0 0 0;
28          0 0 0 0 0;
29          0 0 0 -1 1];
30
31      %
32      %Hago el producto Kronecker
33      m=1; %dimensiones que estoy considerando
34      Im=eye(m);
35      Kron2= kron(L2,Im);
36      Kron3= kron(L3,Im);
37      Kron4= kron(L4,Im);
38      Kron5= kron(L5,Im);
39      %
40      % %% Esta parte es para la simulacion del sistema
41
42      % %Valores iniciales
43      xi0=[0.2;0.4;0.6;0.8;1];%Guardo los valores iniciales para
44      xi=xi0;%Uso estos valores en la aproximacion de Euler
45      %
46      %periodo de muestreo
47      Dt=0.01;
48      tiempo=70; %segundos

```

```

49 iteraciones=tiempo/Dt;
50
51 %Simulo el sistema
52 for k=1:iteraciones
53     % Seleccion aleatoria de la topologia cada dos segundos
54     if mod(k, 200) == 1
55         rand_index = randi([1, 4]);
56     end
57
58     % Seleccion del producto de Kronecker correspondiente
59     switch rand_index
60     case 1
61         Kron = Kron2;
62     case 2
63         Kron = Kron3;
64     case 3
65         Kron = Kron4;
66     case 4
67         Kron = Kron5;
68     end
69     %Aproximacion de Euler con Gamma_kron
70     xi(:,k+1)=xi(:,k)-Dt*(Kron*xi(:,k));
71 end
72
73
74 %Vectores de tiempo para graficar
75 t=linspace(0,tiempo,iteraciones+1);
76
77
78 figure
79 plot(t,xi);
80 xlabel('Tiempo');
81 ylabel('Estado');
82 title('Consenso con topologia cambiante con seguidor')
83 ylim([0 1])

```