



**Cinvestav**

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

## **Tarea 9. Control de formación basado en distancia**

Presentado por

**Jesús Alejandro Díaz Hernández**

Presentado para el curso de  
**Tópicos avanzados de control 2**

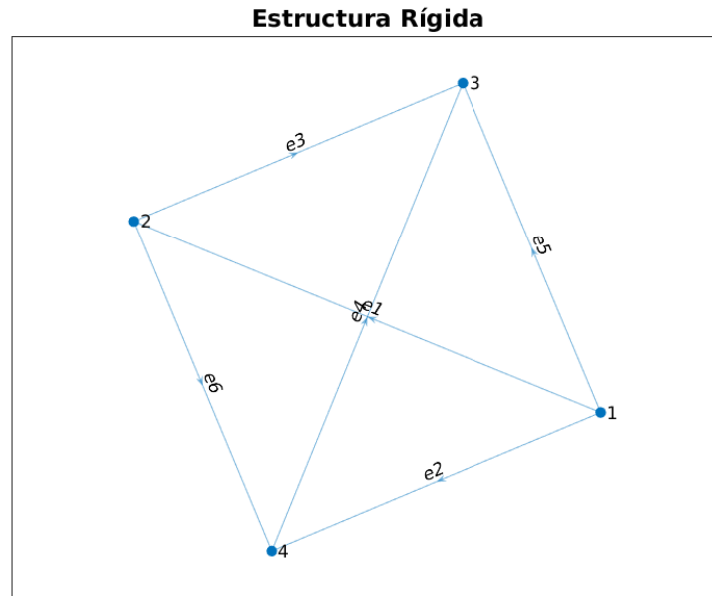
Curso impartido por: Héctor Manuel Becerra Fermín  
Profesor

Guadalajara, Jalisco

31 Julio 2024

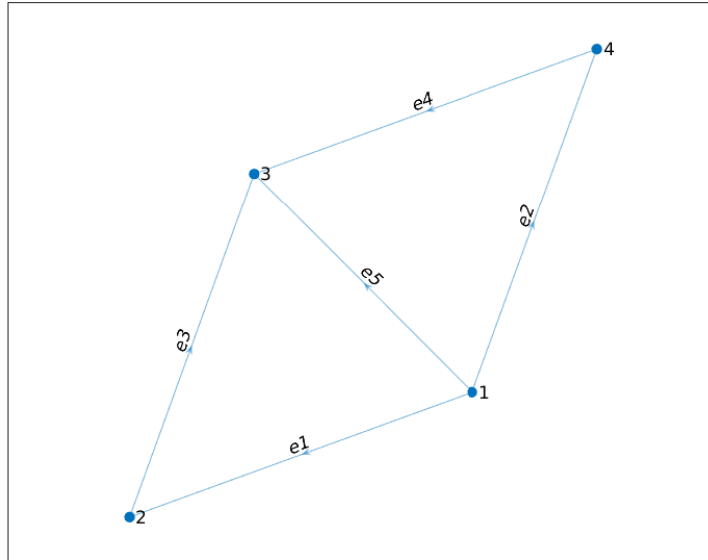
## Pregunta 1.-

Los grafos que estaremos usando durante todo el reporte, son los siguientes. Es importante destacar que las direcciones son consideradas al azar para calcular la matriz de incidencia. El grafo de la estructura rígida es:



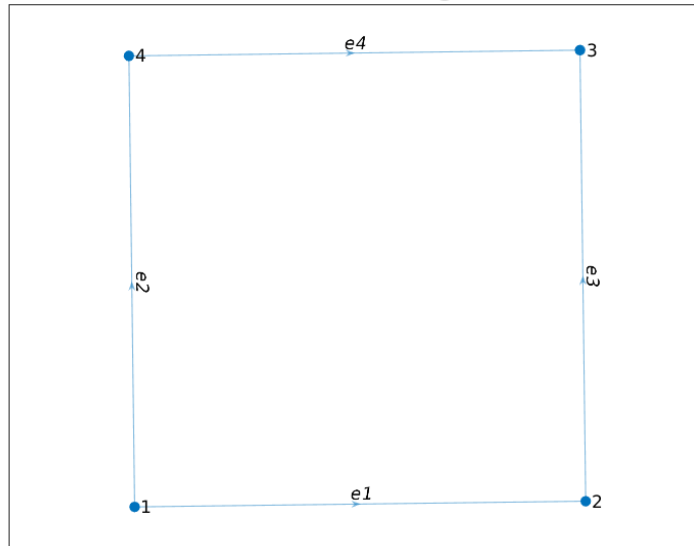
El grafo de la estructura mínimamente rígida es:

**Estructura Minimamente Rígida**



El grafo de la estructura no rígida es:

**Estructura No Rígida**



## Pregunta 2.-

La matriz de rigidez de la estructura rígida es:

$$\begin{bmatrix} p_2^x - p_1^x & p_2^y - p_1^y & p_1^x - p_2^x & p_1^y - p_2^y & 0 & 0 & 0 & 0 \\ p_4^x - p_1^x & p_4^y - p_1^y & 0 & 0 & 0 & 0 & p_1^x - p_4^x & p_1^y - p_4^y \\ 0 & 0 & p_3^x - p_2^x & p_3^y - p_2^y & p_2^x - p_3^x & p_2^y - p_3^y & 0 & 0 \\ 0 & 0 & p_4^x - p_3^x & p_4^y - p_3^y & 0 & 0 & p_3^x - p_4^x & p_3^y - p_4^y \\ p_3^x - p_1^x & p_3^y - p_1^y & 0 & 0 & p_1^x - p_3^x & p_1^y - p_3^y & 0 & 0 \\ 0 & 0 & p_4^x - p_2^x & p_4^y - p_2^y & 0 & 0 & p_2^x - p_4^x & p_2^y - p_4^y \end{bmatrix}$$

La mínimamente rígida es

$$\begin{bmatrix} p_2^x - p_1^x & p_2^y - p_1^y & p_1^x - p_2^x & p_1^y - p_2^y & 0 & 0 & 0 & 0 \\ p_4^x - p_1^x & p_4^y - p_1^y & 0 & 0 & 0 & 0 & p_1^x - p_4^x & p_1^y - p_4^y \\ 0 & 0 & p_3^x - p_2^x & p_3^y - p_2^y & 0 & 0 & p_2^x - p_3^x & p_2^y - p_3^y \\ 0 & 0 & 0 & 0 & p_4^x - p_3^x & p_4^y - p_3^y & p_3^x - p_4^x & p_3^y - p_4^y \\ p_3^x - p_1^x & p_3^y - p_1^y & 0 & 0 & p_1^x - p_3^x & p_1^y - p_3^y & 0 & 0 \end{bmatrix}$$

y de la no rígida

$$\begin{bmatrix} p_2^x - p_1^x & p_2^y - p_1^y & p_1^x - p_2^x & p_1^y - p_2^y & 0 & 0 & 0 & 0 \\ p_4^x - p_1^x & p_4^y - p_1^y & 0 & 0 & 0 & 0 & p_1^x - p_4^x & p_1^y - p_4^y \\ 0 & 0 & p_3^x - p_2^x & p_3^y - p_2^y & p_2^x - p_3^x & p_2^y - p_3^y & 0 & 0 \\ 0 & 0 & 0 & 0 & p_4^x - p_3^x & p_4^y - p_3^y & p_3^x - p_4^x & p_3^y - p_4^y \end{bmatrix}$$

## Pregunta 3.-

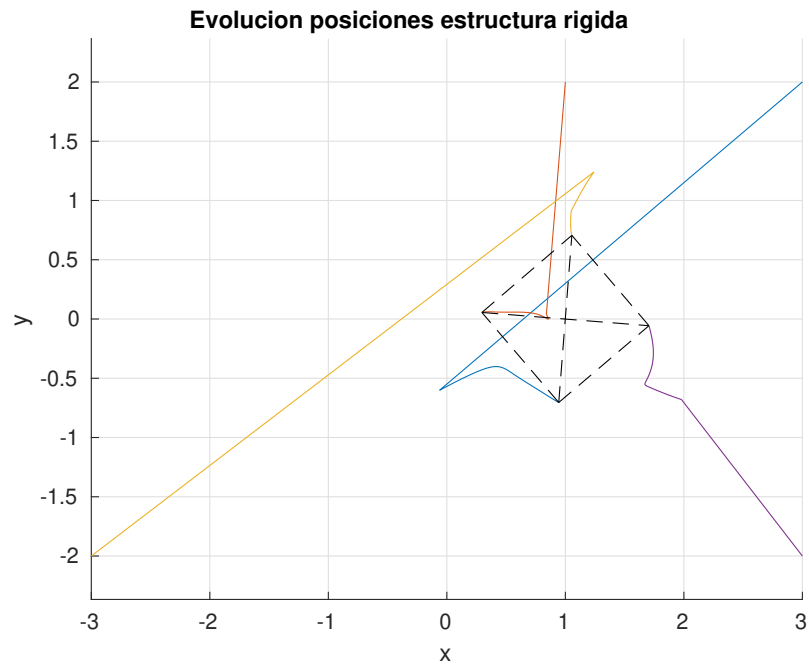
Considerando posiciones genéricas no colineales ni el mismo punto, como:  $p_1 = [0 \ 0]^T$ ,  $p_2 = [1 \ 0]^T$ ,  $p_3 = [1 \ -1]^T$ ,  $p_4 = [0 \ -1]^T$  verificando el rango de la matriz de rigidez tenemos: estructura rígida  $rank(R) = 6$ , la estructura mínimamente rígida  $rang(R) = 5$ , y la no rígida  $rank(R) = 4$ . Considerando que un framework  $(\mathcal{G}, p)$  es infinitesimalmente rígida si solo si  $rank(R) = 2|\mathcal{V}| - 3$ , y que en nuestro caso  $|\mathcal{V}| = 4$  es infinitesimalmente rígida si solo si  $rank(R) = 5$  como la estructura mínimamente rígida

## Pregunta 4.-

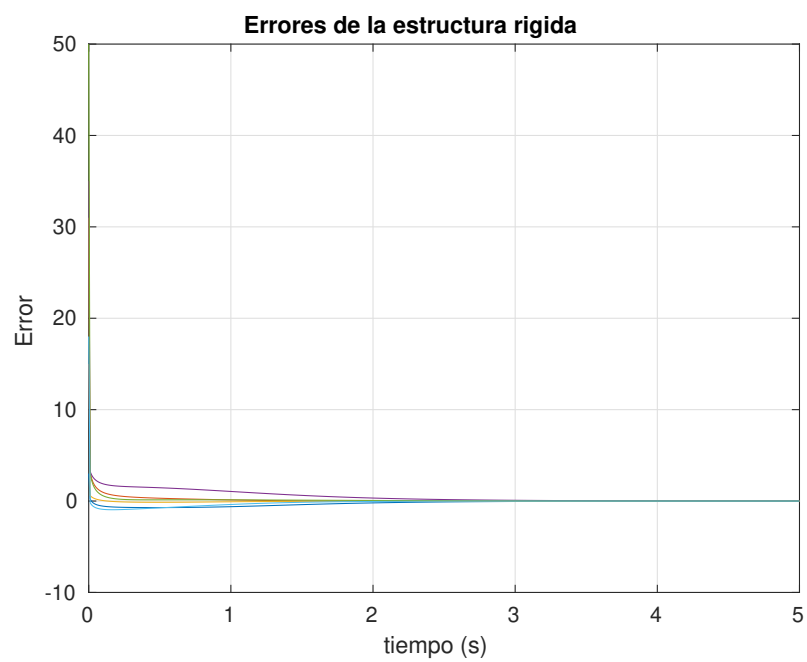
Para los mismos posiciones que en el punto anterior tenemos los siguientes eigenvalores: estructura rígida  $[0 \ 0 \ 0.17 \ 0.70 \ 2 \ 2.31 \ 4.48 \ 6.32]^T$  y el valor de propio de rigidez es  $\lambda_4 = 0.70$ . La estructura mínimamente rígida  $[0 \ 0 \ 0 \ 0.68 \ 1.42 \ 2 \ 2.86 \ 5.03]^T$  y su valor propio de rigidez es  $\lambda_4 = 0.68$ . Finalmente, para la estructura no rígida  $[0 \ 0 \ 0 \ 0 \ 2 \ 2 \ 2 \ 2]^T$  y su valor propio de rigidez es  $\lambda_4 = 0$ .

## Pregunta 5.-

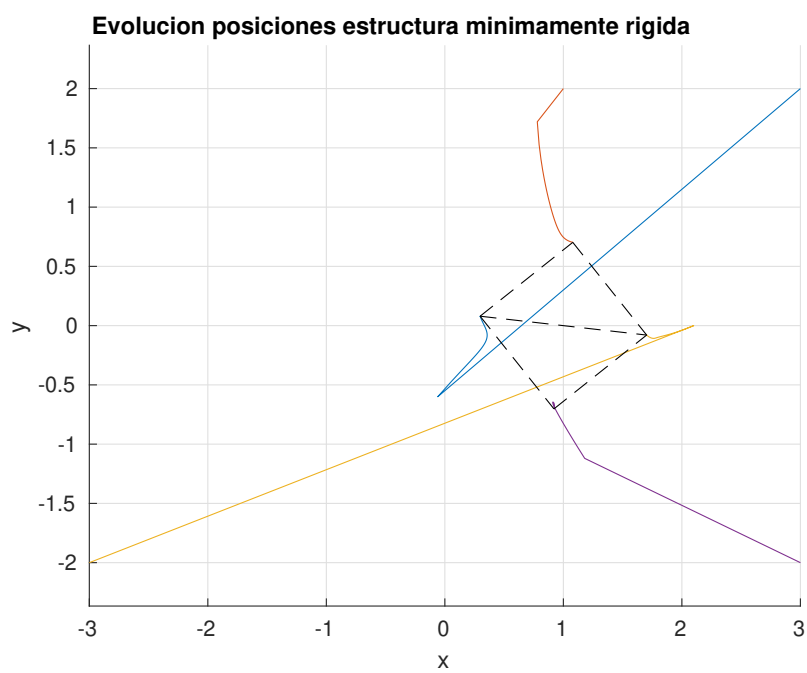
Implementando el control basado en distancia para cada uno de los grafos anteriores considerando las posiciones deseadas dadas por las distancias de separación entre los puntos de la pregunta 3, e inicializándolos al en posiciones aleatorias obtenemos las siguientes trayectorias para: la estructura rígida



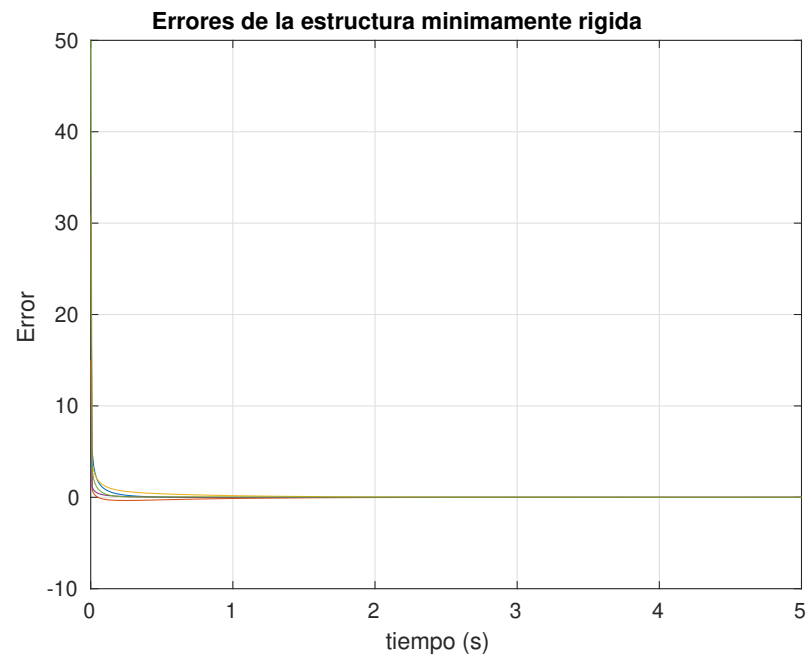
y la evolución de los errores sigmas



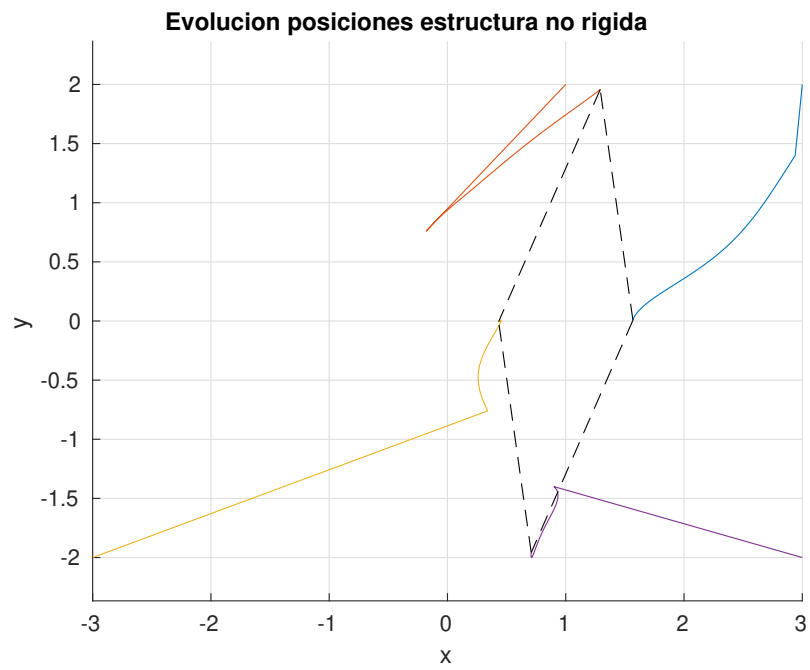
la mínimamente rígida



y la evolución de los errores sigmas

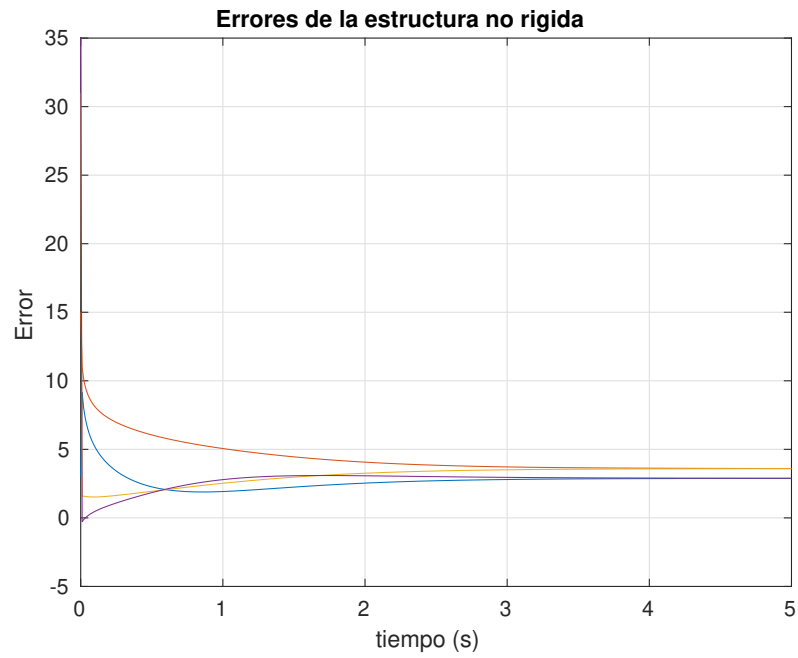


la estructura no rígida



y la evolución de los errores sigmas

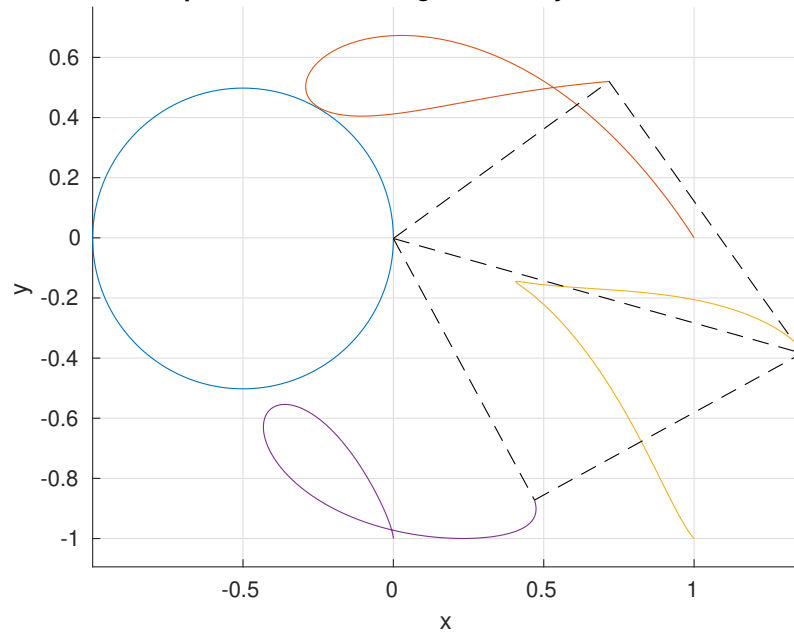




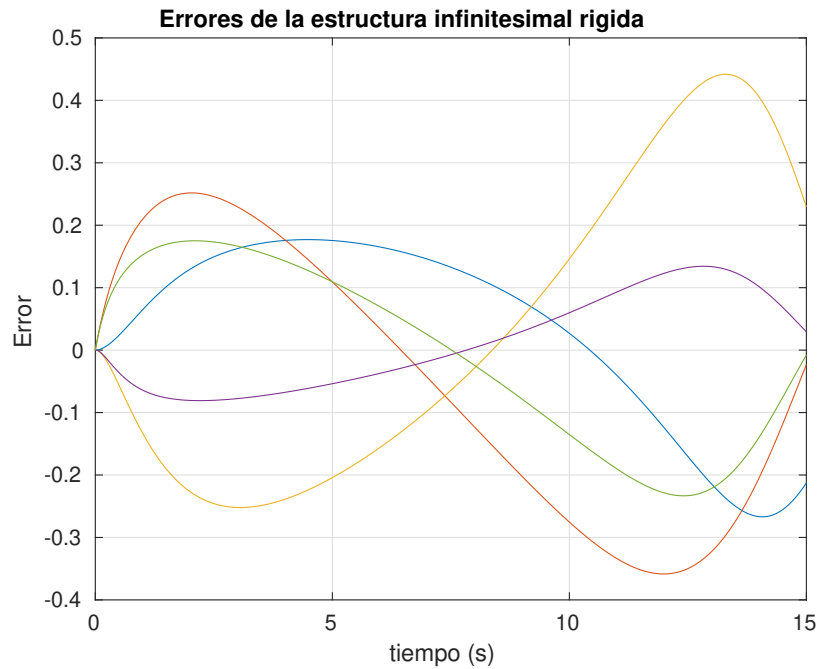
### Pregunta 6.-

Considerando el grafo de la estructura mínimamente rígida el cual como se vio en la pregunta 3 es infinitesimalmente rígida, pero con un nodo siguiendo una trayectoria circular y los otros retroalimentados con el control distribuido basado en distancia obtenemos las siguientes gráficas de las trayectorias la estructura rígida

**Evolución de posiciones con un agente en trayectoria circular**



y la evolución de los errores sigmas



## Pregunta 7.-

Como se puede observar, la estructura rígida es la que menos se deforma, mientras que la estructura no rígida presenta deformaciones significativas. En el caso donde uno de los agentes sigue una trayectoria circular, el error no se reduce a cero; sin embargo, permanece acotado. Esto indica que, aunque los agentes no alcanzan la configuración deseada de manera exacta, la desviación del error está limitada.

## Anexo (código usado)

### Pregunta 5

```

1  clc
2  close all
3  clearvars
4
5  % Coordenadas deseadas
6  p1 = [0; 0];
7  p2 = [1; 0];
8  p3 = [1; -1];
9  p4 = [0; -1];
10
11 % Definir las diferencias cuadradas normadas d^2

```

```

12 d2 = [(p1 - p2)'*(p1 - p2);
13       (p1 - p4)'*(p1 - p4);
14       (p2 - p3)'*(p2 - p3);
15       (p4 - p3)'*(p4 - p3);
16       (p1 - p3)'*(p1 - p3);
17       (p2 - p4)'*(p2 - p4)];
18 d2_2 = [(p1 - p2)'*(p1 - p2);
19          (p1 - p4)'*(p1 - p4);
20          (p2 - p3)'*(p2 - p3);
21          (p4 - p3)'*(p4 - p3);
22          (p1 - p3)'*(p1 - p3)];
23 d2_3 = [(p1 - p2)'*(p1 - p2);
24          (p1 - p4)'*(p1 - p4);
25          (p2 - p3)'*(p2 - p3);
26          (p4 - p3)'*(p4 - p3)];
27 %producto kronecker
28 I=eye(2);
29
30 % Periodo de muestreo
31 Dt = 0.01;
32 tiempo = 5; % segundos
33 iteraciones = tiempo / Dt;
34
35 % Posiciones iniciales de los nodos
36 pInit = [3;2;1;2;-3;-2;3;-2];
37
38 %Inicializo el valor de p
39 p = zeros(8,iteraciones);
40 p(:,1)=pInit;
41 p2 = zeros(8,iteraciones);
42 p2(:,1)=pInit;
43 p3 = zeros(8,iteraciones);
44 p3(:,1)=pInit;
45 sigma=zeros(6,iteraciones);
46 sigma_2=zeros(5,iteraciones);
47 sigma_3=zeros(4,iteraciones);
48
49 % Simulacion con aproximacion de Euler
50 for k = 1:iteraciones
51     %% RIGIDA
52     e1=p(3:4,k)-p(1:2,k);%e1=p2-p1
53     e2=p(7:8,k)-p(1:2,k);%e2=p4-p1
54     e3=p(5:6,k)-p(3:4,k);%e3=p3-p2
55     e4=p(5:6,k)-p(7:8,k);%e4=p3-p4
56     e5=p(5:6,k)-p(1:2,k);%e5=p3-p1
57     e6=p(7:8,k)-p(3:4,k);%e6=p4-p2
58     e=blkdiag(e1',e2',e3',e4',e5',e6');
59     %Calculo los valores de sigma
60     sigma1=e1'*e1-d2(1);
61     sigma2=e2'*e2-d2(2);
62     sigma3=e3'*e3-d2(3);
63     sigma4=e4'*e4-d2(4);
64     sigma5=e5'*e5-d2(5);
65     sigma6=e6'*e6-d2(6);
66     sigma(:,k)=[sigma1;sigma2;sigma3;sigma4;sigma5;sigma6];
67     %Matrices de incidencia locales estructura rigida
68     E = [1 -1 0 0;

```

```

69         1  0  0 -1;
70         0  1 -1  0;
71         0 -1  0  1;
72         1  0 -1  0;
73         0  1  0 -1];
74     kronp=kron(E,I);
75     %Calculo del vector R
76     R=e*kronp;
77
78     % Calcular la derivada de p
79     u = -R' * R * p(:,k) - R' * d2;
80
81     % Aproximacion de Euler
82     p(:,k+1) = p(:,k) + Dt * u;
83
84     %% MINIMAMENTE RIGIDA
85
86     e1_2=p2(3:4,k)-p2(1:2,k);%e1=p2-p1
87     e2_2=p2(7:8,k)-p2(1:2,k);%e2=p4-p1
88     e3_2=p2(5:6,k)-p2(3:4,k);%e3=p3-p2
89     e4_2=p2(5:6,k)-p2(7:8,k);%e4=p3-p4
90     e5_2=p2(5:6,k)-p2(1:2,k);%e5=p3-p1
91     e2=blkdiag(e1_2',e2_2',e3_2',e4_2',e5_2');
92     %Calculo los valores de sigma
93     sigma1_2=e1_2'*e1_2-d2(1);
94     sigma2_2=e2_2'*e2_2-d2(2);
95     sigma3_2=e3_2'*e3_2-d2(3);
96     sigma4_2=e4_2'*e4_2-d2(4);
97     sigma5_2=e5_2'*e5_2-d2(5);
98
99     sigma_2(:,k)=[sigma1_2;sigma2_2;sigma3_2;sigma4_2;sigma5_2];
100     %Matrices de incidencia locales cosa rigida
101     E2 = [1 -1  0  0;
102           1  0  0 -1;
103           0  1  0 -1;
104           0  0 -1  1;
105           1  0 -1  0];
106     kronp2=kron(E2,I);
107     %Calculo del vector R
108     R2=e2*kronp2;
109     % Calcular la derivada de p
110     u2 = -R2' * R2 * p2(:,k) - R2' * d2_2;
111
112     % Aproximacion de Euler
113     p2(:,k+1) = p2(:,k) + Dt * u2;
114
115     %% NO RIGIDA
116     e1_3=p3(3:4,k)-p3(1:2,k);%e1=p2-p1
117     e2_3=p3(7:8,k)-p3(1:2,k);%e2=p4-p1
118     e3_3=p3(5:6,k)-p3(3:4,k);%e3=p3-p2
119     e4_3=p3(5:6,k)-p3(7:8,k);%e4=p3-p4
120     e_3=blkdiag(e1_2',e2_2',e3_2',e4_2');
121     %Calculo los valores de sigma
122     sigma1_3=e1_3'*e1_3-d2(1);
123     sigma2_3=e2_3'*e2_3-d2(2);
124     sigma3_3=e3_3'*e3_3-d2(3);
125     sigma4_3=e4_3'*e4_3-d2(4);

```

```

126
127     sigma_3(:,k)=[sigma1_3;sigma2_3;sigma3_3;sigma4_3];
128     %Matrices de incidencia locales cosa rigida
129     E3 = [1 -1 0 0;
130           1 0 0 -1;
131           0 1 -1 0;
132           0 0 -1 1];
133     kronp3=kron(E3,I);
134     %Calculo del vector R
135     R3=e_3*kronp3;
136     % Calcular la derivada de p
137     u3 = -R3' * R3 * p3(:,k) - R3' * d2_3;
138
139     % Aproximacion de Euler
140     p3(:,k+1) = p3(:,k) + Dt * u3;
141 end
142
143 figure
144 hold on
145 plot(p(1,:),p(2,:))
146 plot(p(3,:),p(4,:))
147 plot(p(5,:),p(6,:))
148 plot(p(7,:),p(8,:))
149 %Lineas entre nodos
150 plot([p(1, end), p(3, end)], [p(2, end), p(4, end)], 'k--');
151 plot([p(1, end), p(7, end)], [p(2, end), p(8, end)], 'k--');
152 plot([p(3, end), p(5, end)], [p(4, end), p(6, end)], 'k--');
153 plot([p(7, end), p(5, end)], [p(8, end), p(6, end)], 'k--');
154 plot([p(1, end), p(5, end)], [p(2, end), p(6, end)], 'k--');
155 plot([p(3, end), p(7, end)], [p(4, end), p(8, end)], 'k--');
156 title('Evolucion posiciones estructura rigida')
157 xlabel('x')
158 ylabel('y')
159 axis equal
160 grid on
161 hold off
162
163 figure
164 hold on
165 plot(p2(1,:),p2(2,:))
166 plot(p2(3,:),p2(4,:))
167 plot(p2(5,:),p2(6,:))
168 plot(p2(7,:),p2(8,:))
169 %Lineas entre nodos
170 plot([p2(1, end), p2(3, end)], [p2(2, end), p2(4, end)], 'k--');
171 plot([p2(1, end), p2(7, end)], [p2(2, end), p2(8, end)], 'k--');
172 plot([p2(3, end), p2(5, end)], [p2(4, end), p2(6, end)], 'k--');
173 plot([p2(7, end), p2(5, end)], [p2(8, end), p2(6, end)], 'k--');
174 plot([p2(1, end), p2(5, end)], [p2(2, end), p2(6, end)], 'k--');
175 title('Evolucion posiciones estructura minimamente rigida')
176 xlabel('x')
177 ylabel('y')
178 axis equal
179 grid on
180 hold off
181
182 figure

```

```

183 hold on
184 plot(p3(1,:),p3(2,:))
185 plot(p3(3,:),p3(4,:))
186 plot(p3(5,:),p3(6,:))
187 plot(p3(7,:),p3(8,:))
188 %Lineas entre nodos
189 plot([p3(1, end), p3(3, end)], [p3(2, end), p3(4, end)], 'k--');
190 plot([p3(1, end), p3(7, end)], [p3(2, end), p3(8, end)], 'k--');
191 plot([p3(3, end), p3(5, end)], [p3(4, end), p3(6, end)], 'k--');
192 plot([p3(7, end), p3(5, end)], [p3(8, end), p3(6, end)], 'k--');
193 title('Evolucion posiciones estructura no rigida')
194 xlabel('x')
195 ylabel('y')
196 axis equal
197 grid on
198 hold off
199
200 t = linspace(0, tiempo, iteraciones);
201 figure
202 plot(t,sigma)
203 title('Errores de la estructura rigida')
204 xlabel('tiempo (s)')
205 ylabel('Error')
206 grid on
207 figure
208 plot(t,sigma_2)
209 title('Errores de la estructura minimamente rigida')
210 xlabel('tiempo (s)')
211 ylabel('Error')
212 grid on
213
214 figure
215 plot(t,sigma_3)
216 title('Errores de la estructura no rigida')
217 xlabel('tiempo (s)')
218 ylabel('Error')
219 grid on

```

## Pregunta 6

```

1      clc
2      close all
3      clearvars
4
5      % Coordenadas deseadas
6      p1 = [0; 0];
7      p2 = [1; 0];
8      p3 = [1; -1];
9      p4 = [0; -1];
10
11     % Definir las diferencias cuadradas normadas d^2
12     d2 =[(p1 - p2)'*(p1 - p2);
13          (p1 - p4)'*(p1 - p4);

```

```

14         (p2 - p3)'*(p2 - p3);
15         (p4 - p3)'*(p4 - p3);
16         (p1 - p3)'*(p1 - p3)];
17
18 %producto kronecker
19 I=eye(2);
20
21 % Periodo de muestreo
22 Dt = 0.01;
23 tiempo = 15; % segundos
24 iteraciones = tiempo / Dt;
25
26 % Posiciones iniciales de los nodos
27 pInit = [0;0;1;0;1;-1;0;-1];
28
29 % Inicializo el valor de p
30 p = zeros(8, iteraciones);
31 p(:, 1) = pInit;
32 sigma = zeros(5, iteraciones);
33
34 % Parametros para la trayectoria circular del agente 1
35 radio = 0.5;
36 frecuencia = 2 * pi / tiempo;
37
38 % Simulacion con aproximacion de Euler
39 for k = 1:iteraciones
40     % Trayectoria circular para el agente 1
41     p(1:2, k) = radio * [cos(frecuencia * k * Dt); sin(frecuencia
        * k * Dt)]-radio*[cos(frecuencia * Dt);sin(frecuencia *
        Dt)];
42     %% MINIMAMENTE RIGIDA
43     e1 = p(3:4, k) - p(1:2, k); % e1 = p2 - p1
44     e2 = p(7:8, k) - p(1:2, k); % e2 = p4 - p1
45     e3 = p(5:6, k) - p(3:4, k); % e3 = p3 - p2
46     e4 = p(5:6, k) - p(7:8, k); % e4 = p3 - p4
47     e5 = p(5:6, k) - p(1:2, k); % e5 = p3 - p1
48     e_2 = blkdiag(e1', e2', e3', e4', e5');
49
50     % Calculo los valores de sigma
51     sigma1 = e1' * e1 - d2(1);
52     sigma2 = e2' * e2 - d2(2);
53     sigma3 = e3' * e3 - d2(3);
54     sigma4 = e4' * e4 - d2(4);
55     sigma5 = e5' * e5 - d2(5);
56     sigma(:, k) = [sigma1; sigma2; sigma3; sigma4; sigma5];
57
58     % Matrices de incidencia locales cosa rigida
59     E = [1 -1 0 0;
60          1 0 0 -1;
61          0 1 0 -1;
62          0 0 -1 1;
63          1 0 -1 0];
64     kronp = kron(E, I);
65
66     % Calculo del vector R
67     R = e_2 * kronp;
68

```



```

69 % Calcular la derivada de p para los nodos controlados
70 if k < iteraciones
71     u = -R' * R * p(:, k) - R' * d2;
72     % Aproximacion de Euler
73     p(3:end, k+1) = p(3:end, k) + Dt * u(3:end);
74 end
75 end
76
77 figure
78 hold on
79 plot(p(1,:), p(2,:))
80 plot(p(3,:), p(4,:))
81 plot(p(5,:), p(6,:))
82 plot(p(7,:), p(8,:))
83 % Lineas entre nodos
84 plot([p(1, end), p(3, end)], [p(2, end), p(4, end)], 'k--');
85 plot([p(1, end), p(7, end)], [p(2, end), p(8, end)], 'k--');
86 plot([p(3, end), p(5, end)], [p(4, end), p(6, end)], 'k--');
87 plot([p(7, end), p(5, end)], [p(8, end), p(6, end)], 'k--');
88 plot([p(1, end), p(5, end)], [p(2, end), p(6, end)], 'k--');
89 title('Evolucion de posiciones con un agente en trayectoria
        circular')
90 xlabel('x')
91 ylabel('y')
92 axis equal
93 grid on
94 hold off
95
96 t = linspace(0, tiempo, iteraciones);
97 figure
98 plot(t, sigma)
99 title('Errores de la estructura infinitesimal rigida')
100 xlabel('tiempo (s)')
101 ylabel('Error')
102 grid on

```