Algoritmo de análisis de Posición, velocidad, aceleración y análisis dinámico de un mecanismo de cuatro barras

Algoritmo elaborado en Octave 6.2.0

Fernando Guiraud 8-945-692

```
%Identificacion del eslabon y junta
a=10;  m2= 2; IG2= 0.1;
b=13;  m3= 3; IG3= 0.2;
c=15;  m4= 4; IG4= 0.3;
d=17;

O2 = [0,0];
O4 = [d,0];

%Puntos de interes
CG2 = 12; deltaCG2 = deg2rad(15);
CG3 = 15; deltaCG3 = deg2rad(30);
CG4 = 10; deltaCG4 = deg2rad(-10);

p = 1.3; deltaP = deg2rad(45);

%identificacion de esfuerzos externos
T4=0.2;
P = 20; deltaFP = deg2rad(30);
FP = P * [cos(deltaFP), sin(deltaFP)];

%posicion inicial t2

%definir veloidad inical y aceleracion angular
n = 0;

%Pos=[]; Vel=[]; Acel=[]; PosAng=[]; VelAng=[]; AcelAng=[]; Torque[];

theta2i = (pi/180)*(0);
w2i = 0.3 ;
alpha2=0.1;

t = 0:0.1:1*(pi)/w2i;

W2 = w2i+ alpha2 * t;


```

```
for theta2 = theta2i + w2i * t + 0.5 * alpha2 * (t.^2)
    n = n + 1;
    w2 = W2(n);

%Punto A

Ax = a * cos(theta2);
Ay = a * sin(theta2);
RA = [ Ax, Ay];

%Punto B

S=(a^2 - b^2 + c^2 - d^2)/(2*(Ax - d));
P = (Ay^2)/(Ax - d)^2 + 1;
Q = (2 * Ay * (d - S) )/ (Ax - d);
R = (d - S)^2 - c^2;

By = (-Q + sqrt(Q^2 - 4 * P * R)) / (2* P);

Bx = S - (Ay * By) / (Ax - d);

RB = [Bx, By];

%Calculo de angulos t3 y t4

theta3 = atan2(By - Ay, Bx - Ax);

theta4 = atan2(By, Bx - d);

%calculo de velocidades angulares

w3 = (a/b) * w2 * (sin(theta4 - theta2) / sin(theta3 - theta4));
w4 = (a/c) * w2 * (sin(theta2 - theta3) / sin(theta4 - theta3));

```

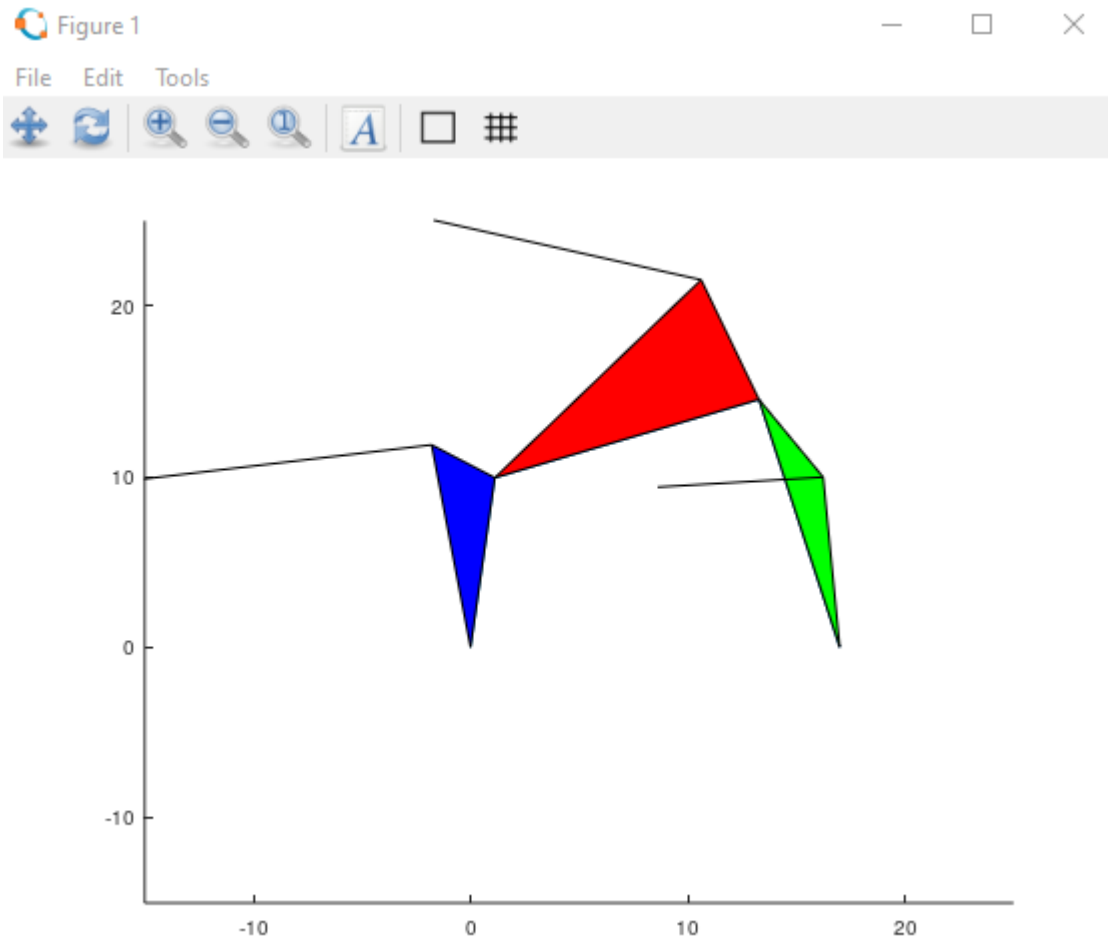
```
%Calculo de Aceleraciones Lineales

VA = a * w2 * [-sin(theta2), cos(theta2)];
VBA = b * w3 * [-sin(theta3), cos(theta3)];
VB = c * w4 * [-sin(theta4), cos(theta4)];

%Calculo de Aceleraciones angulares

A = c * sin(theta4);

B = b * sin(theta3);

C = a * alpha2 * sin(theta2) + a * (w2^2) * cos(theta2) + b * (w3^2) * cos(theta3) - c * (w4^2) * cos(theta4);

D = c * cos(theta4);

E = b * cos(theta3);

F = a * alpha2 * cos(theta2) - a * (w2^2) * sin(theta2) - b * (w3^2) * sin(theta3) + c * (w4^2)*sin(theta4);


alpha3 = (C*D - A*F)/(A*E - B*D);

alpha4 = (C*E - B*F)/(A*E - B*D);

%Calculo de aceleraciones lineales

AA = [-a * alpha2 * sin(theta2) - a * (w2^2) * cos(theta2), a * alpha2 * cos(theta2) - a * (w2^2) * sin(theta2)];

AB = [c * alpha4 * sin(theta4) + c * (w4^2) * cos(theta4), -c * alpha4 * cos(theta4) + c * (w4^2) * sin(theta4)];

ABA = AB - AA;

%Localizacion de los puntos de interes (centro de gravedad)

RCG2 = CG2 * [cos(theta2 + deltaCG2), sin(theta2 + deltaCG2)];

RCG3 = RA + CG3 * [cos(theta3 + deltaCG3), sin(theta3 + deltaCG3)];

RCG4 = O4 + CG4 * [ cos(theta4 + deltaCG4), sin(theta4 + deltaCG4)];

%Localizar el punto P

RP = RCG3 + p * [cos(deltaP), sin(deltaP)];

%Velocidades en los puntos de interes (Centros de Gravedad)

VCG2 = CG2 * w2 * [-sin(theta2 + deltaCG2), cos(theta2 + deltaCG2)];
VCG4 = CG4 * w4 * [-sin(theta4 + deltaCG4), cos(theta4 + deltaCG4)];
VCG3A = CG3 * w3 * [-sin(theta3 + deltaCG3), cos(theta3 + deltaCG3)];

VCG3 = VA - VCG3A;
```

```
124
125 %Aceleraciones de los puntos de interes (Centro de gravedad)
126
127 ACG2 = CG2 * alpha2 * [sin(theta2 + deltaCG2), cos(theta2 + deltaCG2)] - CG2 * (w2^2) * [cos(theta2 + deltaCG2), sin(theta2 + deltaCG2)];
128
129 ACG4 = CG4 * alpha4 * [-sin(theta4 + deltaCG4), cos(theta4 + deltaCG4)] - CG4 * (w4^2) * [cos(theta4 + deltaCG4), sin(theta4 + deltaCG4)];
130
131 ACG3A = CG3 * alpha3 * [-sin(theta3 + deltaCG3), cos(theta3 + deltaCG3)] - CG3 * (w3^2) * [cos(theta3 + deltaCG3), sin(theta3 + deltaCG3)];
132
133 ACG3 = AA + ACG3A;
```

```
134
135 %Vectores de posicion de las reaciones
136
137 R12 = O2 - RCG2;
138 R32 = RA - RCG2;
139 R23 = RA - RCG3;
140 R43 = RB - RCG3;
141 R34 = RB - RCG4;
142 R14 = O4 - RCG4;
143
144 %Dinamica inversa
145
146 matrizA = [1,0,1,0,0,0,0,0,0
147            0,1,0,1,0,0,0,0,0
148            -R12(2),R12(1),-R32(1),R32(1),0,0,0,0,1
149            0,0,-1,0,1,0,0,0,0
150            0,0,0,-1,0,1,0,0,0
151            0,0,R23(2),R23(1),-R43(2),R43(1),0,0,0
152            0,0,0,0,-1,0,1,0,0
153            0,0,0,0,0,-1,0,1,0
154            0,0,0,0,R34(2),-R34(1),-R14(2),R14(1),0];
155
156 matrizC = [m2 * ACG2(1)
157            m2 * ACG2(2)
158            IG2 * alpha2
159            m3*ACG3(1) - FP(1)
160            m3*ACG3(2) - FP(2)
161            IG3*alpha3-RP(1) * FP(2) + RP(2) * FP(1)
162            m4*ACG4(1)
163            m4*ACG4(2)
164            IG4*alpha4-T4];
```

```
165
166 matrizB = matrizA\matrizC;
167
168 F12 = matrizB(1:2);
169 F32 = matrizB(3:4);
170 T12 = matrizB(9);
171
172 clf();
173 %Grafica del mecanismo
174 hold on;
175
176 axis([-15 25 -15 25]);
177 plot([O2(1) RA(1) RB(1) O4(1)],[O2(2) RA(2) RB(2) O4(2)]);
178
179 fill([O2(1) RA(1) RCG2(1)], [O2(2) RA(2) RCG2(2)],"b");
180 fill([RA(1) RB(1) RCG3(1)], [RA(2) RB(2) RCG3(2)],"r");
181 fill([O4(1) RB(1) RCG4(1)], [O4(2) RB(2) RCG4(2)],"g");
182
183 %Grafica de velocidades
184
185 plot([RCG2(1) RCG2(1) + VCG2(1)] , [RCG2(2) RCG2(2) + VCG2(2)],"k");
186 plot([RCG4(1) RCG4(1) + VCG4(1)] , [RCG4(2) RCG4(2) + VCG4(2)],"k");
187 plot([RCG3(1) RCG3(1) + VCG3(1)] , [RCG3(2) RCG3(2) + VCG3(2)],"k");
188 pause(eps);
189 hold off;
190
191
192
193
```

```
%Almacenamiento de datos

%Pos = [Pos; RA RB rS rP rU];
%Vel = [Vel; norm(VA) norm(VB) norm(VS) norm(VP) norm(VU)];
%Acel = [Acel; norm(AA) norm(AB) norm(AS) norm(AP) norm(AU)];

%VelAng = [VelAng; w2 w3 w4];
VelAng(n,1)= w2;
VelAng(n,2)= w3;
VelAng(n,3)= w4;
%AcelAng = [AcelAng; alpha2 alpha3 alpha4];
AcelAng(n,1) = alpha2;
AcelAng(n,2) = alpha3;
AcelAng(n,3) = alpha4;
%Torque = [Torque; T12];
 Torque(n)=T12;
endfor

%Grafico de resultados
figure(2)
plot(t,Torque)
title('Torque vs Tiempo')

%Grafica del mecanismo

%plot(t, PosAng)
%legend PosAng
%title('Posiciond Angular vs Tiempo')

%subplot(4,1,3);
figure(3)
plot(t, VelAng)
legend VelAng
title('Velocidad Angular vs Tiempo')

%subplot(4,1,4);
figure(4)
plot(t, AcelAng)
legend AcelAng
title('Aceleracion Angular vs Tiempo')
```



El plot de la figura 1 se encuentra animado, para ver su animación completa se requiere ejecutar el algoritmo en Octave.

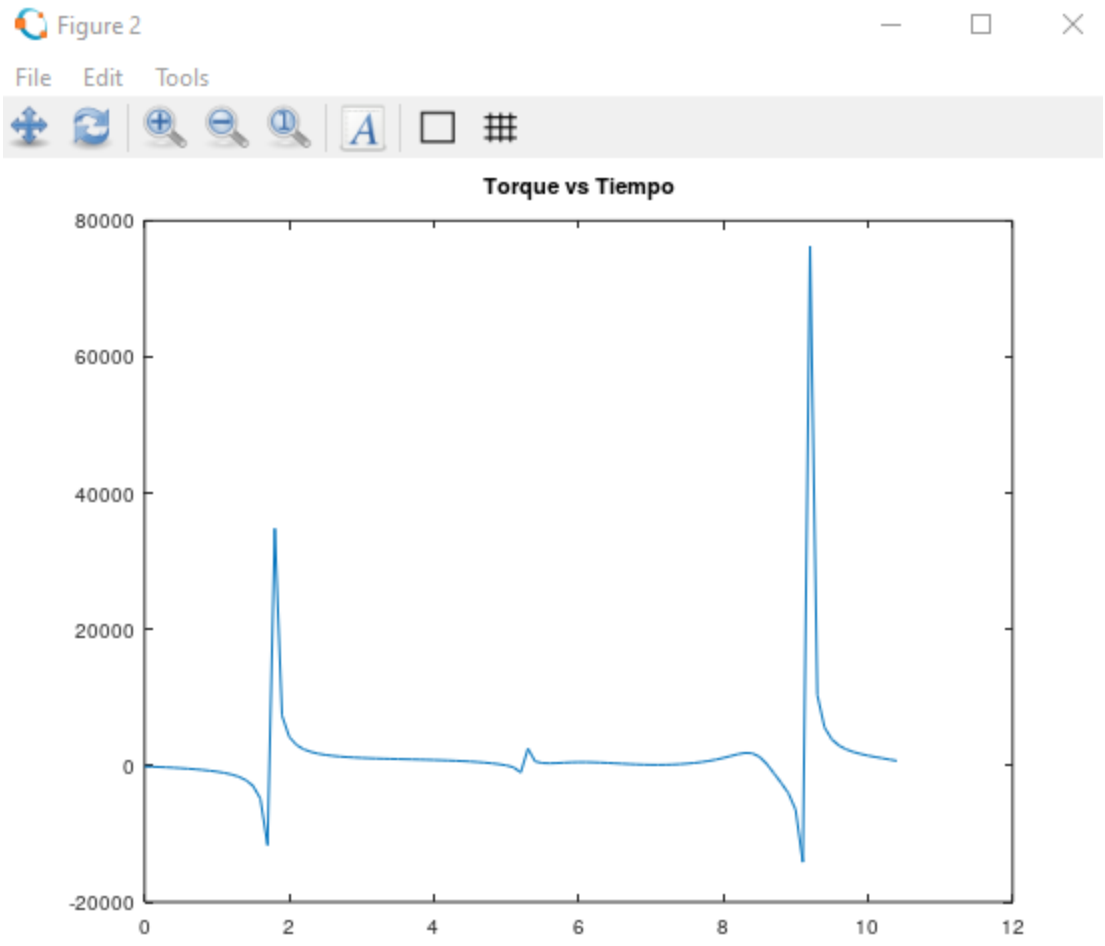


Figure 2
File
Edit
Tools
Torque vs Tiempo
80000
60000
40000
20000
0
-20000
0
2
4
6
8
10
12

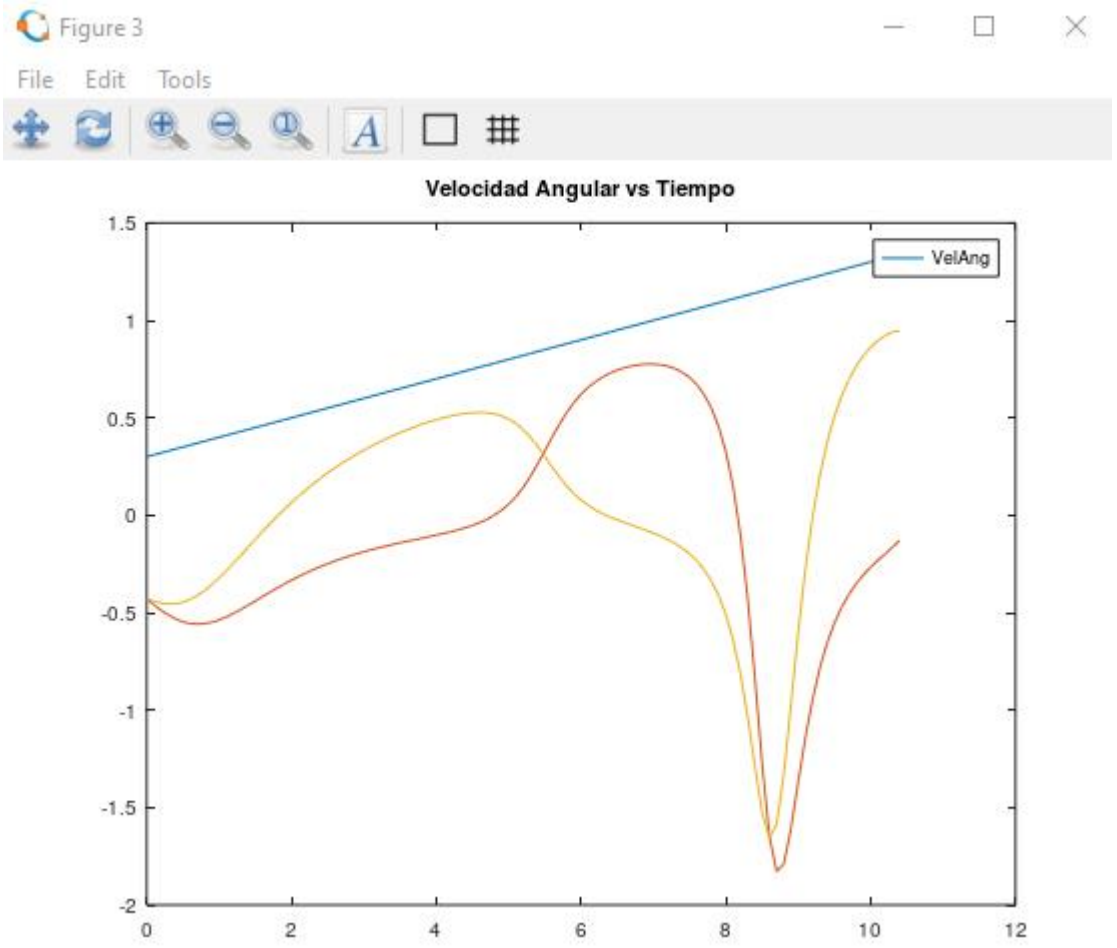


Figure 3
File Edit Tools
Velocidad Angular vs Tiempo
VelAng
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
2
4
6
8
10
12

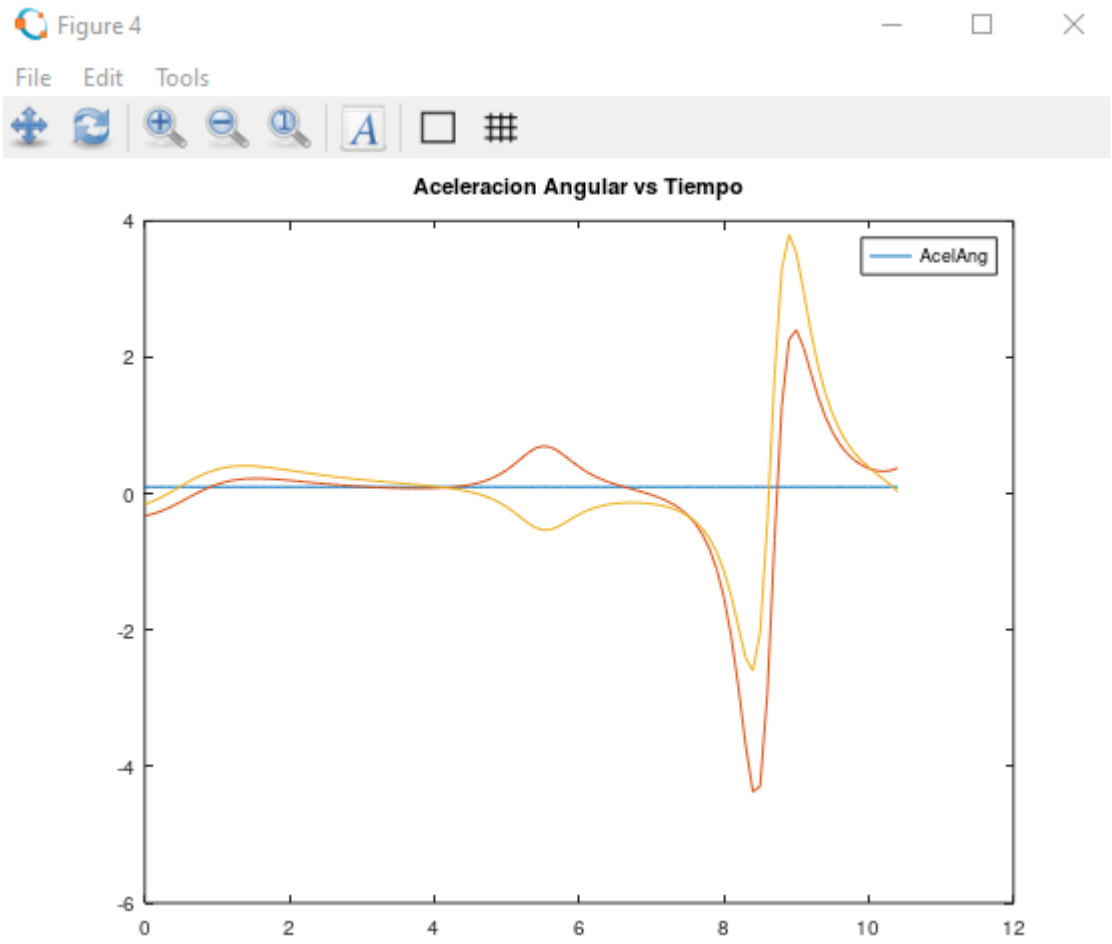


Figure 4
File
Edit
Tools
Aceleracion Angular vs Tiempo
AcelAng
4
2
0
-2
-4
-6
0
2
4
6
8
10
12