

▼ Actividad 2 - Módulo 1

Autor: Walter Allaltune

Link a repositorio GitHub: https://github.com/allaltune/HCSPD_2024/tree/main/Semana%202

Actividad 1

Se solicita obtener el sistema lineal en variables de estado para el equilibrio $x = [0 \ 0 \ 0 \ 0]^T$.

Partiendo de las ecuaciones:

$$\begin{cases} \dot{\alpha} = a(\phi - \alpha) \\ \ddot{\phi} = -\omega^2(\phi - \alpha - b \cdot u) \\ \dot{h} = c \cdot \alpha \end{cases}$$

Considerando: $x_1 = \alpha$, $x_2 = \phi$, $x_3 = \dot{\phi}$ y $x_4 = h$, se tiene que:

$$\dot{x}_t = A \cdot x_t + B \cdot u_t$$

con:

$$A = \begin{bmatrix} -a & a & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \omega^2 & -\omega^2 & 0 & 0 \\ c & 0 & 0 & 0 \end{bmatrix}$$
$$B = \begin{bmatrix} 0 \\ 0 \\ \omega^2 b \\ 0 \end{bmatrix}$$

▼ Actividad 2

Considerando que:

$$\omega = 0.2, a = 0.01, b = 2, c = 100m/s, \Delta t = 10^{-3} \text{ y } t_s = 5s:$$

!apt install octave #Instala octave en el entorno de trabajo

```
%%writefile Avion_Num.m
clc;clear all;
warning('off','all');
X=[0; 0;0;0];t_etapa=10e-3;
tF=5;
color_='r';
color='b';

u=1;
ii=0;
for t=0:t_etapa:tF
    ii=ii+1;
    x2(ii)=X(2);%Ángulo fi
    x4(ii)=X(4);%Altura
    X=modavion(t_etapa, X, u);
    acc(ii)=u;
end

t=0:t_etapa:tF;
hfig1 = figure(1); set(hfig1, 'Visible', 'off');
subplot(3,1,1);hold on;
plot(t,x2,color_);title('x_2 angulo phi'); hold on;
subplot(3,1,2);hold on;
plot(t,x4,color_);title('x_4 Altura');
subplot(3,1,3);hold on;
plot(t,acc,color_);title('Entrada u_t, v_a');
xlabel('Tiempo [Seg.]');

print(hfig1,'VarsAvion_temporal','-r300','-dpng');
save('Verificacion_Num_1Avion.mat','-v7');

Overwriting Avion_Num.m
```

```

%%writefile modavion.m
%Function modelo
function [X]=modavion(t_etapa, xant, accion)
%xant=[alfa,fi,fi_p,h]
a=0.01; b=2; c=100;
omega=0.2;
At=1e-3;
u=accion;

%xant=[alfa,fi,fi_p,h]
alfa=xant(1);
fi=xant(2);
fi_p=xant(3);
h=xant(4);
x=xant;

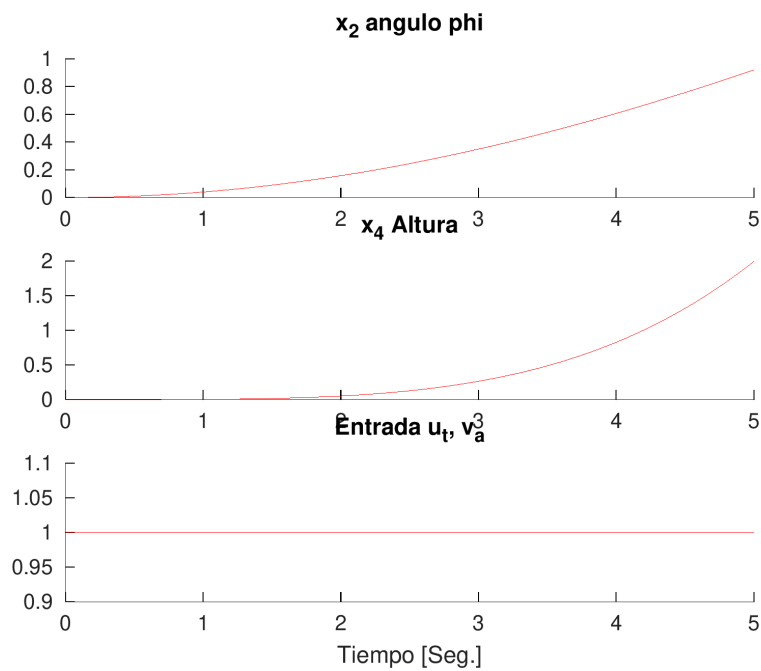
for ii=1:t_etapa/At
    %xp=A*x+B*u;
    alfa_p=a*(fi-alfa);
    fi_pp=(-omega^2)*(fi-alfa-b*u);
    h_p=c*alfa;
    xp=[alfa_p;fi_p;fi_pp;h_p];
    x=x+xp*At;
    fi_p=xp(2);
    alfa=x(1);
    fi=x(2);
    fi_p=x(3);
    h=x(4);
end
X=[x]; %

Overwriting modavion.m

# -W : no window system
loctave -W Avion_Num.m
# import scipy.io
# mat = scipy.io.loadmat('/content/Verificacion_Num_1.mat')

from PIL import Image
imt = Image.open("/content/VarsAvion_temporal.png")
imt

```



Actividad 3

Considerando que:

$\omega = 0.2, a = 0.01, b = 2, c = 50/s, \Delta t = 10^{-3}$ y $t_s = 20s$:

```
%%writefile Avion_Num2.m
clc;clear all;
warning('off','all');
X=[0; 0;0;0];t_etapa=10e-3;
tF=20;
color_='r';
color='b';

u=1;
ii=0;
for t=0:t_etapa:tF
    ii=ii+1;
    x2(ii)=X(2);%Ángulo fi
    x4(ii)=X(4);%Altura
    X=modavion(t_etapa, X, u);
    acc(ii)=u;
end

t=0:t_etapa:tF;
hfig1 = figure(1); set(hfig1, 'Visible', 'off');
subplot(3,1,1);hold on;
plot(t,x2,color_);title('x_2 angulo phi'); hold on;
subplot(3,1,2);hold on;
plot(t,x4,color_);title('x_4 Altura');
subplot(3,1,3);hold on;
plot(t,acc,color_);title('Entrada u_t, v_a');
xlabel('Tiempo [Seg.]');

print(hfig1,'VarsAvion_temporal2','-r300','-dpng');
save('Verificacion_Num_1Avion2.mat','-v7');

Overwriting Avion_Num2.m

%%writefile modavion.m
%Funcion modelo
function [X]=modavion(t_etapa, xant, accion)
%xant=[alfa,fi,fi_p,h]
a=0.01; b=2; c=50;
omega=0.2;
At=1e-3;
u=accion;

%xant=[alfa,fi,fi_p,h]
alfa=xant(1);
fi=xant(2);
fi_p=xant(3);
h=xant(4);
x=xant;

for ii=1:t_etapa/At
    %xp=A*x+B*u;
    alfa_p=a*(fi-alfa);
    fi_pp=(-omega^2)*(fi-alfa-b*u);
    h_p=c*alfa;
    xp=[alfa_p;fi_p;fi_pp;h_p];
    x=x+xp*At;
    fi_p=xp(2);
    alfa=x(1);
    fi=x(2);
    fi_p=x(3);
    h=x(4);
end
X=[x]; %

Overwriting modavion.m

# -W : no window system
!octave -W Avion_Num2.m
# import scipy.io
# mat = scipy.io.loadmat('/content/Verificacion_Num_1Avion2.mat')

from PIL import Image
imt = Image.open("/content/VarsAvion_temporal2.png")
imt
```

