Linearizzazione

lunedì 31 gennaio 2022 15:54

$$m\ddot{p} = -\beta_1 \dot{p} + m(k-1)\left(\frac{k_G M}{p^2} - p\,\omega^2\right)$$

$$\dot{\omega} = -\frac{2\omega\dot{p}}{p} - \frac{\beta_2 \omega}{m} + \frac{\tau}{mp}$$

$x_1 = p$

$x_2 = \dot{p}$

$u = \tau$

$y = \omega$

Forma di stato

$x_1 = p \quad x_2 = \dot{p} \quad x_3 = \omega \quad u = \tau \quad y = \omega$

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -\frac{\beta_1 x_2}{m} + (k-1)\left(\frac{k_G M}{x_1^2} - x_1 x_3^2\right) \\ \dot{x}_3 = -\frac{2 x_3 x_2}{x_1} - \frac{\beta_2 x_3}{m} + \frac{u}{m x_1} \\ y = x_3 \end{cases}$$

SISTEMA IN FORMA DI STATO

$$\boxed{\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} x_2 \\ -\frac{\beta_1 x_2}{m} + (k-1)\left(\frac{k_G M}{x_1^2} - x_1 x_3^2\right) \\ -\frac{2 x_3 x_2}{x_1} - \frac{\beta_2 x_3}{m} + \frac{u}{m x_1} \end{bmatrix} \qquad y = x_3}$$

LINEARIZZAZIONE

```matlab
close all; clear all; clc;
%% Definizioni
syms x1 x2 x3 u b1 b2 m k M autovettori autovalori T_inv
A_hat B_hat C_hat D_hat modello
b1 = 0.3;
b2 = 0.1;
m=1;
k=1.5;
kG=6.67e-11;
M = 5.98e24;
% intervallo di tempo
interv = 0:0.1:100; % da 0 a 10 secondi con passo 0.1
% sistema
f1=x2;
f2=-b1*x2/m + (k-1)*(kG*M/x1^2 -x1*x3^2);
f3=-2*x3*x2/x1 - b2*x3/m + u/(m*x1);
y=x3;
%% Punti di equilibrio
x1_e=3e7;
x2_e=0;
x3_e=0.000121543;
u_e=364.63;
xe=[x1_e,x2_e,x3_e,u_e];
%% Jacobiana e linearizzazione
A = jacobian([f1,f2,f3],[x1,x2,x3])
B = jacobian([f1,f2,f3],u)
C = jacobian(y,[x1,x2,x3])
D = jacobian(y,u)

[T_inv,A_hat]=jordan(A);
A_hat=double(subs(A_hat,[x1,x2,x3,u],xe))
B_hat = double(subs(inv(T_inv)*B,[x1,x2,x3,u],xe))
C_hat = double(subs(C*T_inv,[x1,x2,x3,u],xe))
D_hat = double(subs(D,[x1,x2,x3,u],xe))
%% Trovo sistema lineare
x0=[x1_e;x2_e;x3_e];
modello = ss(A_hat, B_hat, C_hat, D_hat);
x0_J = inv(T_inv)*x0   % stato iniziale nelle nuove
coordinate
```

Punti di equilibrio

$$x_e = \begin{bmatrix} 3 \cdot 10^7 \\ 0 \\ 1{,}21 \cdot 10^{-4} \end{bmatrix} \qquad U_e = 364{,}43$$

```
x0_J =

  1.0e-03 *

  -0.1215 - 0.0000i
   0.2431 + 0.0000i
   0.0000 - 0.0000i
```

```
A =

[                                   0,     1,                0]
[- x3^2/2 - 6381856000000001/(16*x1^3),   -3/10,           -x1*x3]
[          (2*x2*x3)/x1^2 - u/x1^2, -(2*x3)/x1, - (2*x2)/x1 - 1/10]

B =

   0
   0
1/x1

C =

[0, 0, 1]

D =

0

A_hat =

  -0.0000 - 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i
   0.0000 + 0.0000i  -0.1000 + 0.0000i   0.0000 + 0.0000i
```

$\dot{x} = 0$

```
x0=[x1_e;x2_e;x3_e];
modello = ss(A_hat, B_hat, C_hat, D_hat);
x0_J = inv(T_inv)*x0   % stato iniziale nelle nuove coordinate
x0_J= double(subs(x0_J,[x1,x2,x3,u],xe))
uu = zeros(length(interv), 1); % input nullo (evoluzione libera)
[YY_J, TT_J, XX_J] = lsim(modello, uu, interv, x0_J);
YY_J = real(YY_J); % convertiamo il tipo di dato in 'reale' (la parte immaginaria è sicuramente nulla)
%% Grafico
figure;
plot(TT_J,YY_J)
hold on; grid on; zoom on; box on;
title('Traiettoria sistema')
xlim([0, 100])
xlabel('tempo [s]')
ylabel('posizione')
```

```
A_hat =

  -0.0000 - 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i
   0.0000 + 0.0000i  -0.1000 + 0.0000i   0.0000 + 0.0000i
   0.0000 + 0.0000i   0.0000 + 0.0000i  -0.3000 - 0.0000i

B_hat =

   1.0e-07 *

   0.0000 - 0.0000i
   0.3333 - 0.0000i
   0.0000 + 0.0000i

C_hat =

     1     1     1

D_hat =

     0
```

$$\begin{cases} \dot{x}_1 = 0 \\ \dot{x}_2 = -0{,}1\, x_2 + 0{,}33 \cdot 10^{-7} u \\ \dot{x}_3 = -0{,}3\, x_3 \\ \\ y = x_1 + x_2 + x_3 \end{cases}$$