Simulazione di sistemi non lineari con Matlab & Simulink

Gianmaria De Tommasi[1]

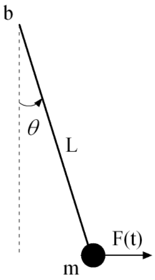
[1]Università degli Studi di Napoli Federico II
detommas@unina.it

Ottobre 2012

Corsi AnsaldoBreda

Outline

Il pendolo



Pendolo di massa *m*, lunghezza *L* e con coefficiente di attrito rotazionale *b*.

Il modello nello spazio di stato

Se

$$x(t) = \begin{pmatrix} \theta(t) \\ \dot{\theta}(t) \end{pmatrix} \qquad u(t) = F(t) \qquad y(t) = \theta(t)$$

Il modello nello spazio di stato

Se

$$x(t) = \begin{pmatrix} \theta(t) \\ \dot{\theta}(t) \end{pmatrix} \qquad u(t) = F(t) \qquad y(t) = \theta(t)$$

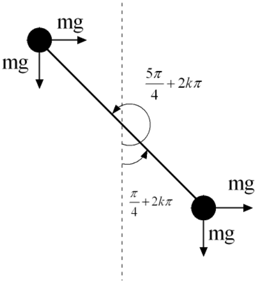
Allora

$$\begin{aligned} \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= -\frac{g}{L}\sin x_1(t) - \frac{b}{mL^2}x_2(t) + \frac{1}{mL}\cos x_1(t)u(t) \\ y(t) &= x_1(t) \end{aligned} \tag{1}$$

Punti d'equilibrio

Se $\bar{u} = mg$, ponendo $f(\bar{x}, \bar{u}) = 0$ si ottengono gli stati di equilibrio

$$\bar{x}_k = \begin{pmatrix} \frac{\pi}{4} + k\pi \\ 0 \end{pmatrix}$$



Il modello linearizzato

Consideriamo lo stato d'equilibrio $\bar{x} = \begin{pmatrix} \frac{\pi}{4} & 0 \end{pmatrix}^{\mathrm{T}}$. Il modello linearizzato sarà

$$\begin{aligned}
\delta\dot{x}_1(t) &= \delta x_2(t) \\
\delta\dot{x}_2(t) &= -\frac{\sqrt{2}g}{L}\delta x_1(t) - \frac{b}{mL^2}\delta x_2(t) + \frac{1}{\sqrt{2}mL}\delta u(t) \\
\delta y(t) &= \delta x_1(t)
\end{aligned}$$

La funzione `pendolo.m` - 1

```
% Parametri del sistema
m = 10;
b = 15;
g = 9.81;
L = 1;
% Stato di equilibrio e corrispondente ingresso
xbar = [pi/4 0];
ubar = m*g;
% Uscita di equilibrio
ybar = xbar(1);
% Variazione delle condizioni iniziali
dx0 = [0.3 3];
% Variazione dell'ingresso, del tipo:  Am*sin(t)
Am = 10;
% Intervallo di simulazione
tfin = 15;
```

La funzione pendolo.m - 2

```
% Matrici del modello linearizzato
A = [ 0 1 ; -sqrt(2)*g/L -b/(m*L^2) ];
B = [ 0 ; 1/(sqrt(2)*m*L) ];
C = [ 1 0 ];
D = 0;
% Costruzione del modello linearizzato
sys_l = ss(A,B,C,D);
% Intervallo di integrazione per la simulazione lineare
tlin = 0:.001:tfin;
% Simulazione lineare
du = Am*sin(tlin);
ylin= lsim(sys_l,du,tlin,dx0);
```

La funzione `pendolo.m` - 3

```
% Simulazione non lineare
[ t,x_nl ]=ode45('p_nl',[0 tfin],xbar+dx0,[],m,b,g,L,ubar,Am);
y_nl = x_nl(:,1);
% Grafico
figure(1)
plot(t,y_nl*180/pi,'-',tlin,(ylin+ybar)*180/pi,'-')
grid on
ylabel('[deg]')
xlabel('tempo [s]')
title('theta')
legend('NL','L')
```

La funzione `p_nl.m`

```
function xdot = p_nl(t,x,flag,m,b,g,L,ubar,Am)
%
% Restituisce il valore della derivata dello stato
% per il pendolo non lineare
%
u = ubar + Am*sin(t);
xdot = [x(2); -g/L*sin(x(1)) - b/(m*L^2)*x(2) +...
u/(m*L)*cos(x(1))];
```