

## 2. Casos de estudio

### 2.1. Modelado en variables de estado de un sistema RLC

Con el objeto de asociar estas definiciones a la modelación de un sistema físico, se toma como ejemplo un circuito elemental RLC; representado en la Fig. 2-1.

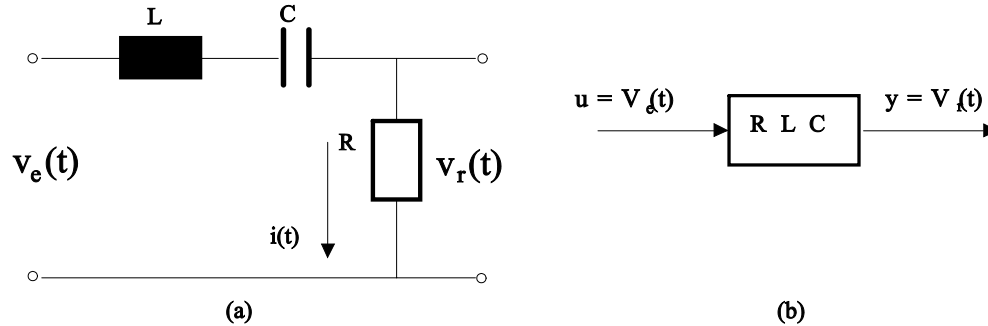


Fig. 2-1. (a) Esquemático del circuito RLC; (b) Modelo entrada-Salida del circuito RLC

Se toma  $u=v_e(t)$  como señal de entrada al sistema y la tensión  $v_r(t)$  sobre el resistor  $R$  como salida.

Por relaciones físicas es conocido que la evolución de las distintas variables físicas en este circuito, tales como tensiones y corrientes, quedará definida para todo  $t \geq t_0$  si se conoce para un instante de tiempo  $t=t_0$ , la corriente que fluye en el inductor  $L$ , la tensión que exista sobre el capacitor  $C$  y la tensión de entrada desde  $t_0$  en adelante.

En base a la definición que se ha dado de variables de estado es posible elegir a la *corriente* en el circuito y a la *tensión* sobre el capacitor como variables de estado, ya que éstas definen el estado dinámico del circuito. La evolución del estado dinámico para  $t \geq t_0$  se podrá determinar si se conoce para  $t=t_0$  las variables de estado  $i(t_0)$ ,  $v_c(t_0)$  y además la tensión de entrada  $v_e(t)$  para  $t \geq t_0$ .

Para analizar la evolución del circuito se pueden plantear las ecuaciones diferenciales del mismo como,

$$\begin{cases} \frac{di}{dt} = -\frac{R}{L} i - \frac{1}{L} v_c + \frac{1}{L} v_e \\ \frac{dv_c}{dt} = \frac{1}{C} i \end{cases} \quad (2-1)$$

Las Ec. (2-1) se pueden expresar en una ecuación matricial-vectorial.

$$\begin{bmatrix} \frac{di}{dt} \\ \frac{dv_c}{dt} \end{bmatrix} = \begin{bmatrix} -\frac{R}{L} & -\frac{1}{L} \\ \frac{1}{C} & 0 \end{bmatrix} \begin{bmatrix} i \\ v_c \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} [v_e] \quad (2-2)$$

Definiendo a  $i$ ,  $v_c$  como variables de estado y a  $\mathbf{x}$  como vector de estado, la Ec. (2-2) se convierte en

$$\dot{\mathbf{x}} = \mathbf{A} \mathbf{x}(t) + \mathbf{b} u(t) \quad (2-3)$$

con

$$A = \begin{bmatrix} -R/L & -1/L \\ 1/C & 0 \end{bmatrix}, b = \begin{bmatrix} 1/L \\ 0 \end{bmatrix}, \quad (2-4)$$

siendo **A** la matriz del sistema y **b** el vector de entrada. La variable de salida  $y=v_R$  puede obtenerse a partir del vector de estado mediante

$$y = c^T x(t) \quad (2-5)$$

con el vector de salida **c** definido como

$$c^T = [R \quad 0] \quad (2-6)$$

De esta forma el circuito RLC de la Fig. 2-1 queda modelado en el espacio de estado por

$$\begin{aligned} \dot{x}(t) &= Ax(t) + b u(t) \\ y(t) &= c^T x(t) \end{aligned} \quad (2-7)$$

con  $x(t) = [i \quad v_c]^T$ ,  $u(t) = v_e(t)$ ,  $y(t) = v_r(t)$  siendo **A**, **b**, **c** definidas por las ecuaciones (2-4) y (2-6).

### Ejercicio sugerido

Asignar valores a  $R=1K\Omega$ ,  $L=10\mu H$ , y  $C=1nF$ . Obtener simulaciones que permitan estudiar la dinámica del sistema, con una entrada de tensión escalón de 12V. Establecer condiciones iniciales nulas y no nulas.

## 2.2. Ejemplos de linealización

En la Fig. 2-2 y Fig. 2-3 se muestran dos esquemas simplificados de sistemas a modelar mediante una expresión lineal y se requiere obtener el modelo en variables de estado de cada uno. Mediante el código para Octave disponible en la Tabla 2-1 para linealizar y simular los sistemas no lineales, obtener una simulación de la evolución temporal y compararla con la de sus respectivos sistemas no lineales.

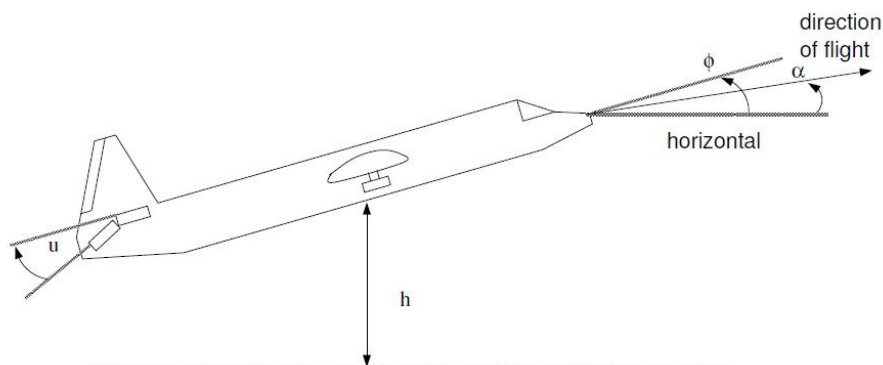


Fig. 2-2. Modelo de sistema de altitud en un avión, extraído de [1].

Para el caso de la Fig. 2-2, modelo válido sólo para pequeños ángulos, se tiene

$$\begin{cases} \dot{\alpha} = a(\phi - \alpha) \\ \ddot{\phi} = -\omega^2(\phi - \alpha - b \cdot u) \\ \dot{h} = c\alpha \end{cases} \quad (2-8)$$

donde  $\omega > 0$  representa la frecuencia natural, y los coeficientes  $a$   $b$  son constantes positivas,  $u$  es la variable manipulada y es proporcional a la posición de los elevadores,  $\phi$  (ángulo de cabeceo) en radianes, vuela a  $c$  metros por segundo, su trayectoria de vuelo forma un ángulo  $\alpha$  con la horizontal (si  $\alpha > 0$  sube, si  $\alpha < 0$  desciende) Elegir  $x_1 = \alpha$ ,  $x_2 = \phi$ ,  $x_3 = \dot{\phi}$  y  $x_4 = h$ . Se pide,

- 1- Obtener el sistema lineal en variables de estado para el equilibrio  $x = [0 \ 0 \ 0 \ 0]^T$ ,  $u=0$ .
- 2- Obtener la solución numérica del sistema lineal para evaluar cuantitativamente el comportamiento con intención de verificar el correcto planteo. Para hacerlo, se le asignan los valores siguientes a los parámetros, son  $\omega=2$ ;  $a=0,05$ ;  $b=5$ ;  $c=100$  m/s, (es decir, 360Km/h),  $\Delta t=10^{-3}$ ; y el tiempo de simulación de 5 segundos.
- 3- Obtener la solución numérica del sistema lineal para  $c=50$  m/s, (es decir, 180Km/h),  $\Delta t=10^{-3}$ ; y el tiempo de simulación de 20 segundos.

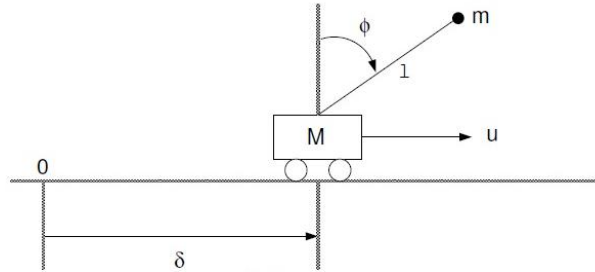


Fig. 2-3. Sistemas para modelar, extraído de [1].

Para el caso del esquema del péndulo invertido de la Fig. 2-3 se tiene,

$$\begin{cases} (M+m)\ddot{\delta} + m l \ddot{\phi} \cos \phi - m l \dot{\phi}^2 \sin \phi + F \dot{\delta} = u \\ l \ddot{\phi} - g \sin \phi + \ddot{\delta} \cos \phi = 0 \end{cases} \quad (2-9)$$

donde el sistema lineal en variables de estado tiene el vector de estados definido como  $x = [\delta \ \dot{\delta} \ \phi \ \dot{\phi}]^T$ , y que las matrices  $A$  y  $B$  para el equilibrio inestable  $x = [0 \ 0 \ 0 \ 0]^T$ ,  $u=0$ , para ángulo pequeño es

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{F}{M} & -\frac{mg}{M} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{F}{lM} & \frac{g(m+M)}{lM} & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ -\frac{1}{lM} \end{bmatrix} \quad (2-10)$$

es decir

$$\begin{bmatrix} \dot{\delta} \\ \ddot{\delta} \\ \dot{\phi} \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{F}{M} & -\frac{mg}{M} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{F}{IM} & \frac{g(m+M)}{IM} & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ -\frac{1}{IM} \end{bmatrix} \cdot u \quad (2-11)$$

Se pide:

- 1- Obtener la solución numérica de los dos sistemas (2-9) y (2-11), del lineal y del no lineal para evaluar cuantitativamente la equivalencia.
- 2- Obtener el sistema lineal para el equilibrio estable  $x = [0 \ 0 \ \pi \ 0]^T$ .
- 3- Obtener la solución numérica de los dos sistemas, del lineal y del no lineal para evaluar cuantitativamente la equivalencia. Emplear los valores de las variables  $m=0,1; F=0,1; l=0,6; g=9,8; M=0,5$  y  $\Delta t=0,0001$  seg. de tal manera que el sistema evolucione durante cinco segundos. Si el sistema es inestable, parta de una condición inicial cercana al equilibrio y luego simule los 5seg sin excitación externa.

La Tabla 2-1 muestra el código para Octave que implementa la linealización en el equilibrio inestable para el péndulo invertido, y la Tabla 2-2 detalla el código para obtener lo propio para el equilibrio estable.

```
clear all; clc;
pkg load symbolic
syms fi fi_p fi_pp p p_p p_pp M m u long Fricc g;
disp('Para el equilibrio inestable')
ang_inic=0;
p_pp=(1/(M+m))*(u-m*long*fi_pp+m*long*fi_p^2*fi-Fricc*p_p); %Pequeños angulos
% fi_pp=(1/long)*(g*sin(fi)-p_pp*cos(fi)); %Expresión completa
% fi_pp=(1/long)*(g*(fi)-p_pp); %Pequeños angulos para fi~0, sin(fi)~fi, cos(fi)~1
fi_pp=solve(fi_pp==(1/long)*(g*fi-p_pp), fi_pp);
%disp('fi_pp='); pretty(simplify(fi_pp));
p_pp=subs(p_pp, 'fi_pp', fi_pp);
%disp('p_pp='); pretty(simplify(p_pp))
Mat_A=[0 1 0 0];
[subs(subs(subs(subs(diff(p_pp, p), p,0), p_p,0), fi, ang_inic), fi_p,0), ...
subs(subs(subs(subs(diff(p_pp, p_p), p,0), p_p,0), fi, ang_inic), fi_p,0), ...
subs(subs(subs(subs(diff(p_pp, fi), p,0), p_p,0), fi, ang_inic), fi_p,0), ...
subs(subs(subs(subs(diff(p_pp, fi_p), p,0), p_p,0), fi, ang_inic), fi_p,0)]];
[0 0 0 1];
[subs(subs(subs(subs(diff(fi_pp, p), p,0), p_p,0), fi, ang_inic), fi_p,0), ...
subs(subs(subs(subs(diff(fi_pp, p_p), p,0), p_p,0), fi, ang_inic), fi_p,0), ...
subs(subs(subs(subs(diff(fi_pp, fi), p,0), p_p,0), fi, ang_inic), fi_p,0), ...
subs(subs(subs(subs(diff(fi_pp, fi_p), p,0), p_p,0), fi, ang_inic), fi_p,0)]];
Mat_B=[0;
subs(subs(subs(subs(diff(p_pp, u), p,0), p_p,0), fi, ang_inic), fi_p,0); ...
0;
subs(subs(subs(subs(diff(fi_pp, u), p,0), p_p,0), fi, ang_inic), fi_p,0)];
pretty(simplify(Mat_A))
pretty(simplify(Mat_B))
```

**Tabla 2-1. Algoritmo simbólico para la linealización del péndulo en un punto de equilibrio inestable.**

```
clear all; clc;
pkg load symbolic
syms fi fi_p fi_pp p p_p p_pp M m u long Fricc g;
disp('Para el equilibrio estable')
ang_inic=pi;
p_pp=(1/(M+m))*(u+m*long*fi_pp+m*long*fi_p^2*(pi-fi)-Fricc*p_p);
```

```

%fi_pp=(1/long)*(-g*(fi)+p_pp); %Pequeños ángulos para fi~pi sin(fi)~(pi-fi),
cos(fi)=-1
fi_pp=solve(fi_pp==(1/long)*(g*(pi-fi)+p_pp),fi_pp);
disp('fi_pp='); pretty(simplify(fi_pp));
p_pp=subs(p_pp,'fi_pp',fi_pp);
disp('p_pp='); pretty(simplify(fi_pp));
Mat_A=[0 1 0 0];
[subs(subs(subs(subs(diff(p_pp, p), p,0),p_p,0),fi,ang_inic),fi_p,0), ...
subs(subs(subs(subs(diff(p_pp, p_p), p,0),p_p,0),fi,ang_inic),fi_p,0), ...
subs(subs(subs(subs(diff(p_pp, fi), p,0),p_p,0),fi,ang_inic),fi_p,0), ...
subs(subs(subs(subs(diff(p_pp, fi_p), p,0),p_p,0),fi,ang_inic),fi_p,0)]; ...
[0 0 0 1];...
[subs(subs(subs(subs(diff(fi_pp, p), p,0),p_p,0),fi,ang_inic),fi_p,0),...
subs(subs(subs(subs(diff(fi_pp, p_p), p,0),p_p,0),fi,ang_inic),fi_p,0),...
subs(subs(subs(subs(diff(fi_pp, fi), p,0),p_p,0),fi,ang_inic),fi_p,0),...
subs(subs(subs(subs(diff(fi_pp, fi_p), p,0),p_p,0),fi,ang_inic),fi_p,0)]];
Mat_B=[0;
subs(subs(subs(subs(diff(p_pp, u), p,0),p_p,0),fi,ang_inic),fi_p,0);...
0;
subs(subs(subs(subs(diff(fi_pp, u), p,0),p_p,0),fi,ang_inic),fi_p,0)];
pretty(simplify(Mat_A))
pretty(simplify(Mat_B))

```

Tabla 2-2. Algoritmo simbólico para la linealización del péndulo en un punto de equilibrio estable.

The screenshot shows the Octave environment with the following content in the Command Window:

```

OctSymPy v2.6.0: this is free software without warranty, see source.
Initializing communication with SymPy using a popen2() pipe.
Some output from the Python subprocess (pid 6088) might appear next.
Python 2.7.9 (default, Dec 10 2014, 12:24:55) [MSC v.1500 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
(InteractiveConsole)
>>> >>>
OctSymPy: Communication established. SymPy v1.1.1.
Python 2.7.9 (default, Dec 10 2014, 12:24:55) [MSC v.1500 32 bit (Intel)]
Para el equilibrio inestable
[0 1 0 0]
[ -Fricc -g*m ]
[0 0 0 1]
[ Fricc g*(M+m) ]
[0 0 0 0]
[ M*long M*long ]
[ 0 ]
[ 1 ]
[ - ]
[ M ]
[ 0 ]
[ -1 ]
[ - ]
[ M*long ]
Para el equilibrio estable
fi_pp=
-Fricc*p_p - M*fi*g - fi*fi_p*long*m - fi*g*m + u
[0 1 0 0]
[ -Fricc -g*m ]
[0 0 0 1]
[ M M ]

```

Fig. 2-4. Salida del entorno Octave. Matrices A y B del equilibrio inestable completo.

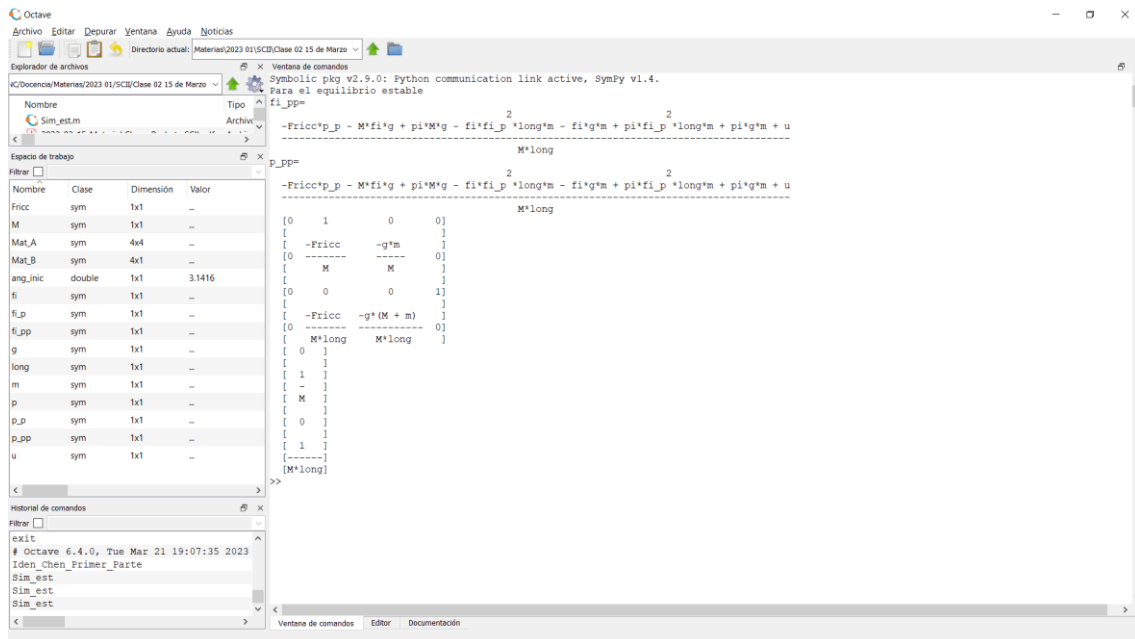


Fig. 2-5. Salida del entorno Octave. Matrices A y B del equilibrio estable completo.

### Verificación del sistema linealizado

Para verificar que los resultados obtenidos son válidos, se debe realizar una simulación. Para ello, se procede a simular el comportamiento del péndulo invertido en su equilibrio estable. Nótese que al sistema lineal se le debe restar el valor de operación del péndulo que sería igual al anterior pero su ángulo  $\phi_{OP}$  es  $\pi$ , y su acción de control  $u_{OP}$  es nula.

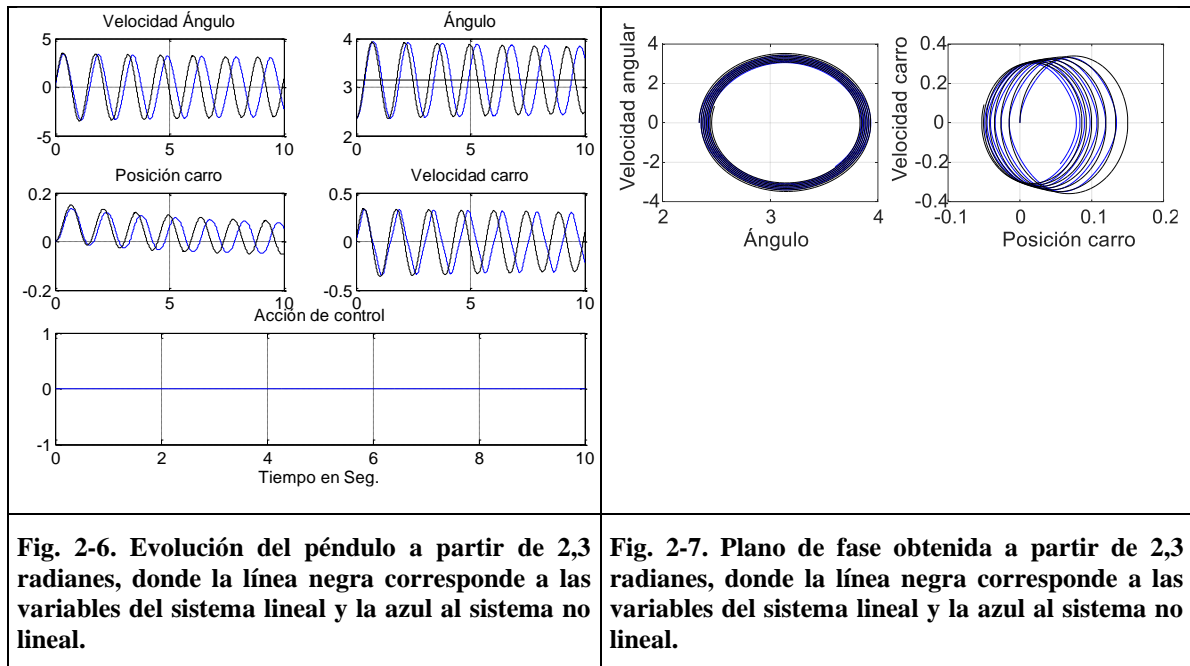
```
clc;clear all;
m=.1;Fricc=0.1; long=0.6;g=9.8;M=.5;
h=0.0001;tiempo=(10/h);p_pp=0;tita_pp=0; t=0:h:tiempo*h;
omega(1)=0;p_p=0:h:tiempo*h; u=linspace(0,0,tiempo+1);
%Condiciones iniciales
alfa(1)=pi-0.8; color='b';
p(1)=0; p_p(1)=0; u(1)=0; p(1)=0; i=1;
%Versión linealizada en el equilibrio estable. Sontag Pp 104.
estado=[p(i); p_p(i); alfa(i); omega(i)]
Mat_A=[0 1 0 0;0 -Fricc/M -m*g/M 0; 0 0 0 1; 0 -Fricc/(long*M) -g*(m+M)/(long*M) 0]
Mat_B=[0; 1/M; 0; 1/(long*M)]
X0=[0 0 pi 0]';x=[0 0 alfa(1) 0]';
while(i<(tiempo+1))
    %Variables del sistema no lineal
    estado=[p(i); p_p(i); alfa(i); omega(i)];
    u(i)=0;
    %Sistema no lineal
    p_pp=(1/(M+m))*(u(i)-m*long*tita_pp*cos(alfa(i))+m*long*omega(i)^2*sin(alfa(i))-
    Fricc*p_p(i));
    tita_pp=(1/long)*(g*sin(alfa(i))-p_pp*cos(alfa(i)));
    p_p(i+1)=p_p(i)+h*p_pp;
    p(i+1)=p(i)+h*p_p(i);
    omega(i+1)=omega(i)+h*tita_pp;
    alfa(i+1)=alfa(i)+h*omega(i);
    %Variables del sistema lineal
    pl(i)=x(1); p_pl(i)=x(2); alfa(i)=x(3); omegal(i)=x(4);
    %Sistema lineal
    xp=Mat_A*(x-X0)+Mat_B*u(i);
    x=x+h*xp;
    i=i+1;
end
pl(i)=x(1); p_pl(i)=x(2); alfa(i)=x(3); omegal(i)=x(4);
figure(1);hold on;
subplot(3,2,1);plot(t,omega,color);grid on; title('Velocidad Ángulo');hold
on;plot(t,omegal,'k');
subplot(3,2,2);plot(t,alfa,color);hold
on;plot(t,pi*ones(size(t)),'k');plot(t,alfal,'k');
```

```

grid on;title('Ángulo');hold on;
subplot(3,2,3); plot(t,p,color);grid on;title('Posición carro');hold
on;plot(t,p_l,'k');
subplot(3,2,4);plot(t,p_p,color);grid on;title('Velocidad carro');hold
on;plot(t,p_pl,'k');
subplot(3,1,3);plot(t,u,color);grid on;title('Acción de control');xlabel('Tiempo en
Seg. ');hold on;
figure(2);hold on;
subplot(2,2,1);plot(alfa,omega,color);grid on;xlabel('Ángulo');ylabel('Velocidad
angular');hold on;
subplot(2,2,1);plot(alfal,omegal,'k');
subplot(2,2,2);plot(p,p_p,color);grid on;xlabel('Posición carro');ylabel('Velocidad
carro');hold on;
subplot(2,2,2);plot(pl,p_pl,'k');

```

**Tabla 2-3. Algoritmo que compara el desempeño del sistema lineal y del no lineal en el punto de equilibrio estable.**



Nótese que los sistemas muestran un desempeño similar con una diferencia que es acumulativa en el tiempo.

### 2.3. Modelado a partir de las señales de entrada y salida

Es frecuente que se desea controlar a un proceso que está funcionando operativamente, y el objetivo de control es mantener los valores de la salida con mayor exactitud y menores costos relacionados a las acciones de control. En esos casos, es probable que se disponga de datos de entrada y salida del proceso. Una forma de emplear esos datos y obtener una versión lineal de primer orden, es determinar la respuesta al escalón [2] [3]. Debe tenerse en cuenta que hay que excitar al sistema con un escalón, y debe medirse la salida para establecer relaciones en función de su respuesta temporal que permitan obtener los parámetros.

#### Sistemas de primer orden

En la Fig. 2-8 se muestra la respuesta al escalón de un sistema de primer orden que tiene  $T_1=0,5\text{seg}$ , de ganancia  $g_1$  dividido la amplitud del escalón de entrada.

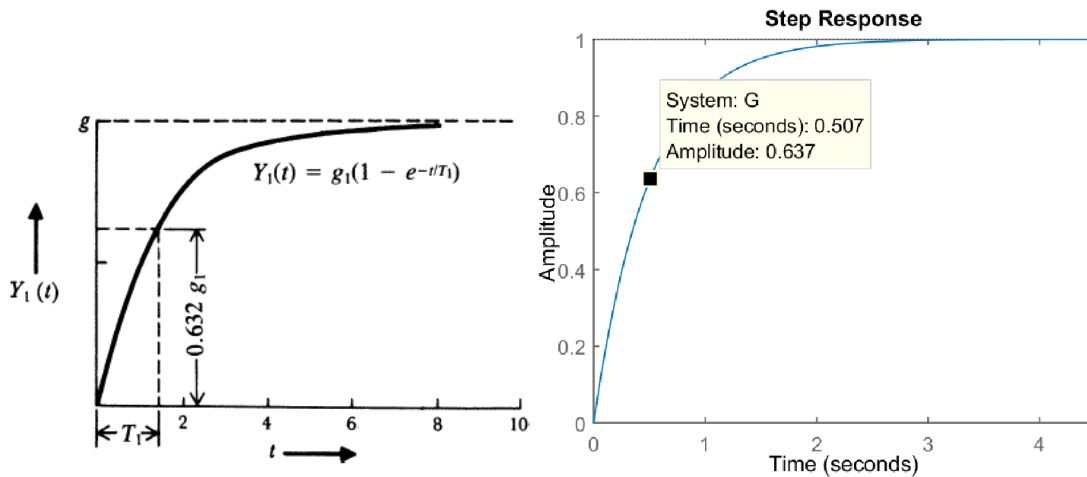


Fig. 2-8. Respuesta temporal donde se destaca derivada muy grande al inicio.

Sabiendo que la función de transferencia es

$$G_1(s) = \frac{Y(s)}{U(s)} = K \frac{1}{T_1 s + 1} \quad (2-12)$$

y si la entrada es un escalón unitario, la salida será

$$y_t = K \left( 1 - e^{-\frac{t}{T_1}} \right) \quad (2-13)$$

donde la lectura de la respuesta Fig. 2-8 y un escalón unitario permiten encontrar la constante de tiempo  $T_1$ , ya que en el estado estacionario se puede obtener la ganancia  $K$  para el escalón unitario. Dicho escalón puede no ser unitario, y debe tenerse en cuenta esa escala. Aquí sólo es necesario conocer las relaciones que existe entre la respuesta temporal de un sistema y su respuesta en frecuencia.

Cuando existe un retardo en el sistema, la respuesta al escalón unitario tendrá una forma como la de la Fig. 2-9, donde se ve la comparación con la versión aproximada de Padé.

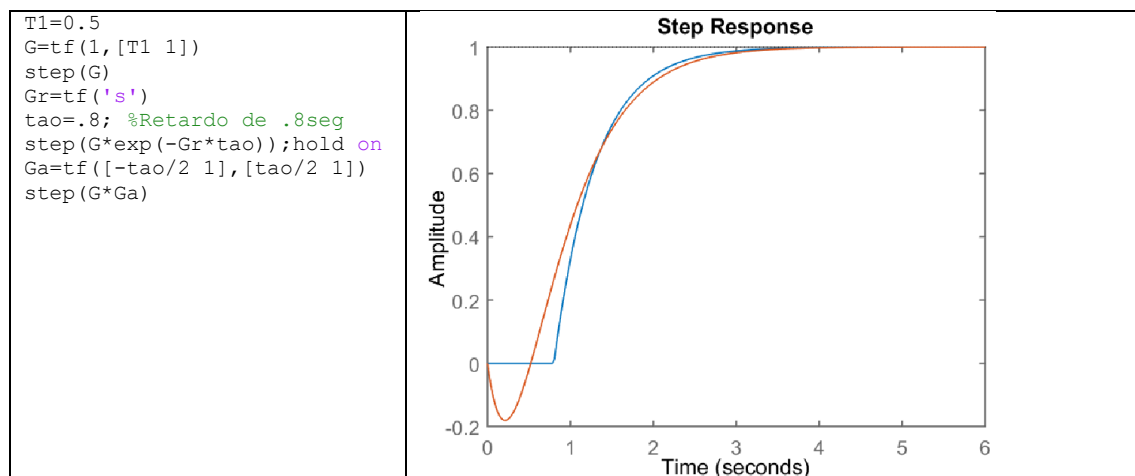


Fig. 2-9. Respuesta temporal donde se destaca un retardo y una derivada muy grande al inicio.



### Sistemas de segundo orden

Cuando la derivada inicial es nula, entonces la implementación del método anterior tendrá un error que puede no ser admisible para la solución que se está necesitando.

Para el caso de los sistemas de segundo orden con polos complejos conjugados,

$$G(s) = \frac{Y(s)}{U(s)} = K \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (2-14)$$

donde K es el valor que toma la salida en el estado estacionario,  $\omega_n$  es la frecuencia natural y  $\zeta$  es el amortiguamiento del sistema.

La respuesta temporal está dada, por ejemplo, para el caso sub-amortiguado ( $0 < \zeta < 1$ ) con los polos  $\alpha_i = -\zeta\omega_n \pm j\omega_d$ , siendo  $\omega_d = \omega_n \sqrt{1 - \zeta^2}$

$$y(t) = K \left[ 1 - \exp(-\zeta\omega_n t) \left( \cos(\omega_d t) + \frac{\zeta}{\sqrt{1 - \zeta^2}} \sin(\omega_d t) \right) \right] \quad (2-15)$$

que es lo mismo a

$$y(t) = K \left[ 1 - \frac{\exp(-\zeta\omega_n t)}{\sqrt{1 - \zeta^2}} \sin\left(\omega_n \sqrt{1 - \zeta^2} \cdot t + \cos^{-1}(\zeta)\right) \right] \quad (2-16)$$

cuya evolución temporal para el caso de  $K=1$  se muestra en la Fig. 2-10.

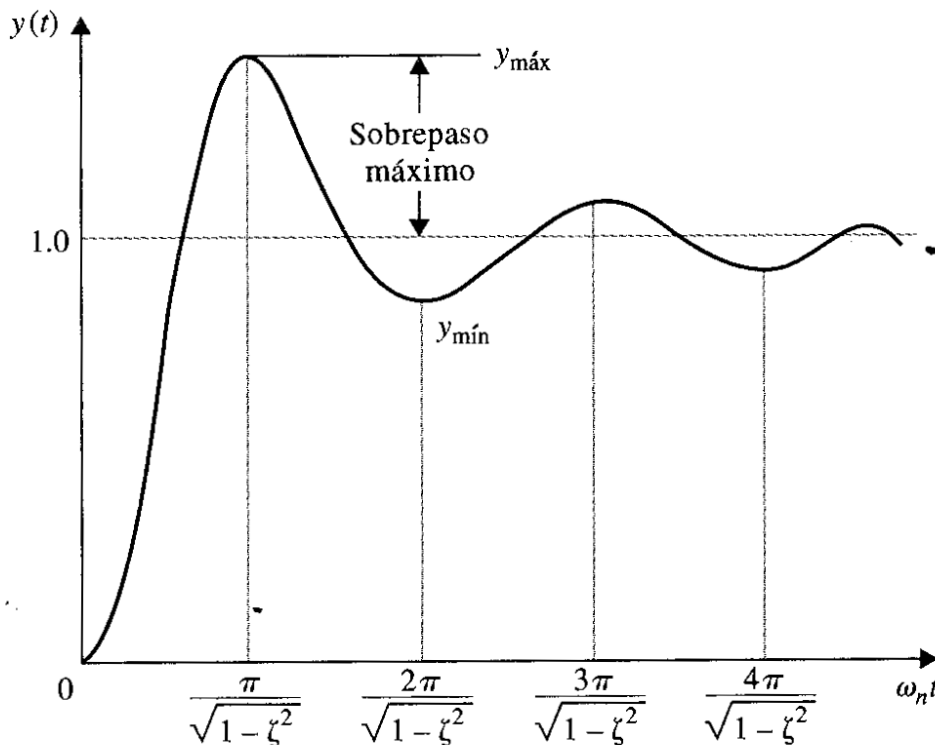


Fig. 2-10. Respuesta temporal donde se indican los tiempos de los valores picos.

Nótese que es factible emplear muestras de  $y(t)$  para obtener los parámetros que se están tratando de identificar aquí que son  $\omega_n$ ,  $\zeta$  y  $K$ . En la gráfica de la Fig. 2-10 se ha normalizado dividiendo por  $K$  a  $y(t)$  y al tiempo se lo expresa normalizado en  $\omega_n$ . Tener en cuenta que la relación entre pulsación natural medida en la Ec (2-14) y período natural observado en la misma figura está dado por el cociente

$$\omega_d = \frac{2\pi}{T_n}. \quad (2-17)$$

Para obtener el valor de  $\zeta$ , se puede emplear la igualdad (2-15) valuando al valor máximo de la salida y el tiempo en el que ése pico ocurre Figura 7-18 de [2]

$$y_{\text{Max}} = 1 + \exp\left(-\pi \frac{\zeta}{\sqrt{1-\zeta^2}}\right) \quad (2-18)$$

y de allí puede despejarse  $\zeta$  como

$$\zeta = \frac{\ln(y_{\text{Max}} - 1)}{\pi \sqrt{1 + \left[\frac{\ln(y_{\text{Max}} - 1)}{\pi}\right]^2}} \quad (2-19)$$

para el caso de que la respuesta al escalón corresponda a un sistema de ganancia unitaria con un escalón unitario. De otro modo, se deben realizar las escalas correspondientes para ajustar las ganancias. El método es directo, ya que a partir del valor máximo de  $y(t)$  se pueden obtener los polos del sistema de segundo orden con polos complejos conjugados. En la Fig. 2-11 se muestra el empleo de este método para un sistema de polos complejos conjugados de ganancia unitaria, tiene superpuestas las respuestas del sistema real y del sistema identificado. También tiene indicados los valores máximos y mínimos de  $y(t)$ . En la Tabla 2-4 está el código para Octave.

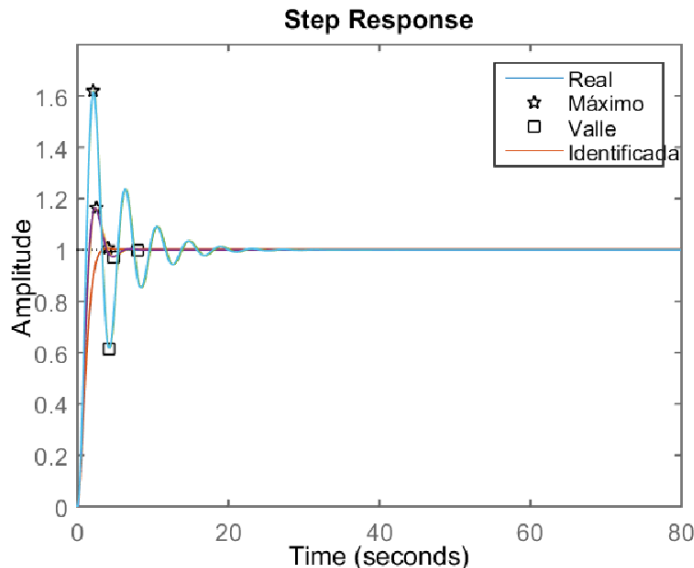


Fig. 2-11. Respuesta al escalón de dos sistemas identificados mediante el método detallado en [2].

```
clear all;
% Kuo Sistemas de control automático Fig 7-18, Sección 7-5 Pp395.
%Codigo realizado por JAP
clear all;
wn_r=1.5;sita_r=.85;TMax=20;
```

```

pkg load control
pkg load signal
% wn_r=1.5;sita_r=.5;TMax=50;
% wn_r=1.5;sita_r=.15;TMax=120;
sys_G=tf([wn_r^2],[1 2*sita_r*wn_r wn_r^2]);
sys_real=sys_G
[y,t0]=step(sys_G,TMax/wn_r);
step(sys_G,TMax/wn_r)
[ymax a ]=max(y);
t1=t0(a);
[ymin b ]=min(y(a:end));
t2=t0(b+a-1);
hold on
plot(t1,ymax,'pk');
plot(t2,ymin,'sk');
w_n=(2*pi)/(2*(t2-t1));
beta=-log(ymax-1)/pi;
sita_id=beta/(sqrt(1+beta^2));
wn_id=w_n/(sqrt(1-sita_id^2));
den_id=[1 2*sita_id*wn_id wn_id^2];
sys_id=tf(wn_id^2,den_id)
step(sys_id)
legend('Real','Máximo','Valle','Identificada')
% figure(1);
% Nombre_Figura=['Identificacion_Simple'];
% print('-dtiff','-r600',Nombre_Figura);%--FIGURA temporal

```

**Tabla 2-4. Código que permite identificar una planta de segundo orden con polos complejos conjugados de ganancia unitaria.**

### Extensión a sistemas de dos polos reales y un cero

Se detalla un método originalmente presentado en [4]. Asumiendo una función de transferencia con  $n$  polos y  $n-1$  ceros expresada como

$$G(s) = \frac{Y(s)}{U(s)} = K \frac{(T_{n+1}s + 1)(T_{n+2}s + 1) \cdots (T_{2n-1}s + 1)}{(T_1s + 1)(T_2s + 1) \cdots (T_ns + 1)} \quad (2-20)$$

se propone desarrollar un método basado en la respuesta al escalón que permita hallar las constantes de tiempo  $T_i$ , ya que la constante  $K$  se halla mediante el teorema del valor final.

Por ejemplo, para el caso en que dos de los polos sean iguales y asumiendo un cero, se tiene una función de transferencia del tipo

$$G(s) = \frac{Y(s)}{U(s)} = K \frac{(T_2s + 1)}{(T_1s + 1)^2} \quad (2-21)$$

donde la respuesta al escalón en el dominio del tiempo es

$$y(t) = K \left[ 1 - e^{-\frac{t}{T_1}} - \frac{T_1 - T_2}{T_1^2} t e^{-\frac{t}{T_1}} \right] \quad (2-22)$$

Se emplea el método descrito en [4], donde se debe muestrear la respuesta temporal en intervalos equidistantes, denominados  $\{t_1, 2t_1\}$ , y se obtienen los pares  $(x,y)$  como  $(t_1, y(t_1)), (2t_1, y(2t_1))$ .

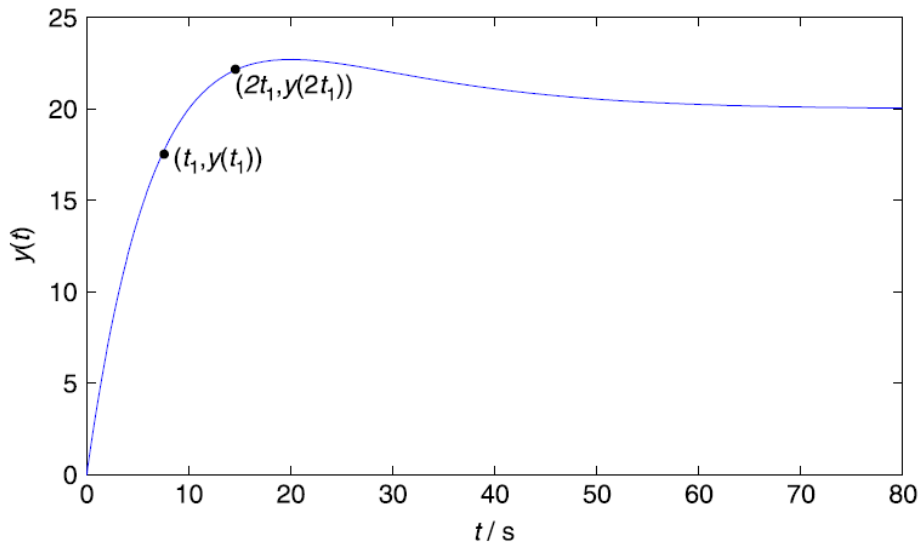


Fig. 2-12. Respuesta al escalón de un sistema con dos polos reales iguales.

Este muestreo deja dos ecuaciones con dos incógnitas, y el valor de  $K$  se obtiene a partir de la magnitud del escalón y el valor de  $y(t)$  cuando  $t \rightarrow \infty$  empleando el mencionado teorema del valor final.

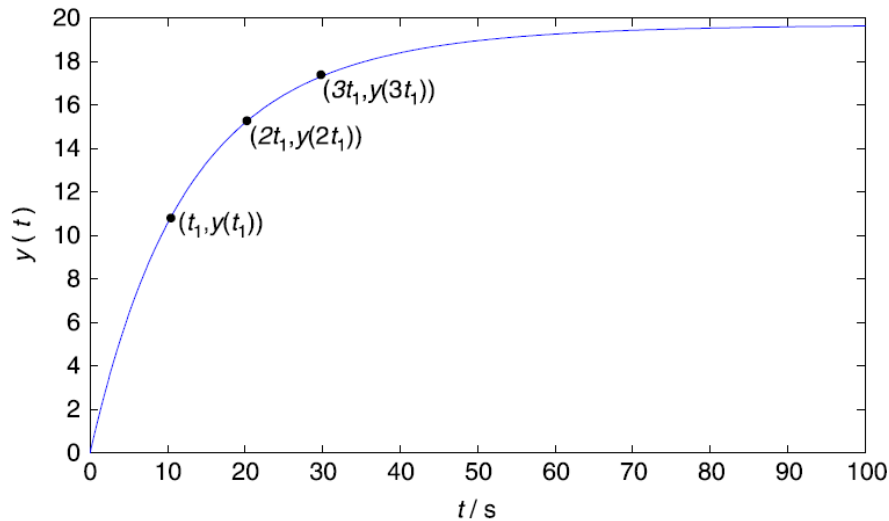
Si los polos fueran distintos, también se emplea el método descrito en [4], pero esta vez se toman más mediciones de la salida  $y(t)$  del sistema a intervalos  $t_1, 2t_1, 3t_1$ , donde las constantes de tiempo estarían determinadas por un sistema con tres ecuaciones y tres incógnitas

$$y(t_1) = K \left[ 1 + \frac{T_3 - T_1}{T_1 - T_2} e^{-\frac{t_1}{T_1}} - \frac{T_3 - T_2}{T_1 - T_2} e^{-\frac{t_1}{T_2}} \right] \quad (2-23)$$

$$y(2t_1) = K \left[ 1 + \frac{T_3 - T_1}{T_1 - T_2} e^{-\frac{2t_1}{T_1}} - \frac{T_3 - T_2}{T_1 - T_2} e^{-\frac{2t_1}{T_2}} \right] \quad (2-24)$$

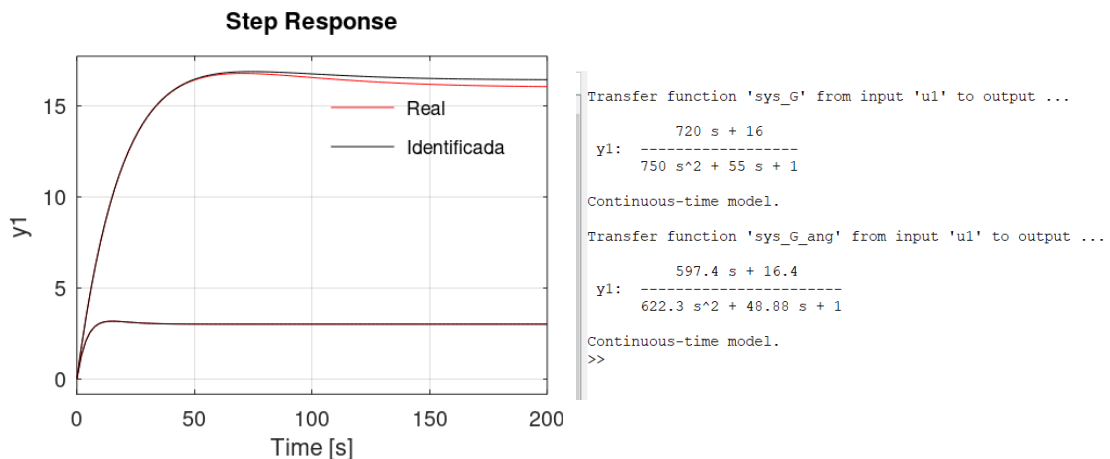
$$y(3t_1) = K \left[ 1 + \frac{T_3 - T_1}{T_1 - T_2} e^{-\frac{3t_1}{T_1}} - \frac{T_3 - T_2}{T_1 - T_2} e^{-\frac{3t_1}{T_2}} \right], \quad (2-25)$$

donde  $T_1, T_2$  son las constantes de tiempo del denominador de la función de transferencia  $T_3$  del numerador y  $K$  es la ganancia de estado estacionario. Es decir, se asume un orden determinado en numerador y denominador. El ensayo puede resultar, por ejemplo, en la obtención de parámetros que den una respuesta al escalón similar a la del sistema real.



**Fig. 2-13.** Muestreo de la respuesta al escalón de un sistema de segundo orden con polos reales y distintos.

En la Tabla 2-5 se muestra un código que permite resolver la identificación de los parámetros para polos reales y distintos mediante la respuesta al escalón. En la misma tabla se encuentran diferentes casos y diferentes ganancias. En la Fig. 2-14 se muestran los resultados obtenidos para uno de ellos. Nótese que es difícil distinguir la respuesta del modelo identificado del real. En la Fig. 2-14 se muestra la respuesta al escalón empleada para obtener la función de transferencia incógnita, detalla a su derecha junto con la original.



**Fig. 2-14.** Respuesta al escalón de un sistema identificado mediante el método detallado en [4].

```
clear all;clc;
pkg load control;
pkg load signal; %pkg install control-3.4.0.tar.gz
% Extraído de "Identification for the second-order systems based on the step response"
% Lei Chen, Junhong Li, Ruifeng Ding
% Mathematical and Computer Modelling 53 (2011) 1074-1083
%Codigo realizado por JAP
% syms k1 k2 k3 alfa1 alfa2 beta
% s=solve('alfa1*alfa2=(k2^2-k1*k3)/(k1^2+k2)',...
% 'alfa1+alfa2=(k3+k1*k2)/(k1^2+k2)', 'alfa1', 'alfa2');
% simplify(s.alfa1)
% simplify(s.alfa2)
% alfa1=s.alfa1(1);
% alfa2=s.alfa1(2);
% s1=solve('beta*(alfa1-alfa2)=k1+alfa2',...
```

```

%      'beta*(alfa1^2-alfa2^2)=k2+alfa2^2',...
%      'beta*(alfa1^3-alfa2^3)=k3+alfa2^3','beta');
% sys_G=tf(2*[8 1],conv([5 1],[6 1])); %la otra planta
sys_G=tf(16*[45 1],conv([25 1],[30 1]))
%sys_G=tf(3*[10 1],conv([6 1],[6 1])); %esta tiene los polos iguales
% sys_G=tf(2*[8 1],[1 2 2])); %la otra planta
StepAmplitude = 1;
t_s=0:2:50;
[y,t0]=step(StepAmplitude*sys_G,t_s);
t_inic=4;
[val lugar] =min(abs(t_inic-t0)); y_t1=y(lugar);
t_t1=t0(lugar);
ii=1;
[val lugar] =min(abs(2*t_inic-t0));
t_2t1=t0(lugar);
y_2t1=y(lugar);
[val lugar] =min(abs(3*t_inic-t0));
t_3t1=t0(lugar);
y_3t1=y(lugar);
K=y(end)/StepAmplitude;
k1=(1/StepAmplitude)*y_t1/K-1;%Afecto el valor del Escalon
k2=(1/StepAmplitude)*y_2t1/K-1;
k3=(1/StepAmplitude)*y_3t1/K-1;
be=4*k1^3*k3-3*k1^2*k2^2-4*k2^3+k3^2+6*k1*k2*k3;
alfa1=(k1*k2+k3-sqrt(be))/(2*(k1^2+k2));
alfa2=(k1*k2+k3+sqrt(be))/(2*(k1^2+k2));
beta=(k1+alfa2)/(alfa1-alfa2);
%(2*k1^3+3*k1*k2+k3-sqrt(be))/(sqrt(be));
% alfa2= (k3 + k1*k2 + (4*k1^3*k3 - 3*k1^2*k2^2 + 6*k1*k2*k3 - 4*k2^3 +
k3^2)^(1/2))/(2*(k1^2 + k2));
% alfa1= (k3 + k1*k2 - (4*k1^3*k3 - 3*k1^2*k2^2 + 6*k1*k2*k3 - 4*k2^3 +
k3^2)^(1/2))/(2*(k1^2 + k2));
T1_ang=-t_t1/log(alfa1);
T2_ang=-t_t1/log(alfa2);
T3_ang=beta*(T1_ang-T2_ang)+T1_ang;
T1(ii)=T1_ang;
T2(ii)=T2_ang;
T3(ii)=T3_ang;
T3_ang=sum(T3/length(T3));
T2_ang=sum(T2/length(T2));
T1_ang=sum(T1/length(T1));
sys_G_ang=tf(K*[T3_ang 1],conv([T1_ang 1],[T2_ang 1]))
step(StepAmplitude*sys_G,'r',StepAmplitude*sys_G_ang,'k',200),hold on
legend('Real','Identificada');
legend('boxoff');

```

Tabla 2-5. Código para Octave que permite identificar una planta de segundo orden con polos reales y distintos.

### 3. Forma lineal general de orden $n$

Cuando se linealiza a las funciones  $f$  y  $g$  en un punto de operación ( $\mathbf{x}_{OP}$ ,  $\mathbf{u}_{OP}$ ), se obtiene la representación lineal general

$$\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) \quad (3-1)$$

y para la salida

$$\mathbf{y}(t) = \mathbf{C}(t)\mathbf{x}(t) + \mathbf{D}(t)\mathbf{u}(t) \quad (3-2)$$

donde  $\mathbf{A}(t)$  es la *matriz de estados*,  $\mathbf{B}(t)$  es la *matriz de entrada*,  $\mathbf{C}(t)$  es la *matriz de salida* y  $\mathbf{D}(t)$  es la *matriz de transmisión directa*. En la Fig. 3-1 se representa el diagrama en bloques de este modelo dinámico.

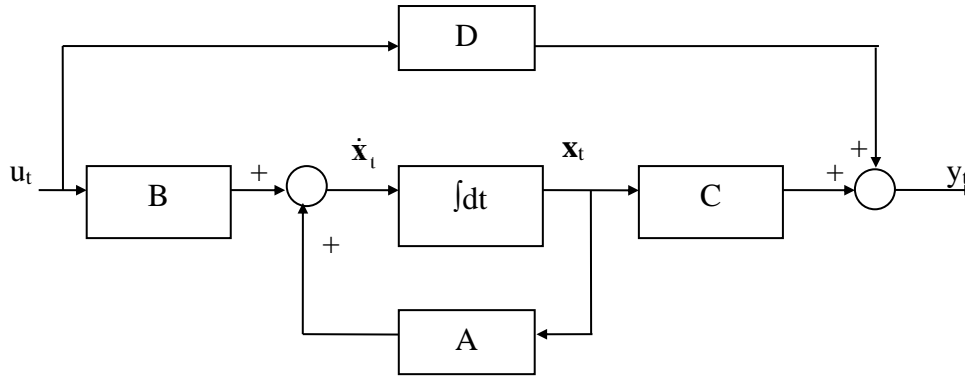


Fig. 3-1. Diagrama en bloques de un modelo dado por las Ecs (3-1) y (3-2).

Si las funciones vectoriales  $f(\cdot)$  y  $g(\cdot)$  no incorporan explícitamente al tiempo  $t$ , el sistema se denomina sistema lineal invariante en el tiempo, y las Ecs (3-1) y (3-2) se simplifican como

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad (3-3)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \quad (3-4)$$

donde la Ec (3-3) es la *ecuación de estados* y la Ec (3-4) es la *ecuación de salida* del sistema lineal invariante en el tiempo con  $\mathbf{x}_0 = \mathbf{x}(0)$ .

### 3.1. Relación entre funciones de transferencia y ecuaciones en espacio de estados

Se pretende obtener la función de transferencia de un sistema representado en el espacio de estados. Se asume un sistema de una entrada y una salida, con la función

$$\frac{Y(s)}{U(s)} = G(s) \quad (3-5)$$

donde  $Y$ ,  $U$  son las salida y entrada en el dominio  $s$  del sistema.

En el espacio de estados, el sistema tiene la representación de las Ecs (3-3) y (3-4), cuya transformada de Laplace es

$$s\mathbf{X}(s) - \mathbf{X}(0) = \mathbf{A}\mathbf{X}(s) + \mathbf{B}\mathbf{U}(s) \quad (3-6)$$

$$\mathbf{Y}(s) = \mathbf{C}\mathbf{X}(s) + \mathbf{D}\mathbf{U}(s) \quad (3-7)$$

pero como por *definición de función de transferencia* se consideran las condiciones iniciales nulas, lo que resulta la transformada  $s$  de la ecuación de estados en

$$\mathbf{X}(s) = (s\mathbf{I} - \mathbf{A})^{-1} \mathbf{B}\mathbf{U}(s) \quad (3-8)$$

que si se sustituye en la (3-7) se llega a

$$\mathbf{Y}(s) = [\mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1} \mathbf{B} + \mathbf{D}]\mathbf{U}(s). \quad (3-9)$$

Comparando la Ec. (3-9) con la Ec (3-5), se observa que

$$G(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1} \mathbf{B} + \mathbf{D}, \quad (3-10)$$

que es la expresión de la función de transferencia en términos de A, B, C y D.

Además, desarrollando la inversión de la matriz  $(sI-A)$ , se tiene

$$(sI - \mathbf{A})^{-1} = \frac{1}{|sI - \mathbf{A}|} \text{adj}(sI - \mathbf{A}) \quad (3-11)$$

se tendrá el término  $|sI-A|$  como denominador común, y como numerador un polinomio  $Q(s)$ , tal que

$$G(s) = \frac{Q(s)}{|sI - \mathbf{A}|} \quad (3-12)$$

de donde se concluye que  $|sI-A|$  es igual a la *ecuación característica* de  $G(s)$ .

Para el caso de que el vector  $\mathbf{u}$  tenga  $r$  componentes y el vector  $\mathbf{y}$  tenga  $m$  componentes, entonces la Ec (3-5) se convierte en

$$\mathbf{Y}(s) = \mathbf{G}(s)\mathbf{U}(s) \quad (3-13)$$

donde  $G(s)$  tiene  $m$  filas y  $r$  columnas y se denomina *matriz de transferencia*.

### 3.2. Bibliografía

- [1] [Sontag](http://www.sontaglab.org). Mathematical control theory 1998. Pag 104. <http://www.sontaglab.org>.
- [2] [Kuo](#), B. Sistemas de control automático. 1996. Prentice Hall.
- [3] [Ogata](#), K. Modern Control Engineering. 1997. Prentice Hall.
- [4] Lei [Chen](#), Junhong Li. “Identification for the second-order systems based on the step response”. Ruifeng Ding Mathematical and Computer Modelling 53 (2011) 1074–1083. <https://linkinghub.elsevier.com/retrieve/pii/S0895717710005613>.