Parcial 2 – Modelación Experimental– Punto 2

Andrés Grimaldos Echavarría

Andrés Grimaldos Echavarría

2.
$$\begin{cases} \dot{x}_1 = -a x_1 + b x_1 x_2 \\ \dot{x}_2 = c x_1^2 - d x_2 + u \end{cases}$$

Dado que no tenemos una forma lineal para la ecuación de estado, utilizamos el filtro de Kalman extendido. Así, hacemos:

- Discretización: (Método de Euler)

$$\begin{cases} x_1(k+1) = x_1(k) + T(b\ x_1(k)x_2(k) - a x_1(k)) \\ x_2(k+1) = x_2(k) + T(c x_1^2(k) - d x_2(k) + u(k)) \end{cases}$$

siendo T el tamaño de paso o tiempo de muestreo.

- Hallamos Jacobiano, matriz para linealizar, pero sin ser evaluado en un punto de equilibrio:

$$J = \begin{bmatrix} -a + b\ x_2 & b x_1 \\ 2 c x_1 & -d \end{bmatrix}$$

- Hallamos la matriz del modelo linealizado discretizado:

$$A = I + TJ = \begin{bmatrix} 1 + T(b\ x_2 - a) & T b x_1 \\ 2 T c x_1 & 1 - dT \end{bmatrix}$$

- Así, con el sistema:

$$x(k+1) = f(x_1(k), u(k)) + G w(k)$$
$$y(k) = C x + n(k)$$

Siendo $w(k)$ y $n(k)$ ruidos blancos y además, $E[n(k)n^T(k)] = N$ y $E[w(k)w^T(k)] = M$. Y midiendo x_1 y estimando x_2, tenemos que:

$C = [1\ \ 0]$ y así, nuestras ecuaciones para el filtro de Kalman serían:

Predicción:

$$x(k+1|k) = f(x(k|k), u(k)) = \begin{bmatrix} x_1(k|k) + \tau(b x_1(k|k) x_2(k|k) - a x_1(k|k)) \\ x_2(k|k) + \tau(c x_1^2(k|k) - d x_2(k|k) + u(k)) \end{bmatrix}$$

$$P(k+1|k) = A P(k|k) A^T + V M V^T$$

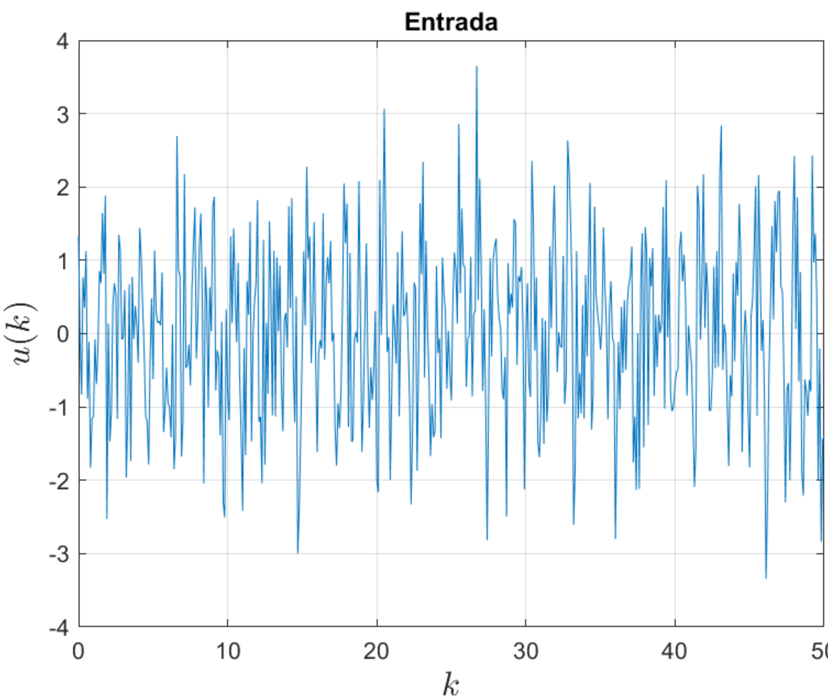
Corrección:

$$K(k+1) = P(k+1|k) C^T [C P(k+1|k) C - N]^{-1}$$

$$x(k+1|k+1) = x(k+1|k) + K(k+1)[y(k+1) - C x(k+1|k)]$$

$$P(k+1|k+1) = [I - K(k+1) C] P(k+1|k)$$

Ahora, como tenemos el modelo, podemos simular una solución con las condiciones iniciales $x_0 = (1,1), a = 0.01, b = 0.02, c = 0.1$ y $d = 0.2$, además, usaremos una entrada dada por $u(t) = sen(5t) + r$, donde $r \sim N(0,1)$, dando la entrada:



Ahora, bien el código para esto fue:

```
Ts = 0.1; % tiempo de muestreo T.
tiempo = 0:Ts:50; % periodo de tiempo a usar.
x0 = [1, 1]; % Condición inicial
% Probamos dos entradas:
u = sin(5*tiempo) + randn(1,length(tiempo));%ones(1, length(tiempo));%sin(5*
plot(tiempo, u)
title('Entrada');
xlabel('$k$', 'Interpreter', 'Latex', 'FontSize', 15);
ylabel('$u(k)$', 'Interpreter', 'Latex', 'FontSize', 15);
grid on
a = 0.01; b = 0.02; , c = 0.1; d = 0.2;
[t, x] = ode45(@(t,x) ode_parcial(t, x, u, tiempo, a, b, c, d), tiempo, x0);
```

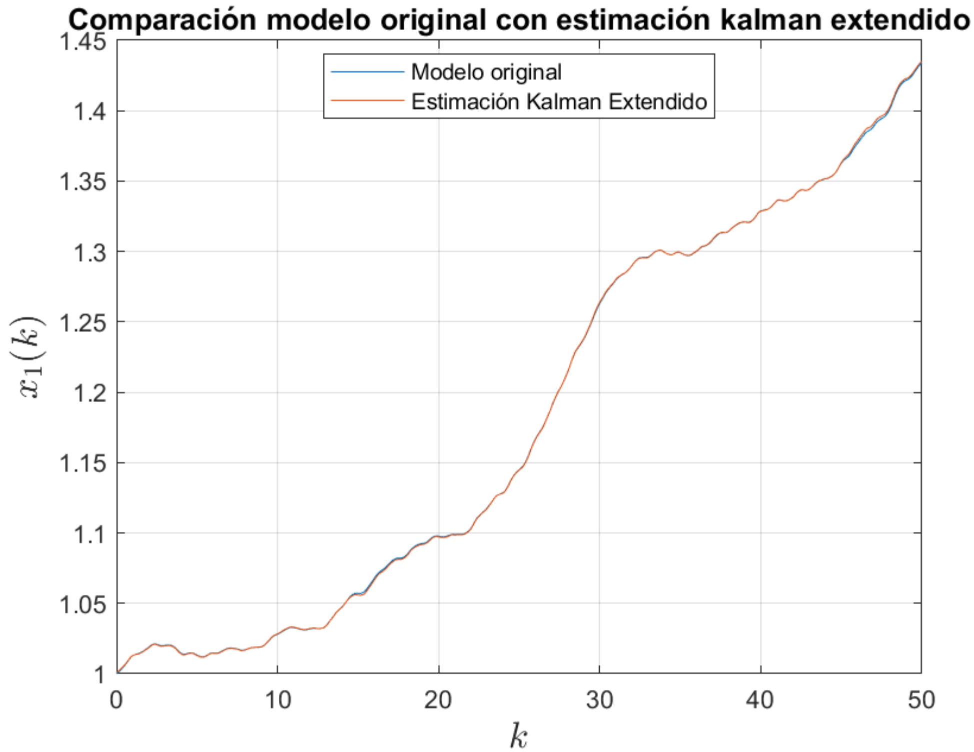
Donde la función ode_parcial es:

```
function dxdt = ode_parcial(t, x, u, tiempo, a, b, c, d)
    u_int = interp1(tiempo, u, t);
    dxdt = [-a*x(1) + b*x(1)*x(2); c*x(1)^2 - d*x(2) + u_int];
end
```

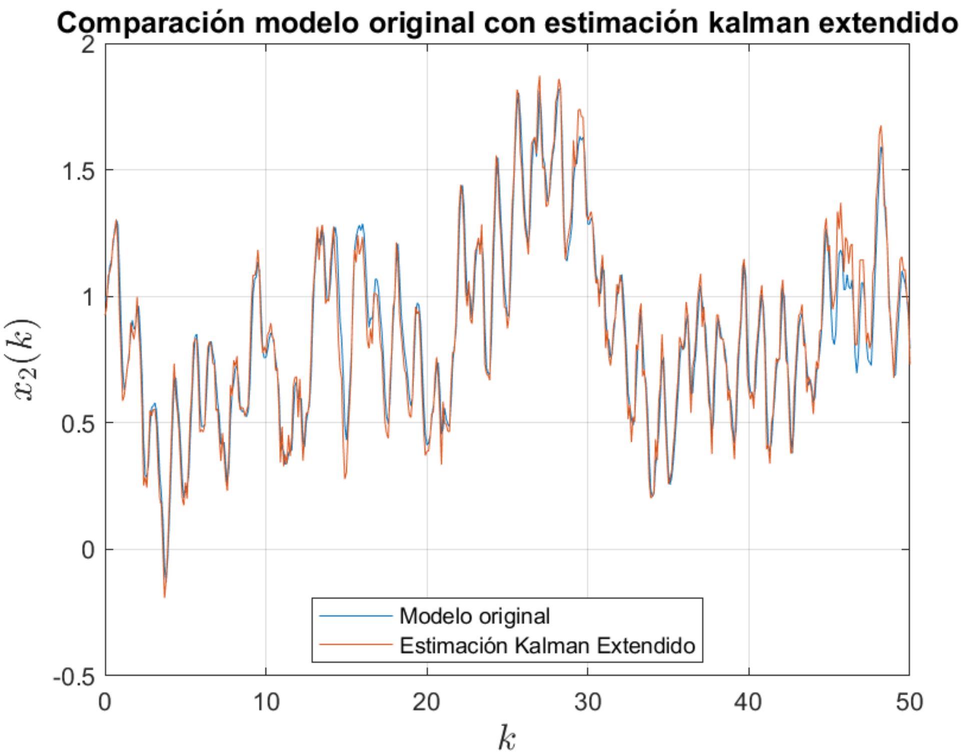
Ahora bien, para la implementación del filtro de Kalman, tenemos el código donde se especifica las condiciones de cada término:

```
y = x(:,1);
% Condiciones iniciales dados por el usuario:
M = 0.01*eye(2); N = 0.5; P0 = M; x0 = [1 1]'; Ts = 0.1;
% Parámetros conocidos:
C = [1 0];
n = length(y); x_est = zeros(n,2); P_trace = zeros(n,1); K_norm = zeros(n,1);
x_act = x0; P_act = P0;
for k = 1:n
    A_ = A(Ts,a,b,c,d,x_act(1), x_act(2));
    % Predicción:
    x_pred = [x_act(1) + Ts*(b*x_act(1)*x_act(2)-a*x_act(1));
              x_act(2) + Ts*(c*x_act(1)^2 - d*x_act(2) + u(k))];
    P_pred = A_*P_act*A_' + M;
    % Corrección:
    K = P_pred*C'*inv(C*P_pred*C' + N);
    x_act = x_pred + K*(y(k) - C*x_pred);
    P_act = (eye(2) - K*C)*P_pred;
    % Estimación:
    x_est(k,:) = x_act';
    % Medidas:
    P_trace(k) = trace(P_act);
    K_norm(k) = norm(K);
end
plot(t, x(:,1), t, x_est(:,1))
legend({'Modelo original', 'Estimación Kalman Extendido'}, 'Location', 'best');
xlabel('$k$', 'Interpreter', 'Latex', 'FontSize', 15);
ylabel('$x_{2}(k)$', 'Interpreter', 'Latex', 'FontSize', 15);
title('Comparación modelo original con estimación kalman extendido', 'FontSize', 12);
grid on
```

Donde podemos hacer la comparación para x_1, ya que fue medida (simulada, en este caso):



Además, podemos también comparar x_2, ya que conocemos el modelo y fue simulada, claramente, en la vida real, no tenemos esta posibilidad, pues no conocemos el modelo subyacente, ya que, si lo tuviéramos, no tendría sentido hacer la estimación, pero acá por ver el resultado del código, se hace en la figura siguiente:



Ahora, como se observa, el filtro de Kalman extendido nos da una gran aproximación, mostrando así los buenos resultados que da e incluso su robustez. Por otro lado, el código de todo se encuentra en el archivo de MATLAB.