

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA
SECCIÓN DE ELECTRICIDAD Y ELECTRÓNICA

TEORÍA DE CONTROL 2

Laboratorio N°4



Laboratorio de simulación de sistemas de control digital

Salvador Yábar

20200408

H0821

2024-1

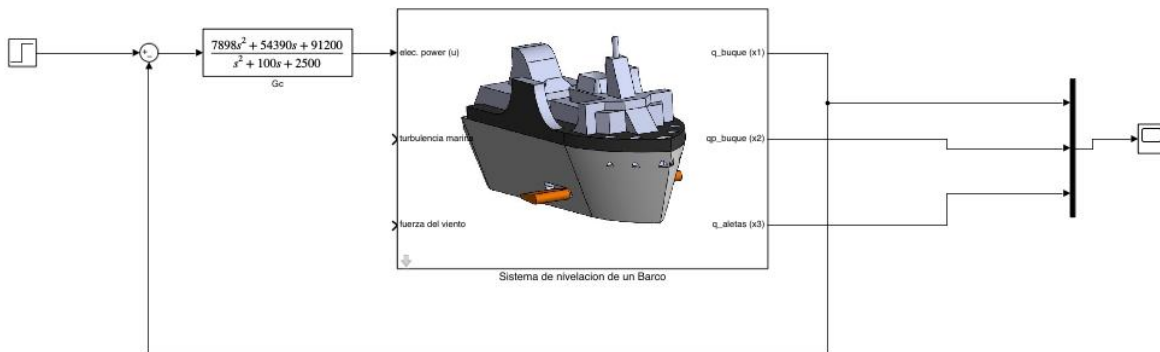
1. Objetivos

- Determinar un periodo de muestreo apropiado para el sistema
- Discretizar el controlador en Matlab
- Simular el controlador discreto

2. Desarrollo

a) Simular el controlador continuo

Se implementó el controlador continuo mediante el bloque *transfer function* en Simulink. Se realizó el esquema que se muestra en la figura 1, colocando como referencia 0.1 radianes, como plantea la guía.



A partir de la simulación, se obtuvo la siguiente gráfica en la que se observan las 3 salidas:

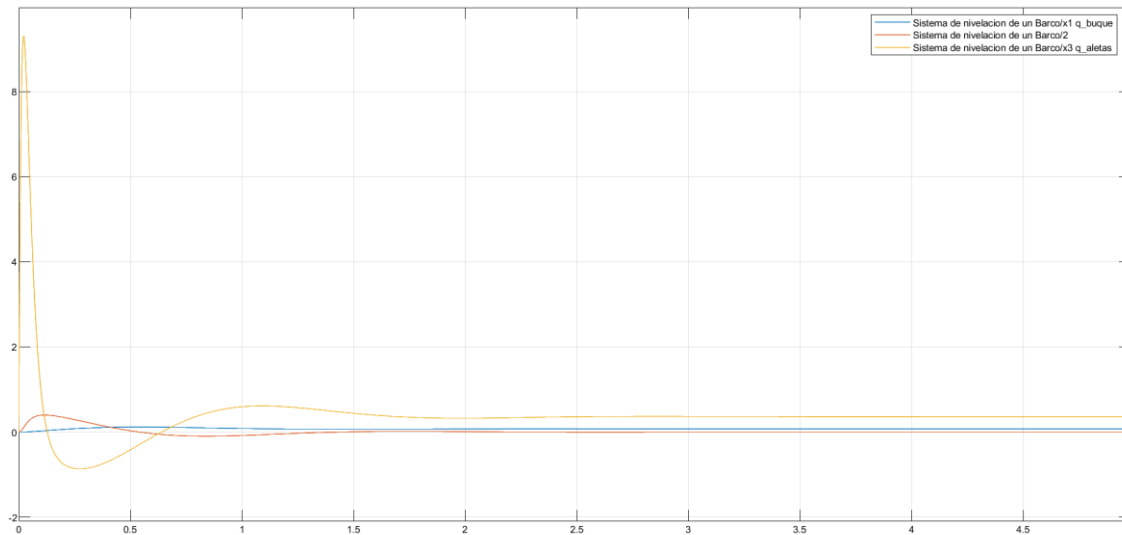


Fig 2. Gráfica de las variables de estado del sistema

Se observa que el menor tiempo de subida lo tiene la salida que corresponde al ángulo de las aletas.

A continuación, se analiza el tiempo de subida para cada salida.

Para el ángulo del buque, se tiene un máximo de 1.176×10^{-1} . Se colocan los cursores en los valores correspondientes al 10 y 90%, y se obtiene un tiempo de subida de 306.164 ms.

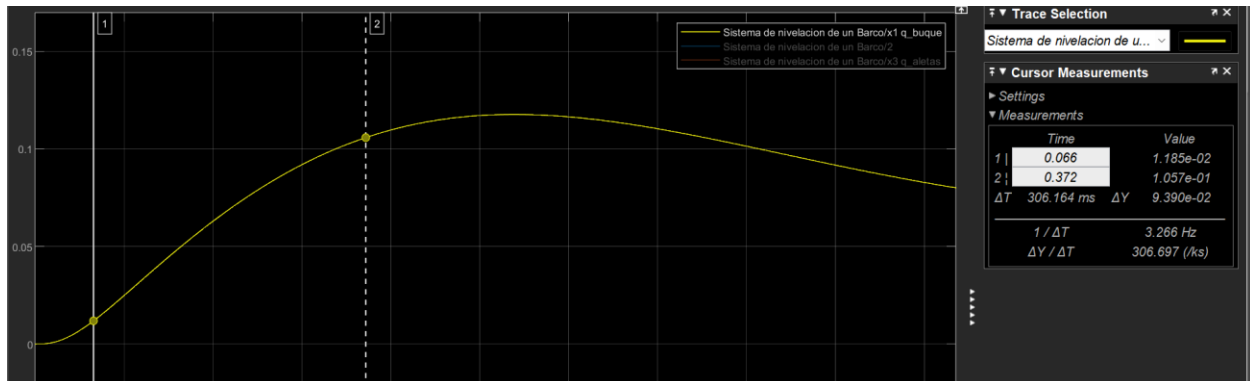


Fig 3. Gráfica de x_1

De forma similar, para x_2 se tiene un máximo de 3.992×10^{-1} . Se obtiene un tiempo de subida de 60.19 ms.

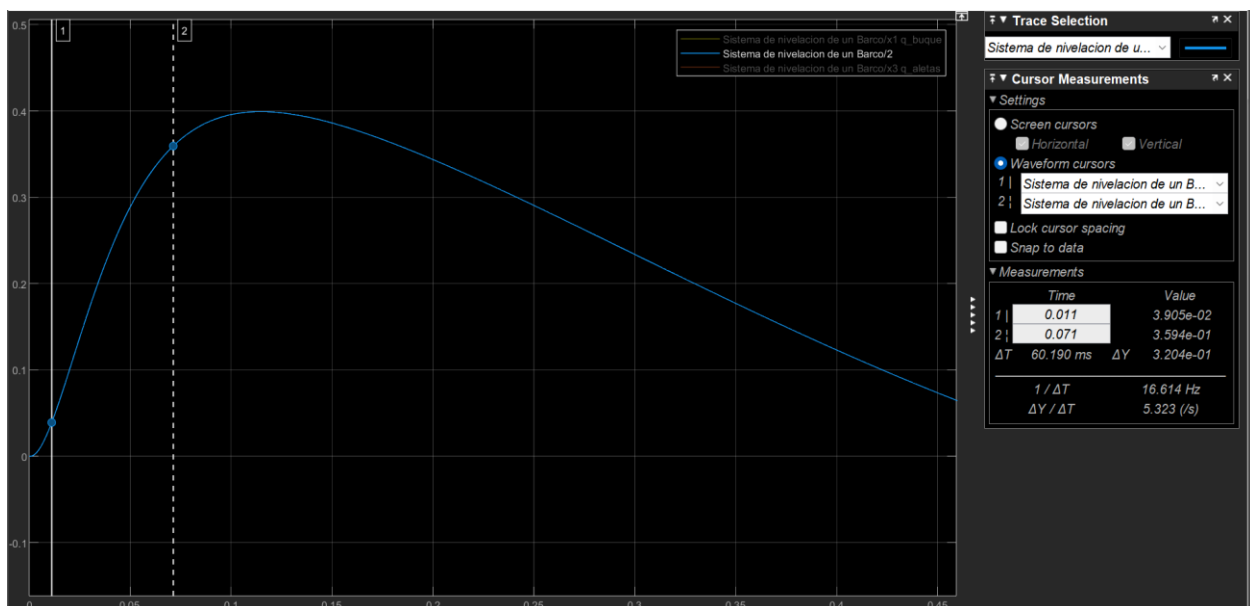


Fig 4. Gráfica de x_2

Finalmente, para x_3 se tiene un máximo de 9.304, y se obtiene un tiempo de subida de 20.298 ms.

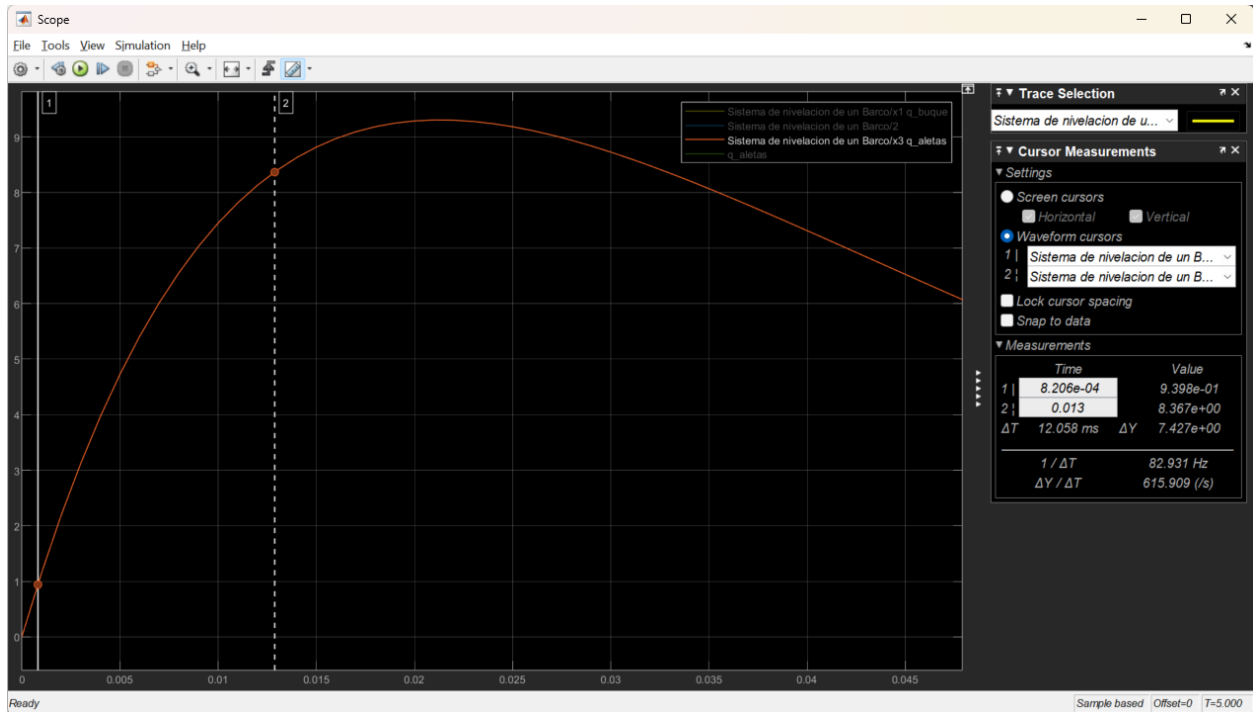


Fig 5. Gráfica de x_3

A partir de esto, se confirma que el tiempo de subida menor es el de x_3 . Se emplea este valor para obtener el periodo de muestreo, dividiendo el tiempo de subida entre 10.

b) Discretizar el controlador

Se determina el periodo de muestreo, y mediante la función `c2d` se obtiene el controlador discreto G_z .

```
Tsubida = 20.298 * 10^(-3);
T = Tsubida/10;
Gc = tf([7898 54390 91200], [1 100 2500])
Gz = c2d(Gc, T, 'zoh')
```

Fig 6. Código para obtener controlador discreto

Gz =

$$\frac{7898 z^2 - 1.566e04 z + 7760}{z^2 - 1.807 z + 0.8163}$$

Sample time: 0.0020298 seconds
Discrete-time transfer function.

Fig 7. Controlador discreto

Luego, se emplea el bloque de función de transferencia discreta para implementar el controlador discreto en Simulink, como se muestra en la figura 8.

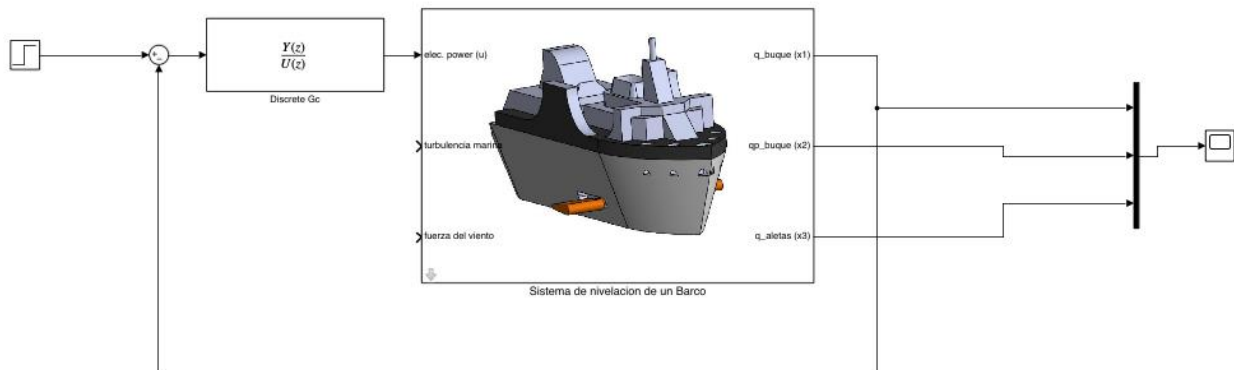


Fig 8. Simulink con controlador discreto

Se presenta la gráfica correspondiente a las variables de estado del sistema, empleando el controlador discreto.

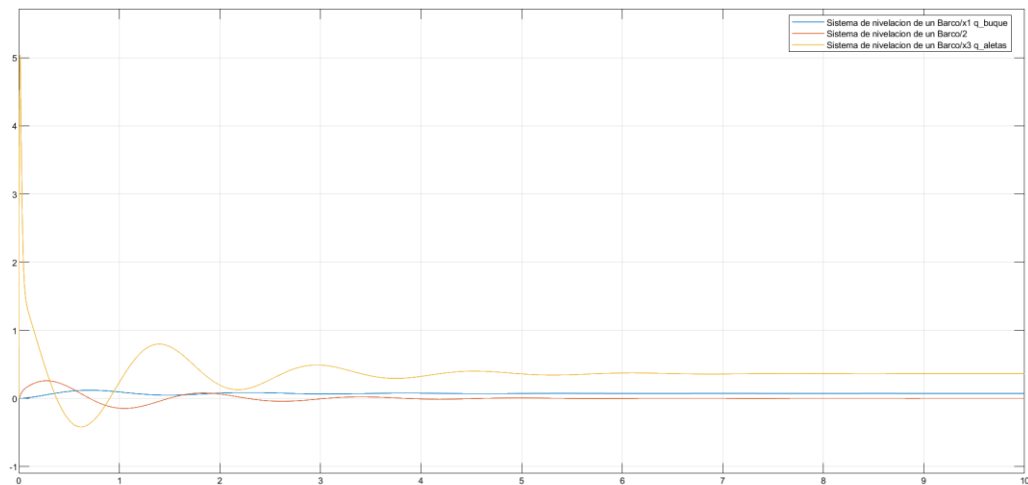


Fig 9. Gráfica de variables de estado

c) Simulación en Matlab

Primero, se discretiza la planta mediante el comando c2d.

```
% Discretizar la planta
Gp = tf(1.16, [1 4.4 5.6 16])
Gpz = c2d(Gp, T, 'zoh')
```

Fig 10. Código para discretizar la planta

$$G_{pz} = \frac{1.613e-09 z^2 + 6.439e-09 z + 1.606e-09}{z^3 - 2.991 z^2 + 2.982 z - 0.9911}$$

Fig 11. Planta discretizada

Se obtiene la función de transferencia en lazo cerrado mediante el comando feedback.

```
Gsz = feedback(Gz*Gpz, 1, -1)
```

Fig 12. Lazo cerrado

$$G_{sz} = \frac{1.274e-05 z^4 + 2.559e-05 z^3 - 7.561e-05 z^2 + 2.482e-05 z + 1.246e-05}{z^5 - 4.798 z^4 + 9.203 z^3 - 8.822 z^2 + 4.225 z - 0.809}$$

Fig 13. Función de transferencia en lazo cerrado

Se obtiene la ecuación en diferencias en base a la función de transferencia

```
q = 7898*r - 3.928e04*r_1 + 7.815e04*r_2 - 7.774e04*r_3 + 3.866e04*r_4 - 7691*r_5  
+ 4.798*q_1 - 9.203*q_2 + 8.822*q_3 + 4.225*q_4 - 0.809*q_5;
```

Se muestra el código para obtener los valores de $q(k)$ a partir de la ecuación en diferencias.

```
for k=0:kf
    q = 7898*r - 3.928e04*r_1 + 7.815e04*r_2 - 7.774e04*r_3 + 3.866e04*r_4 - 76
    qdato = [qdato q];
    q_5 = q_4;
    q_4 = q_3;
    q_3 = q_2;
    q_2 = q_1;
    q_1 = q;
    r_5 = r_4;
    r_4 = r_3;
    r_3 = r_2;
    r_2 = r_1;
    r_1 = r;
end;

stem((0:kf)*T, qdato)
```

Fig 14. For loop para la ecuación en diferencias

Sin embargo, la gráfica obtenida no es la esperada. Esta debería asemejarse al resultado obtenido en el inciso b, ya que se trata del mismo controlador discreto y la misma planta.

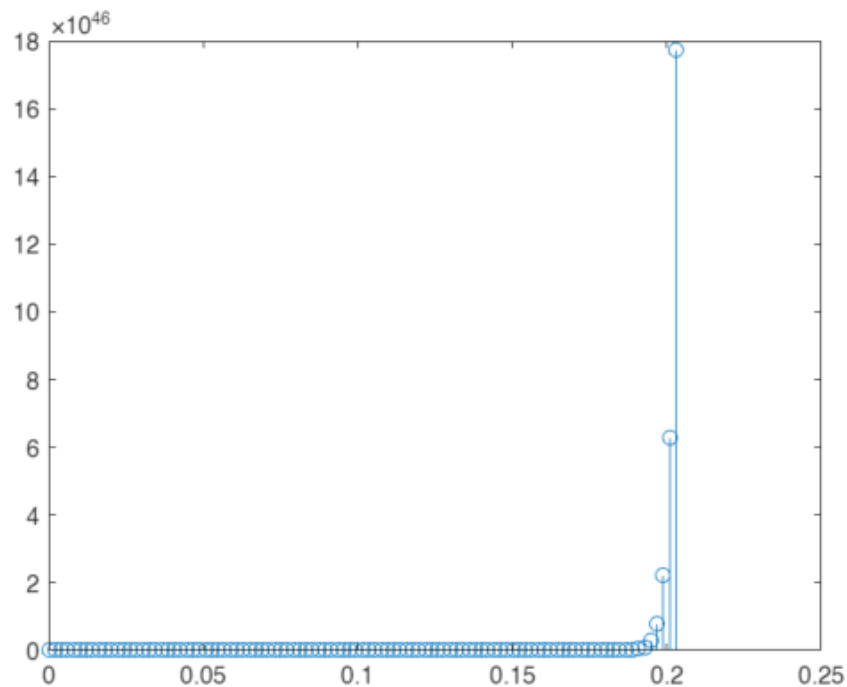


Fig 15. Gráfica de q

3. Conclusiones

En conclusión, se pudo determinar un periodo de muestreo apropiado para el sistema, en base al tiempo de subida de la variable más rápida. Al comparar las 3 variables de estado, se determinó que x_3 era la más rápida, y se empleó un periodo de muestreo de la décima parte de su tiempo de subida.

Empleando el comando `c2d` en Matlab se pudo obtener el controlador discreto y simularlo con la planta en Simulink. Se obtuvo un resultado similar al de la planta con controlador continuo.

Finalmente, se discretizó la planta empleando el comando `c2d`, pero al realizar la gráfica se obtuvo un resultado no esperado.