

Control Anticipativo

Juan Pablo Velandia Suarez, 45161133

Harold David León Hurtado, 45161031

Automatización de procesos - Ing. Tumialan Borja José Antonio
Universidad de la Salle

I. RESUMEN

El presente informe nace como una evidencia del trabajo realizado en la práctica de laboratorio consistió en el diseño e implementación de un control anticipativo por medio de un controlador diseñado por medio del lugar geométrico de las raíces, complementado por un predictor Smith, para hacerlo más robusto a perturbaciones y lograr mantener la estabilidad del sistema.

Se optó por el desarrollo de un controlador PI, ya que se quería lograr el seguimiento de referencias y además establecer un error estacionario cercano a cero, por lo que al ser un sistema que controla variables como flujo y presión, muchas veces no se requiere que sea tan rápido, por lo que con las componentes proporcional e integral se cumplían los requerimientos.

Primeramente, se realizó el desarrollo en Matlab, donde se llevó a cabo la identificación del modelo, la simulación del controlador con su predictor Smith en el dominio continuo, discreto y ecuación en diferencias para luego pasar a su implementación en PLC, con ayuda del entorno de desarrollo TIA Portal se implementó el controlador, encontrando unos controladores que seguían referencias y sin

error estado estacionario, teniendo en cuenta mayores aspectos de la planta. Sin embargo, se logró un paso aceptable de la simulación a la vida real, dando por cumplidos los objetivos de la práctica.

II. INTRODUCCIÓN

El control anticipativo usa la medida de la perturbación para actuar sobre la planta compensándola. Diseñado para compensar el efecto de $G_D(s)$, además es un control en bucle abierto, por lo que debe utilizarse siempre junto con un control por realimentación de la salida (C.R.B.), para mejorar las características de éste.

A fin de compensar los errores en el modelo de la planta y la perturbación el controlador anticipativo se incluye en un lazo de realimentación:

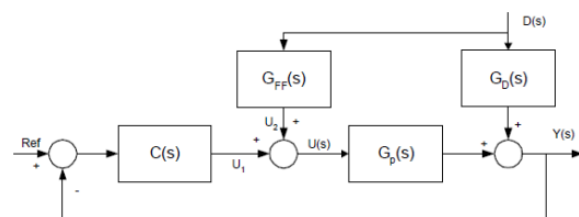


Figura 1 Estructura de un control anticipativo

El predictor de Smith es una estrategia de control, que permite complementar los controles tradicionales para hacerlo más

robusto ante perturbaciones externas, se compone por una configuración distinta dentro de su diagrama de bloques, es de gran importancia en sistemas con tiempo de retardo, donde poseen un tiempo muerto donde su comportamiento no puede ser descrito por una ecuación de un sistema lineal, por lo que el predictor intenta acoplar el comportamiento del sistema en el tiempo conocido, dentro de su tiempo muerto, para así lograr una mejor caracterización de la dinámica del sistema, es por esto que trabaja tanto con el retardo del sistema como sin él, es de ahí que se resalta la importancia de las cualidades de la implementación de esta configuración de control, ya que existen en la industria muchos sistemas que poseen este tipo de retardos.

Para el módulo de procesos de la Universidad de la Salle, esto no es la excepción, ya que el rango de operación del sistema no arranca de una entrada en cero, sino que tiene límite superior en inferior, por lo que existe un lapso de tiempo donde el sistema recibe una entrada pero no genera una salida, generando así un tiempo muerto que luego para el diseño de controladores puede llegar a hacer más complejo su desarrollo, por lo que una herramienta como el predictor de Smith son de gran utilidad para este tipo de sistemas.

El PI por lugar de las raíces es diseñado teniendo en cuenta el trasciente del sistema, por lo que se hace necesario conocer su respuesta en el tiempo, para luego establecer unas condiciones de error en estado estacionario e ir acoplando la dinámica para lograr ese comportamiento.

OBJETIVOS

Diseñar un controlador PI de presión y de flujo para el módulo de procesos.

Implementar un control anticipativo en conjunto con un predictor de Smith a la técnica de control.

III. METODOLOGÍA

Se desea implementar un controlador de Flujo y Presión en la unidad de entrenamiento en Automatización proporcionada por la Universidad de la Salle.

Dicha unidad cuenta con los siguientes elementos:

- Una bomba controlada mediante un variador de frecuencia. (MB01)
- 2 tanques de almacenamiento. (TK01, TK02)
- 4 electroválvulas. (VA11, VA21, VA12, VA22)
- 1 válvula proporcional (VC01).
- Transmisor de nivel.
- Transmisor de presión (TP01)
- Transmisor de flujo (TF01).

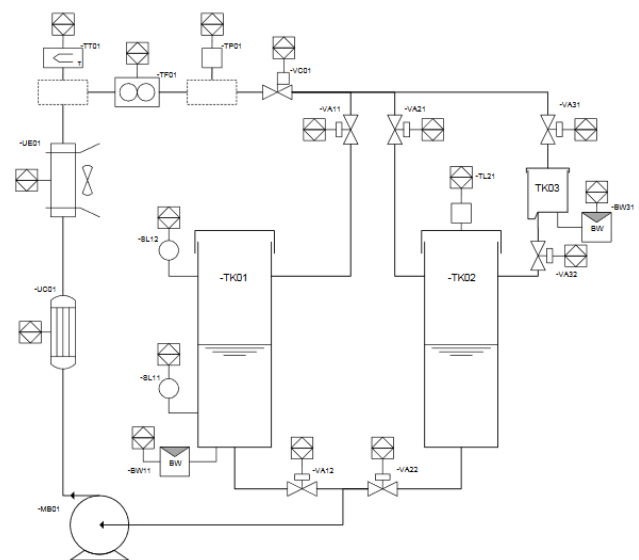


Figura 2 Planta de estudio

Como se describió anteriormente, la unidad de entrenamiento cuenta con un transmisor de presión y otro de flujo, los cuales miden dichas variables a medida que el fluido circula por el sistema. La apertura y cierre de las válvulas pueden modificar el comportamiento de la presión y el flujo. Sin embargo, para este caso se desea trabajar con todas las válvulas abiertas, y excitar el sistema mediante el variador que controla a la bomba.

Para lograr los objetivos propuestos, se realizaron las siguientes actividades:

Actividad 1

Poner en funcionamiento la unidad de entrenamiento mediante el uso del PLC 1500 de Siemens con el que cuenta, dejando todas las válvulas abiertas para que el fluido pueda circular fácilmente por el sistema. Posteriormente, realizar una conexión OPC entre TIA Portal y Matlab mediante el uso de KepServer y tomar medidas de flujo y presión para el sistema.

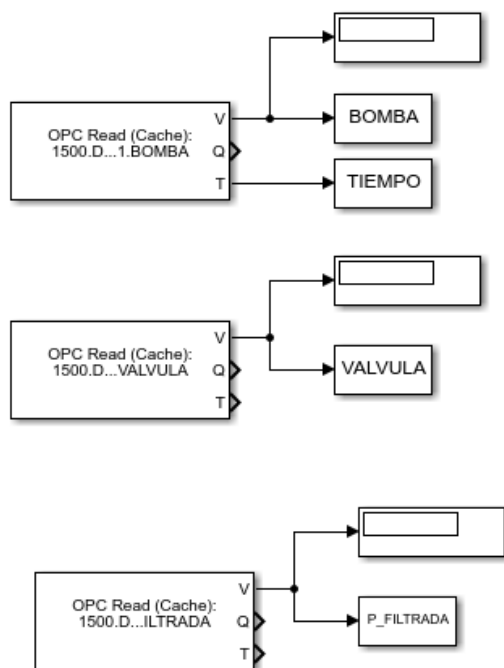


Figura 3 Bloque simulink para la adquisición de datos en Matlab

Actividad 2:

Realizar el respectivo tratamiento de los vectores de datos registrados mediante la conexión OPC entre Matlab y TIA para el tiempo, flujo y presión. Es decir, enfocar el análisis sobre la ventana de tiempo en que se excito el sistema hasta que se estabilizo.

Posteriormente, se deben identificar ambos sistemas y construir un modelo de primer

orden más tiempo muerto tanto para la presión como para el flujo.

Actividad 3:

Diseñar un controlador predictivo Smith. Para ello se diseña un controlador PI para la planta sin considerar el tiempo muerto y luego, se construye la estructura Smith teniendo en cuenta: la planta más tiempo muerto y el controlador PI diseñado.

Control PI para Flujo (Teorema del límite):

Se requiere que el sistema siga referencias con error en estado estacionario de cero. Es por ello que se recurre al diseño de un controlador PI teniendo las siguientes consideraciones en cuenta:

$$G_c = K_p + \frac{K_i}{s} = \frac{K_p * s + K_i}{s} = \frac{K_p \left(s + \frac{K_i}{K_p} \right)}{s}$$

$$G_c = \frac{K_p * (s + a)}{s}$$

Donde:

K_p : es la ganancia proporcional del controlador.

a : es el cero del controlador ubicado en $-K_i/K_p$. además, que posee un polo en el origen.

Para calcular K_p se puede utilizar el teorema del límite final de la siguiente forma:

$$K = \lim_{s \rightarrow 0} (K_p * G_p) = \lim_{s \rightarrow 0} \left(\frac{K_p * a1}{s + b1} \right) = \frac{K_p * a1}{b1}$$

$$essp = \frac{1}{1 + K} = \frac{b1}{b1 + a1 * K_p}$$

$$K_p = \frac{b1 * (1 - essp)}{essp * a1}$$

Donde:

G_p es la función de transferencia del sistema.

$Essp$ es el error de posición deseado, en este caso cero.

K es la constante de posición.

a1 y b1: Son constantes conocidas dadas por el modelo de identificación.

Una vez calculada la ganancia Kp que permite seguir referencias en el controlador PI, se debe calcular la ganancia Ki. Para ello se usará la teoría del diseño por el lugar geométrico de las raíces, teniendo en cuenta que el controlador PI ubica un polo en el origen y un cero muy cercano al origen según lo permita el sistema. Este cero viene dado por la proporción:

$$\frac{Ki}{Kp} = a$$

Es decir, que se puede calcular el valor de Ki, teniendo la constante Kp y el cero (a) del controlador o también, se puede calcular la ubicación del cero (a) teniendo las constantes del controlador PI.

La otra alternativa es utilizar el teorema del límite para velocidad, según la siguiente expresión:

$$Gcl = Gc * Gp = \left(Kp + \frac{Ki}{s} \right) * \left(\frac{a1}{s + b1} \right)$$

Reemplazando la función en lazo cerrado en el teorema del límite:

$$\begin{aligned} Kv &= \lim_{s \rightarrow 0} s * (Gcl) = \\ &= \lim_{s \rightarrow 0} s * \left(\frac{(Kp * s + Ki)(a1 * Kp)}{s(s + b1)} \right) = \\ &= \frac{Ki * Kp * a1}{b1} \end{aligned}$$

$$Kv = Ki * Kp * c1 \quad (\text{donde: } c1 = a1/b1)$$

Reemplazando Kv en la expresión del error para velocidad y despejando Ki:

$$essv = \frac{1}{Kv} = \frac{1}{Ki * Kp * c1}$$

$$Ki = \frac{1}{Kp * c1 * essv}$$

Gcl es el sistema en lazo cerrado entre el controlador PI y la planta.

Kp y Ki son las constantes del controlador.

Essv es el error deseado en estado estacionario.

C1 es conocido ya que es la división entre a1 sobre b1.

De esta forma, la expresión 4 y 5 describen la forma en que se puede calcular las constante Kp y Ki del controlador conociendo la función de transferencia de la planta y el error deseado en estado estacionario.

Control PI para Presión (Asignación de polos):

La metodología anterior, se usará para diseñar el controlador PI del Flujo, sin embargo, para el de Presión se optó por diseñar un controlador PI por asignación de polos.

Para diseñar un controlador por este método, se parte de las siguientes consideraciones:

$$Gp = \frac{a1}{s + b1}$$

Donde:

Gp es la función de transferencia de la planta.

a1 es la ganancia de la planta en lazo abierto.

b1 es el polo de la planta en lazo abierto.

Se desea que la planta en lazo cerrado se comporte de acuerdo con unos criterios de diseño: Tiempo de establecimiento (Tss) y Overshoot (Os). Asignando estos criterios se puede calcular la constante de amortiguamiento y la frecuencia deseada para el sistema:

$$\xi = \frac{\ln\left(\frac{\%Os}{100}\right)^2}{\sqrt{\pi^2 + \ln\left(\frac{\%Os}{100}\right)^2}}$$

$$\omega_n = \frac{3}{\xi * T_{ss}}$$

Donde:

ξ es la constante de amortiguamiento.

ω_n es la frecuencia del sistema.

%os es el porcentaje de overshoot.

Tss es el tiempo de establecimiento.

Estos criterios de diseño construyen la ecuación característica del controlador para que el sistema se comporte como se desea en lazo cerrado. Esta ecuación está dada así:

$$G_c = s^2 + (2 * \xi * \omega_n) * s + \omega_n^2$$

La expresión anterior, expresa la ubicación de los polos en lazo cerrado para el sistema controlado mediante una función cuadrática, de la siguiente forma:

$$ax^2 + bx + c = 0$$

$$p_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

La ecuación característica deseada será comparada con la ecuación en lazo cerrado, es decir, controlador más planta, de la siguiente forma:

Planta:

$$G_p = \frac{a_1}{s + b_1} = \frac{A}{B}$$

Controlador:

$$G_c = \frac{kp * T_d * s^2 + Kp * s + \frac{Kp}{Ti}}{s} = \frac{d_2 * s^2 + d_1 * s + d_0}{s} = \frac{D}{E}$$

Donde:

T_d es la constante derivativa = 0.

Ecuación en lazo cerrado (H):

$$H(s) = \frac{DA}{EB + DA} =$$

$$= \frac{a_1(d_2 * s^2 + d_1 * s + d_0)}{s(s + b_1) + a_1(d_2 * s^2 + d_1 * s + d_0)}$$

$H(s)$

$$= \frac{a_1(d_2 * s^2 + d_1 * s + d_0)}{s^2(1 + a_1 * d_2) + s(b_1 + a_1 * d_1) + a_1 * d_0}$$

La ecuación en lazo cerrado característica es:

$$s^2(1 + a_1 * d_2) + s(b_1 + a_1 * d_1) + a_1 * d_0$$

Cálculo de las constantes Kp y Ti:

La expresión anterior se compara con la ecuación deseada y se despejan las expresiones para cada constante del controlador (Kp, Ti, Td):

$$s^2(1 + a_1 * d_2) + s(b_1 + a_1 * d_1) + a_1 * d_0 =$$

$$= s^2 + (2 * \xi * \omega_n) * s + \omega_n^2$$

Entonces:

$$1 = (1 + a_1 * d_2) = (1 + a_1 * Kp * T_d)$$

$$(2 * \xi * \omega_n) = (b_1 + a_1 * d_1) = (b_1 + a_1 * Kp)$$

$$\omega_n^2 = a_1 * d_0 = a_1 * \frac{Kp}{Ti}$$

Sabiendo que: $T_d=0$, y usando las últimas dos expresiones anteriores, se encuentra que las expresiones para las constantes son:

$$Kp = \frac{(2 * \xi * \omega_n) - b_1}{a_1}; \quad Ti = \frac{a_1 * Kp}{\omega_n^2}$$

Al diseñar el controlador PI para cada sistema, se construirá el controlador predictor Smith teniendo en cuenta la planta más tiempo muerto del sistema:

$$G_{pt} = \frac{e^{(-tm)*s} a_1}{s + b_1}$$

Actividad 4:

Discretizar el controlador Smith y expresarlo en ecuaciones en diferencias para su implementación en un controlador lógico programable que controle el flujo y presión dentro de la unidad de entrenamiento en automatización.

Para la discretización se debe seleccionar un tiempo de muestreo según los siguientes criterios:

$$\frac{Tr}{20} < Tm < \frac{Tr}{10}$$
$$\frac{Tss}{75} < Tm < \frac{Tss}{25}$$

Donde:

Tr es el tiempo de subida en lazo abierto.

Tss tiempo de establecimiento en lazo abierto.

Tm es el tiempo de muestreo.

Se puede usar cualquiera de los dos criterios y discretizar la ecuación de la planta y el controlador usando Tustin.

Posteriormente se expresa ambas ecuaciones discretizadas en ecuación en diferencias y se implementa en el PLC usando variables, de la siguiente forma:

Implementación en PLC:

Controlador de Flujo

Para el controlador de flujo y presión para el módulo de procesos de la Universidad de la Salle se utilizó el entorno de programación TIA Portal de Siemens, ya que el módulo está controlado por un PLC 1500 de Siemens, por lo que, para la implementación de controladores externos, se debe crear un bloque de funciones (FB), para desarrollar ahí la ecuación en diferencias y utilizar las señales y variables globales del sistema.

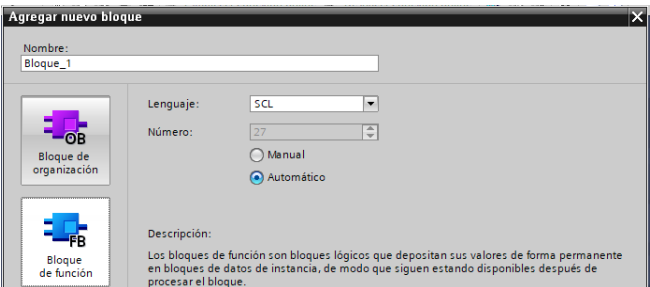


Figura 4 Creación de nuevo bloque en TIA Portal

Como lo muestra la Figura 4, existen diversos bloques de programa los cuales están dados por la norma IEC-61133-3, por lo que este entorno de desarrollo está basado en la norma y listo para su ejecución, en este caso se creara un bloque de función, el cual utiliza unas entradas y salidas y el genera un procesamiento interno para la ejecución de la tarea.

	Name	Data type
Input		
Initial_Call		Bool
Event_Count		Int
Temp		
PIDDT		Bool

Figura 5 Entradas y salidas del bloque de función para el controlador de flujo

En la Figura 5 se evidencia como se hace la asignación de las entradas y salidas del bloque de función, en este apartado seleccionamos el nombre de la variable para uso interno, no significa que este sea el nombre de la variable global, ya que esa asignación se realiza de manera diferente, además de selección el tipo de variable y un valor inicial que será el primer valor en el primer ciclo scan o de interrupción del bloque.

Las demás variables pueden ser llamadas como estáticas o temporales, ya que son usadas solo en el funcionamiento del bloque, por lo que no afectan variables globales de manera directa.

En la Figura 4 se evidencia que el lenguaje de programación escogido es el SCL, conocido como texto estructurado, el cual consiste en una programación por código de las funciones del PLC, es ideal para la implementación de controladores y de procesos complejos y matemáticos, por lo que nuestros bloques fueron programados en este lenguaje.

Figura 6 Ecuaciones de diferencia predictor de Smith implementada en en Tia Porta.

Primeramente, se realizó el cálculo del error, que corresponde a la resta entre la referencia y la salida de la planta, que en este caso es la medida del sensor, luego se implementa la ecuación en diferencias del sistema, donde los coeficientes fueron declarados en un vector para su cambio rápido. Luego se implementa un saturador para la seguridad del sistema, donde se asegura que la salida del controlador no sobrepase los valores máximos y mínimos posibles por los equipos del sistema, para que así cuando se realice un cálculo que salga de estos valores, no entre en error el PLC.

```
36 #e4 := #e3;
37 #e3 := #e2;
38 #e2 := #e1;
39 #e1 := #e;
40
41 #e4_out_1 := #e4;
42 #e3_out_1 := #e3;
43 #e2_out_1 := #e2;
44 #e1_out_1 := #e;
45
46 #u4 := #u3;
47 #u3 := #u2;
48 #u2 := #u1;
49 #u1 := #baux;
50 #u4_out_1 := #u4;
51 #u3_out_1 := #u3;
52 #u2_out_1 := #u2;
53 #u1_out_1 := #u1;
```

Figura 7 Actualización de variables del corrimiento

El último paso es realizar la actualización de las variables para generar ese corrimiento dentro de los valores y así poder tener valores pasados del sistema para el desarrollo del controlador, por lo que para este caso se actualizan los valores de salida y los del error.

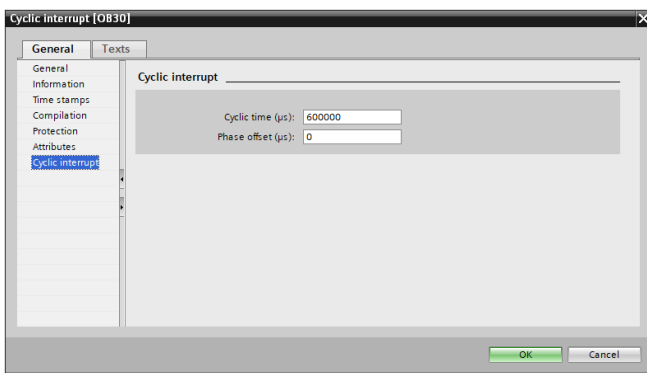


Figura 8 Configuración ciclo de interrupción

Los bloques OB30 en Siemens corresponden a bloques de interrupción, los cuales se ejecutan y envían datos en un tiempo cíclico, en este caso se debe poner el tiempo de diseño del controlador, es decir debe ser igual al tiempo de muestreo para la discretización del controlador en el proceso de diseño, para generar una salida equivalente en una implementación real.

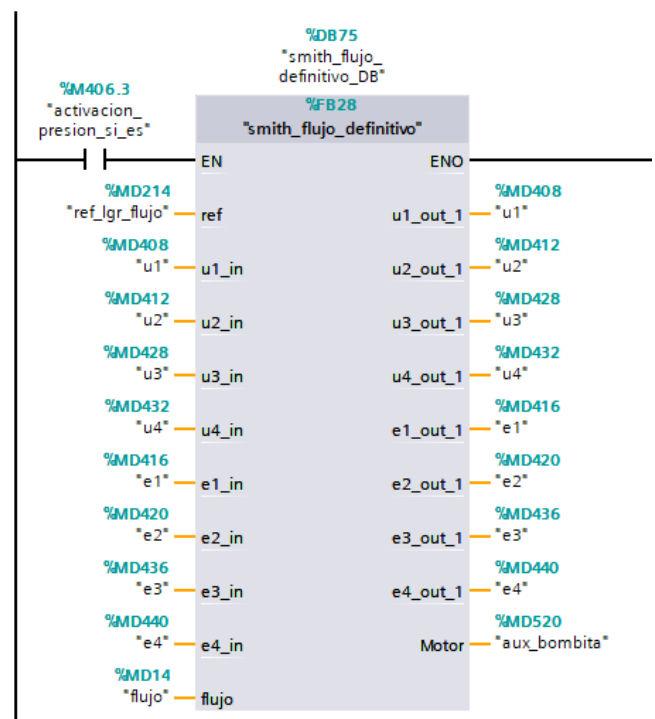


Figura 9 Llamado del bloque en el ciclo de interrupción

Como se muestra en la Figura 9, el llamado del bloque de función es mucho más gráfico, es donde se asignan las variables globales a las preestablecidas internamente del bloque, en este caso se utiliza la referencia y el sensor, para mandar la señal de control a la bomba que controla el motor y permite el flujo dentro del módulo, además al inicio un contacto normalmente abierto, para poder activar y desactivar el controlador y así poder evidenciar su funcionamiento normal.

A continuación se presenta el diseño de controlador anticipativo para flujo.

```

kpPF = 0.008857;
kp = 0.02167;
KFF=kpPF/kp;
tm = 0.853;
tmPF = 1.25;
tau = 5.628;
tauPF = 8.245;

gamma=tmPF-tm;
GFF=-((KFF*(tau*s+1))/(tauPF*s+1))*exp(-gamma*s);

```

Figura 10 Configuración del control anticipativo de flujo.

GFF =

$$\exp(-0.397*s) * \frac{-2.3 \text{ s} - 0.4087}{8.245 \text{ s} + 1}$$

Figura 11 Resultado del diseño de control anticipativo de flujo.

Perturbacion_diseno =

$$\frac{0.279 \text{ z}^2 - 0.8073 \text{ z} + 0.501}{\text{z}^2 - 0.9785 \text{ z} + 0.04526}$$

Figura 12 Resultado de funcion de transferencia discreta para el control anticipativo de flujo.

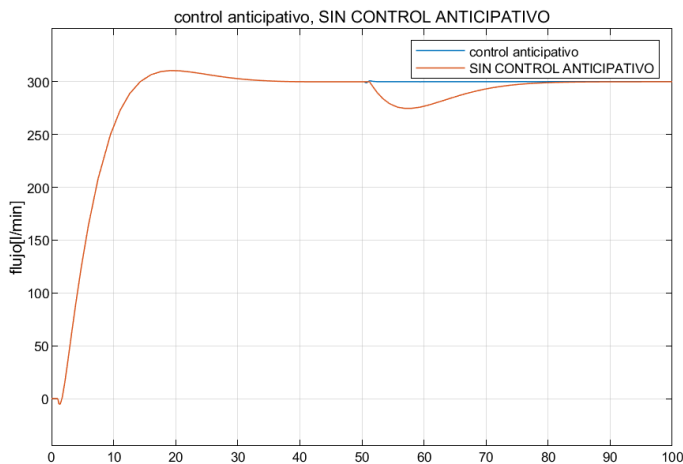


Figura 13 Simulación de anticipativo de flujo vs Smith.

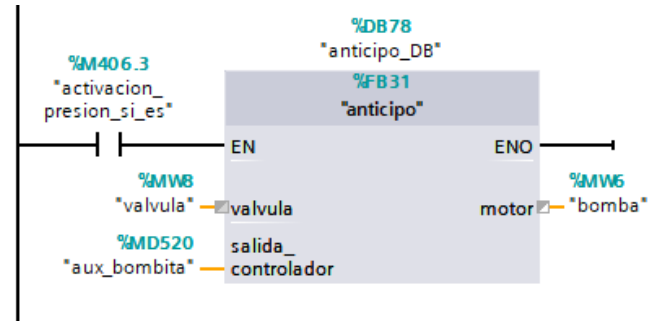


Figura 14 Llamado del bloque anticipativo para flujo

```

1 #entrada := #valvula - 27000;
2 IF #entrada<27000 THEN
3 // Statement section IF
4 #salida := 0.279*#entrada-0.8073*#entradal+0.5010*#entrada2 +0.9785*#salidal-0.0453*#salida2;
5 #baux := #salida_controlador + #salida;
6 ELSE
7 #baux := #salida_controlador;
8 END_IF;
9
10 IF #baux > 27000.0 THEN
11 #motor := 27000.0;
12 ELSIF #baux < 6500.0 THEN
13 #motor := 6500.0;
14 ELSE
15 #motor := #baux;
16 END_IF;
17 #entrada2 := #entradal;
18 #entradal := #entrada;
19
20 #salida2 := #salidal;
21 #salidal := #salida;

```

Figura 15 Código con las constantes para el control anticipativo de flujo

Controlador de Presión

A continuación, se presenta el calculo para el control anticipativo.

```

kpPF = 0.008857;
kp = 0.02167;
KFF=kpPF/kp;
tm = 0.853;
tmPF = 1.25;
tau = 5.628;
tauPF = 8.245;

gamma=tmPF-tm;
GFF=-((KFF*(tau*s+1))/(tauPF*s+1))*exp(-gamma*s);

```

Figura 16 Calculo del control anticipativo de presión.

Al ejecutar las líneas de código de la Figura 16 se consiguió la siguiente ecuación de diferencia.

GFF =

$$\exp(-0.397*s) * \frac{-2.3 \text{ s} - 0.4087}{8.245 \text{ s} + 1}$$

Figura 17 Resultado del control anticipativo para presión.

$$\frac{0.279 z^2 - 0.8073 z + 0.501}{z^2 - 0.9785 z + 0.04526}$$

Figura 18 Discretization de control anticipativo de presión para un tiempo de muestreo de 0.6 segundos.

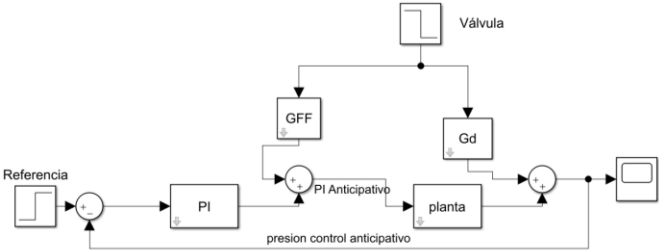


Figura 19 Estructura Smith implementada para presión

A continuación, en la Figura 20 se presenta las simulaciones correspondientes al controlador Smith y el anticipativo Smith implementado.

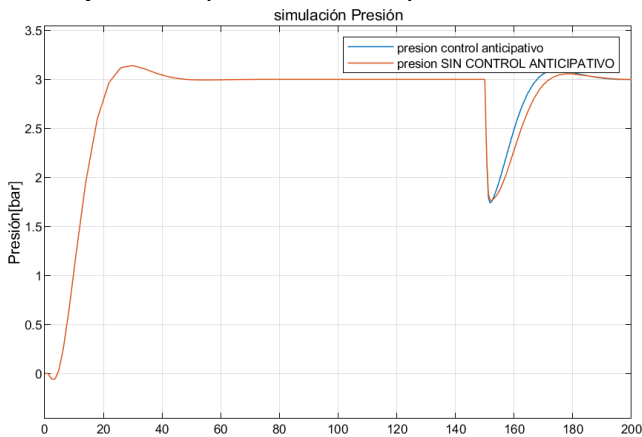


Figura 20 Simulación de control anticipativo de presión.

Para la presión el proceso es análogo, solo que ahora se cambia la ecuación en diferencias y se crea en otro bloque FB, es ahí donde se crea un proceso estándar donde solo cambia los coeficientes del controlador y el orden de este, esto permite generar controladores genéricos y que pueden ser utilizados en varias plantas.

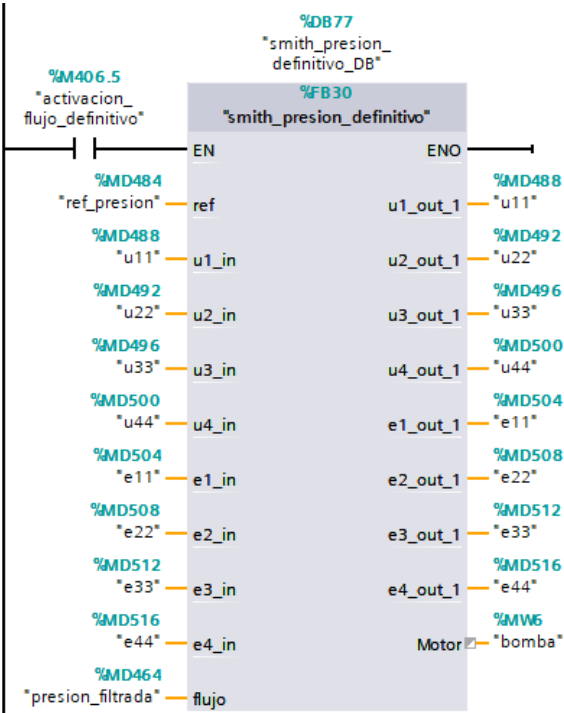


Figura 21 Llamado del bloque en el ciclo de interrupción para presión

En la Figura 21 es mostrado el llamado del bloque con el controlador de control, en este caso las variables son de presión, por lo que el sistema ya es cambiado para genera el control de esta variable, pero los demás procesos son iguales que para la implementación del controlador de flujo.

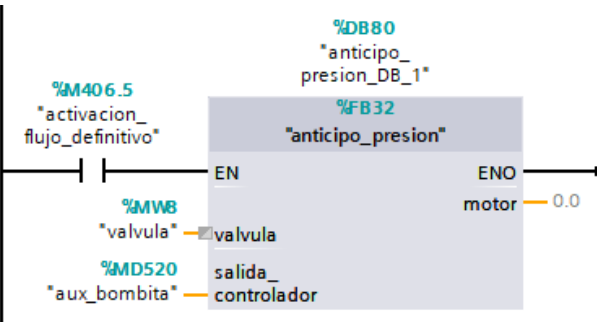


Figura 22 Llamado del bloque en el ciclo de interrupción para presión

```

1 #entrada := #valvula - 27000;
2 //4.847z2 - 9.84z + 4.884
3 //z2 - 1.504z + 0.5546
4 IF #entrada<27000 THEN
5   // Statement section IF
6   #salida := 4.847*#entrada-9.84*#entradas1+4.884*#entrada2 +1.504*#salida1-0.5546*#salida2;
7   #baux := #salida_controlador + #salida;
8 ELSE
9   #baux := #salida_controlador;
10 END_IF;
11
12 IF #baux > 27000.0 THEN
13   #motor := 27000.0;
14 ELSIF #baux < 6500.0 THEN
15   #motor := 6500.0;
16 ELSE
17   #motor := #baux;
18 END_IF;
19 #entrada2 := #entradas1;
20 #entradas1 := #entrada;
21
22 #salida2 := #salida1;
23 #salida1 := #salida;

```

Figura 23 Código con las constantes para el control anticipativo de presión

Actividad 5: Realizar pruebas y correcciones del controlador sobre el proceso.

IV. ANÁLISIS DE RESULTADOS:

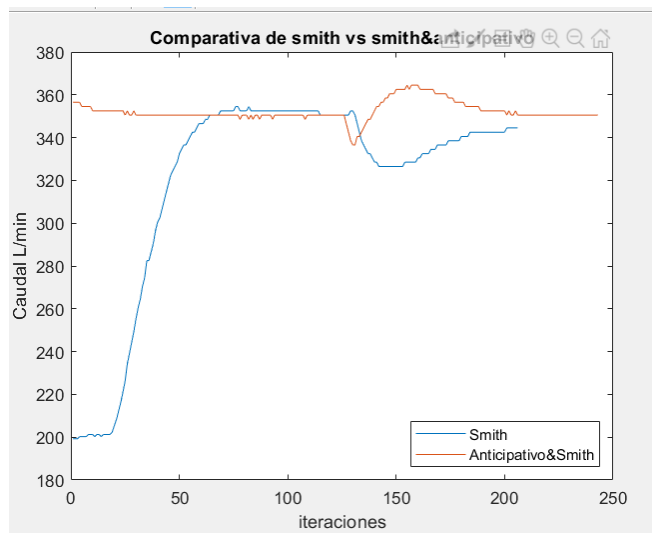


Figura 24 Comparativa Smith y Smith con control anticipativo ante la misma perturbación.

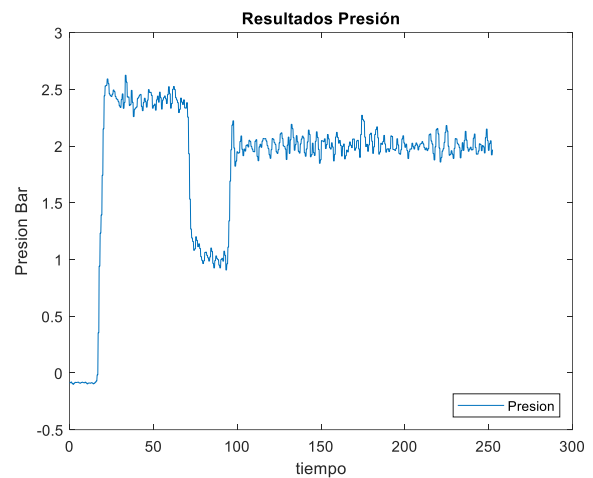


Figura 25 Resultados de la presión una vez implementado el control

V. CONCLUSIONES

- El anticipativo de presión funciona correctamente aumentando la salida de la bomba proporcional al modelo, para el caso de la válvula en un cierre de 22000 le aumenta 3000 a la consigna de la bomba.
- Es importante al momento de simular el diseño del controlador visualizar la respuesta del controlador que no sea brusca de tal forma de que no se active el saturador.
- El control anticipativo se puede integrar con estrategias como el control Smith, de tal forma de que se vuelva un sistema mas robusto ante las perturbaciones.
- En la Figura 24 se puede evidenciar la diferencia de implementar un control anticipativo, para este caso en particular el anticipativo reduce más rápido.

VI. REFERENCIAS

- [1] Víctor M. Alfaro Ruiz (2002). Métodos de sintonización de controladores PID que operan como reguladores. Ingeniería: Revista de la Universidad de Costa Rica. Vol. 12., 17 páginas.