

# CONTROLADOR DE MODOS DESLIZANTES BASADO EN PREDICTOR DE SMITH Y MODELO DE SEGUNDO ORDEN PARA SISTEMA DE FLUJO EN EL MODULO DE PROCESOS

**Daniel Emanuel Ramírez Junco<sup>1</sup>, Rodrigo Harley Jimenez Fuentes<sup>1</sup>**

<sup>1</sup> Univ. de la Salle, ZIP-110911 Bogotá, Colombia

Rjimenez00@unisalle.edu.co: Danieleramirez12@unisalle.edu.co

**Resumen:** En este artículo se presenta el desarrollo y funcionamiento de un controlador de modos deslizantes basado en un Predictor de Smith para el módulo de procesos de la Universidad de la Salle, donde a partir de la identificación de segundo orden más tiempo muerto (SOMTM) para el sistema de Flujo y una superficie deslizante de tipo PID, se presenta el modelo y desarrollo del controlador, se compara el comportamiento del sistema con este controlador respecto a dos métodos de control, control en cascada y control anticipativo diseñados para el mismo sistema de flujo.

**Palabras Clave:** Control, Sliding Modes, Predictor Smith.

**Abstract:** This article presents the development and operation of a sliding mode controller based on a Smith Predictor for the processes module of the University of La Salle, where from the identification of the second order plus dead time (SOMTM) for the Flow system and a PID-type sliding surface, the model and development of the controller are presented, the behavior of the system with this controller is compared with respect to two control methods, cascade control and anticipatory control designed for the same flow system.

**Keywords:** Control, Sliding Modes, Predictor Smith.

## 1. Introducción

En el diseño de controladores muchas veces se encuentran problemas relacionados con la no-exactitud del modelo de la planta, las perturbaciones externas o la existencia de no linealidades en el proceso. Una de las no-linealidades más comunes es el tiempo muerto o retardo de transporte, que afecta el comportamiento de la planta, haciendo que su análisis sea más complicado. Este retardo puede ser causado por el transporte de masa o energía de un lugar a otro, tiempos de procesamiento o acumulación de pequeños retardos en sistemas conectados en serie, entre otros. Para los procesos que exhiben tiempo muerto, cada acción ejecutada en la variable manipulada del proceso sólo afectará a la variable controlada después del tiempo muerto del proceso. Debido a esto, es más difícil de analizar y diseñar controladores para sistemas de tiempo muerto. [1]

Las técnicas de control clásicas, basadas en ubicar los polos en una posición deseada para cumplir con ciertas características de diseño, son usadas en muchos campos de investigación permitiendo obtener buenos resultados en sistemas con perturbaciones y tiempo muerto. En la actualidad se han hecho propuestas para el análisis, diseño e implementación de sistemas que permitan realizar esquemas y estrategias de control que afronten la no-linealidad del tiempo muerto y dar mejoras a las respuestas de dichos

sistemas, además de darle un mejor desempeño y robustez ante perturbaciones en el proceso, algunas de estas estrategias se le conocen como compensadores de tiempo muerto.

El predictor de Smith es una estrategia de control, que permite complementar los controladores tradicionales para hacerlo más robusto ante perturbaciones externas y los retrasos del sistema, tiene una configuración propia en su diagrama de bloques, donde poseen un tiempo muerto dado el comportamiento real de una planta que presenta retardos en su medición de variables, por lo que el predictor intenta acoplar el comportamiento del sistema en el tiempo conocido, con 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.

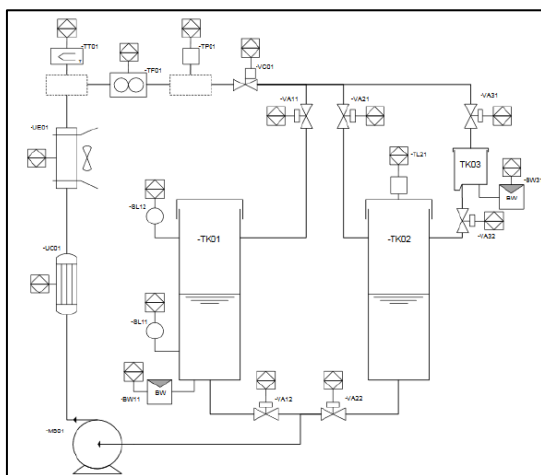
El control de modo deslizante (Sliding Mode Control o SMC) es un procedimiento robusto y simple para desarrollar controladores de procesos lineales y no lineales. La idea detrás del control de modo deslizante consiste en definir una superficie (superficie deslizante) a lo largo de la cual el proceso pueda deslizarse hacia su valor final deseado. Esto se logra por medio de una ley de control que forma parte del diseño de un

controlador de modo deslizante (Sliding Mode Controller) o SMCr. [1]

## 2. Módulo de procesos.

Este artículo tiene como finalidad implementar un controlador de Flujo en la unidad del módulo de procesos de la Universidad de la Salle, basado en un predictor Smith y modos deslizantes.

La figura 1 muestra la representación y ubicación de los elementos que se encuentran en el módulo de procesos.



**Figura 1. Módulo de procesos.**

Este módulo cuenta con un transmisor de flujo (TF01) y presión (TP01), una bomba controlada mediante un variador de frecuencia (MB01), dos tanques de almacenamiento (TK01, TK02), cuatro electroválvulas (VA11, VA21, VA12, VA22), una válvula proporcional (VC01), un transmisor de nivel.

Mediante un PLC S7 1500, permite implementar diferentes tipos de control para sistemas de flujo y presión, midiendo el comportamiento de dichas variables, con lo cual permite realizar la identificación de estos sistemas, además permite la apertura y cierre de las válvulas que pueden modificar el comportamiento de la planta, por lo que el control que se implementa debe ser robusto ante las perturbaciones de este tipo y corrija el tiempo muerto.

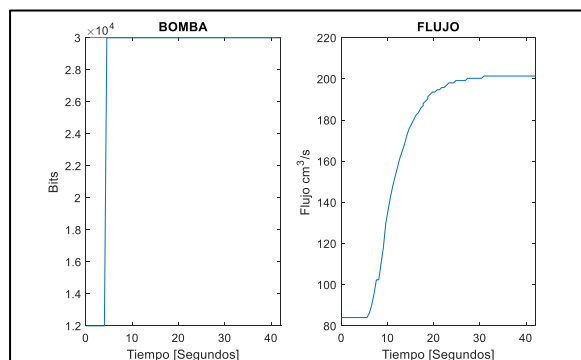
La metodología de validación de este tipo controlador SMCr consistirá en controlar el sistema de flujo a diferentes referencias y la perturbación de la válvula proporcional, este tipo de control se comparará con un control en cascada y control anticipativo diseñados para este mismo sistema.

## 3. Identificación de Segundo Orden.

Para diseñar o sintonizar controladores, es necesario contar con información del comportamiento dinámico del proceso, normalmente en forma de un modelo de orden reducido (primero o segundo orden, más tiempo muerto). El método de Stark considera de tres puntos, los cuales corresponden a los tiempos para que la respuesta del sistema alcance 15% ( $t_{15}$ ), 45% ( $t_{45}$ ) y 75% ( $t_{75}$ ) del cambio en la salida del sistema ( $\Delta y$ ) a un cambio escalón en la entrada ( $\Delta u$ ), para obtener un modelo o de segundo orden más tiempo muerto (SOMTM) de la forma:

$$\frac{Y(s)}{M(s)} = \frac{K \cdot w_n^2}{s^2 + 2 \cdot \zeta \cdot w_n \cdot s + w_n^2} \cdot e^{-t_0 \cdot s}$$

De esta forma primero se almacenan datos experimentales en los cuales se tiene una curva de reacción del sistema ante una entrada tipo escalón, como se ve a continuación:



**Figura 2. Tomados de la planta real.**

Una vez se toman estos datos experimentales, para hacer la debida interpolación de los tiempos en los que la reacción del sistema alcanza el 15, 45 y 75% del valor final.

```
clc; clear all; close all;
%% Datos tomados del módulo
load('datos_l2_a_30.mat')
s=tf('s'); %S es ahora de una TF
%% Lectura de datos
t = TIEMPO(1:94,1);
y=FUJO(165:258,1);
Sy=size(y);
u=BOMBA(165:258,1);
%% Identificación de planta de segundo orden más tiempo muerto
dy=max(y)-min(y);
du=max(u)-min(u);
dy15=dy*0.15+min(y);
dy45=dy*0.45+min(y);
dy75=dy*0.75+min(y);
vdy=[dy15; dy45; dy75];
vdt= [Funcion(y,t,dy15) Funcion(y,t,dy45) Funcion(y,t,dy75)]';
kp=dy/du;
```

**Figura 3. Interpolación de los tres puntos sobre la curva de reacción.**

Para llevar a cabo la interpolación de los tiempos deseados, se diseñó la siguiente función:

```
%% Función de interpolación
function res = Funcion(v1, v2, valor)
for k=1:length(v1)
    Aux1=v1(k);
    if Aux1>valor
        res=v2(k-1)+((v2(k)-v2(k-1))/(v1(k)-v1(k-1)))*(valor-v1(k-1));
        break
    end
end
end
```

**Figura 4. Función de interpolación.**

Quando ya se tienen las parejas de valores de los tres puntos sobre la curva de reacción, se procede a hacer el cálculo de las constantes de ganancia (kp), zeta ( $\zeta$ ) y la frecuencia natural ( $w_n$ ), según el método de Stark:

$$\zeta = \frac{0.0805 - 5.547 \cdot (0.475 - x)^2}{x - 0.356}$$

Donde x:

$$x = \frac{t_{45} - t_{15}}{t_{75} - t_{15}}$$

```
x=(vdt(2)-vdt(1))/(vdt(3)-vdt(1));
%% Zeta
zeta=(0.0805-5.547*(0.475-x)^2)/(x-0.356);
```

**Figura 5. Cálculo de Zeta.**

Según el método de stark, de acuerdo con el valor que resulte de zeta, se debe usar una determinada ecuación auxiliar para obtener las demás constantes faltantes.

$$f_2(\zeta) = 0.708 \cdot (2.811)^\zeta$$

$$f_2(\zeta) = 2.6 \cdot \zeta - 0.6$$

```
if zeta<=1
    f2= 0.708*(2.811)^zeta;
else
    f2= 2.6*zeta-0.6;
end
%% Wn
wn= f2/(vdt(3)-vdt(1));
%% tm - tiempo muerto (t0)
f3= 0.922*(1.66^zeta);
tm= vdt(2)-(f3/wn);
%% Identificación segundo orden mas tiempo muerto
G= (kp*(wn^2)*exp(-tm*s))/(s^2+2*zeta*wn*s+wn^2);
```

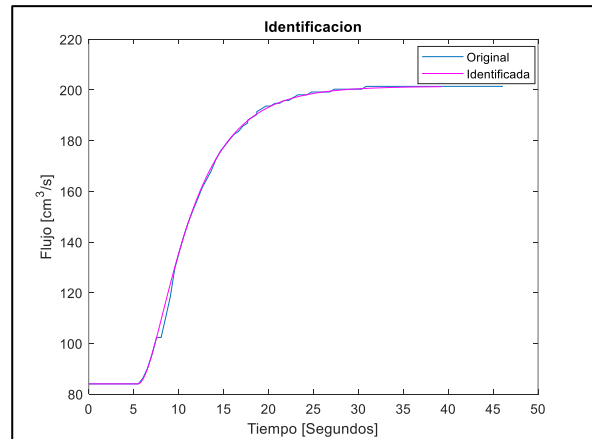
**Figura 6. Identificación de la planta.**

De esta forma se obtiene el modelo de identificación del proceso de flujo según el método de Stark:

```
G =
0.002869
exp(-5.48*s) * -----
s^2 + 0.7821 s + 0.1222
Continuous-time transfer function.
```

**Figura 7. Planta identificada.**

Con la cual se procede a comparar su respuesta ante un step equivalente al  $\Delta u$ , con lo cual se obtiene la siguiente gráfica:



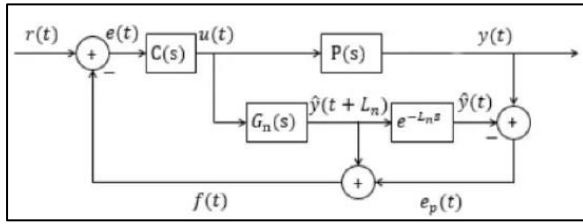
**Figura 8. Comparación de la identificación.**

Como se puede ver en la figura anterior, se obtiene un modelo que representa de una excelente forma la dinámica del proceso al cual se le va a implementar el control.

#### 4. Controlador en Modos Deslizantes basado en un Predictor Smith.

El controlador de modo deslizante basado en predictor de Smith (SPSMCr) propuesto usa la arquitectura estándar del SP (Smith Predictor) mientras que el controlador se basa en un controlador de modo deslizante (SMCr). Se obtiene así un controlador robusto y que compensa procesos con retardo elevado.

Varios trabajos presentan controladores de modo deslizante basado en predictor de Smith y todos ellos emplean un modelo POMTM para el desarrollo del SMCr. En este trabajo se utiliza un modelo SOMTM para desarrollar el controlador de modo deslizante basado en Predictor de Smith o SP2-SMCr. Para desarrollar este controlador se considera el diagrama de bloques del predictor de Smith mostrado en la siguiente figura, donde la función de transferencia del modelo P(s) corresponde al modelo ya identificado.



**Figura 9. Predictor Smith.**

Si no hay diferencia entre el proceso y el modelo, se puede tomar el modelo de la función de transferencia identificada sin el tiempo muerto.

$$G(s) = \frac{X_1(s)}{M(s)} = \frac{K_1 \cdot w_n^2}{s^2 + 2 \cdot \zeta \cdot w_n \cdot s + w_n^2} = \frac{K}{s^2 + a_1 \cdot s + a_2}$$

Donde  $K=K_1 w_n^2$ ,  $a_1=2 \zeta w_n$  y  $a_2=w_n^2$ . Ya que el tiempo muerto  $t_0$  ha sido aislado usando una estructura SP, se puede ignorar en el diseño del controlador, con la cual se transforma la ecuación anterior en forma de ecuación diferencial

$$\frac{d^2 X_1(t)}{dt^2} = -a_1 \cdot \frac{dx_1(t)}{dt} - a_2 \cdot x_1(t) + K \cdot m(t)$$

Según Slotine y Li, la superficie deslizante puede ser de la forma:

$$S(t) = \left( \frac{d}{dt} + \lambda \right)^n \int e(t) \cdot dt$$

donde  $e(t)$  es el error de seguimiento y el coeficiente  $\lambda$  es un parámetro de sintonización. Ya que la ecuación diferencial es una función de transferencia de segundo orden, entonces  $n = 2$  y  $S(t)$  puede ser expresada como

$$S(t) = \frac{de_1(t)}{dt} + \lambda_1 \cdot e_1(t) + \lambda_0 \cdot \int_0^t e_1(t) \cdot dt$$

Donde  $e_1(t)$  es la diferencia entre el valor de referencia o set-point,  $r(t)$ , y el valor de la salida de un modelo del proceso sin retardo,  $x_1(t)$ , cuando se considera un modelo perfecto, o sea,  $e_1(t) = r(t) - x_1(t)$ . La ecuación anterior representa una superficie PID, la cual se desea que  $\frac{dS(t)}{dt} = 0$ , y asumiendo un error de modelado igual a cero, se tiene:

$$\frac{d^2 r(t)}{dt^2} - \frac{d^2 x_1(t)}{dt^2} + \lambda_1 \cdot \frac{dr(t)}{dt} - \lambda_1 \cdot \frac{dx_1(t)}{dt} + \lambda_0 \cdot e_1(t) = 0$$

Descartando de la ecuación anterior las derivadas del valor de referencia  $r(t)$ , se llega a

$$\frac{d^2 x_1(t)}{dt^2} = -\lambda_1 \cdot \frac{dx_1(t)}{dt} + \lambda_0 \cdot e_1(t)$$

De esta forma, se iguala la ecuación anterior con la ecuación diferencial del modelo identificado, resultando así la siguiente ecuación de control para este controlador SP2-SMCr

$$m(t) = \frac{1}{K} \cdot \left[ (2 \cdot \zeta \cdot w_n) \cdot \frac{dx_1(t)}{dt} + (\lambda_0 \cdot e_1(t)) + (w_n^2 \cdot x_1(t)) \right] + K_D \cdot \frac{S(t)}{|S(t)| + \delta}$$

Donde:

$$S(t) = \text{sign}(k) \cdot \left[ -\frac{dx_1(t)}{dt} + \lambda_1 \cdot e_1(t) + \lambda_0 \cdot \int_0^t e_1(t) \cdot dt \right]$$

La función  $\text{sign}(K)$  en la ecuación anterior, fue incluida en la ecuación de la superficie deslizante para garantizar la acción apropiada del controlador para el sistema dado.

En la ecuación anterior se asegura una superficie deslizante críticamente o sobre amortiguada. Según las recomendaciones y con base a las simulaciones realizadas se seleccionaron

$$\lambda_1 = 2 \cdot \zeta \cdot w_n; \quad \lambda_0 = \frac{\lambda_1^2}{8}$$

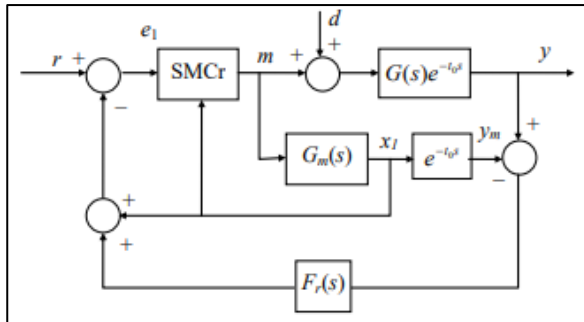
Los parámetros  $K_D$  y  $\delta$  tienen relación con la velocidad del sistema, el sobrepaso y el chattering. Basado en procedimientos previos donde se usó el algoritmo de búsqueda Nelder-Mead, los parámetros de ajuste de la parte discontinua del controlador son:

$$K_D = \frac{0.75}{|K|} \cdot \left( \frac{\lambda_1}{\lambda_0} \right)^{0.76}; \quad \delta = 0.68 + 0.12 \cdot (|K| \cdot K_D \cdot \lambda_1)$$

Los errores de modelado del tiempo muerto pueden llevar al SP a la inestabilidad. Una solución simple a este problema es usar un filtro  $F_f(s)$  de ganancia unitaria en el lazo de realimentación de la salida. Dicho filtro puede ser definido como el filtro de primer orden con un único parámetro  $T_f$  donde con  $T_f=0.5 t_0$ .

$$F_r(s) = \frac{1}{s \cdot T_f + 1}$$

De esta manera ya se tiene los elementos necesarios para armar la arquitectura del SP2-SMCr:



**Figura 10. Predictor Smith con modos deslizantes.**

Como ya se conocen las ecuaciones necesarias para hacer la sintonización, se puede realizar el cálculo del controlador, como en esta estructura el bloque del controlador tiene dos entradas, se decide hacer el cálculo por separado de cada una de ellas de tal forma que  $m_t$  será igual a  $m_{e1}$  y  $m_{x1}$ .

```

%% Ventana Deslizante
L1=2*zeta*wn;
L0=(L1^2)/8;
KD=((0.75)/(abs(kp*(wn^2))))*((L1/L0)^0.76);
Delta=0.68+0.12*(abs(kp*(wn^2))*KD*L1);
%% M(t)
Sx1=sign(kp*(wn^2))*-s;
mx1=(1/(kp*(wn^2)))*((2*zeta*wn-L1)*s+(wn^2))+KD*(Sx1/(-(Sx1)+Delta));
Sel=sign(kp*(wn^2))*(L1+(L0/s));
me1=(1/(kp*(wn^2)))*(L0)+KD*(Sel/((Sel)+Delta));

```

**Figura 11. Cálculo del controlador.**

Obteniendo los siguientes resultados para las partes de controlador en función del  $m$  o de  $x_1$ .

```

mx1 =
    -1488 s + 46.52
    -----
           s + 1.092
Continuous-time transfer function.

me1 =
    1247 s^2 + 119.1 s
    -----
    1.874 s^2 + 0.07645 s
Continuous-time transfer function.

```

**Figura 12. Controlador resultante.**

Y el filtro de primer orden quedaría de la siguiente manera:

```

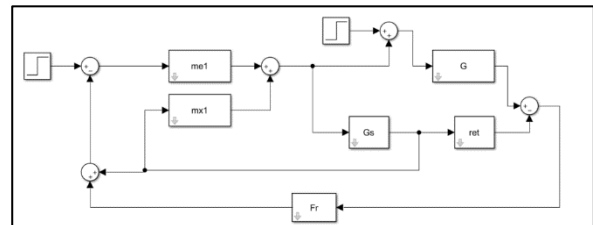
Fr =
    1
    -----
    2.74 s + 1
Continuous-time transfer function.

```

**Figura 13. Filtro implementado.**

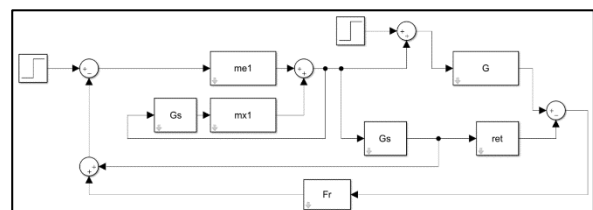
## 5. Simulación en Matlab y Simulink

Para poder llevar a implementar el controlador en un PLC S7-1500, se debe poder reducir el diagrama de bloques para que la implementación de esta estrategia de control sea más simple, de esta forma, se realiza una simplificación de bloques, pero primero partimos de un nuevo esquema del controlador, donde se tiene por separado el bloque del SMCr.



**Figura 14. P2-SMCr con el SMCr separado.**

De esta forma lo primero que se decide hacer es separar la salida de la planta sin retardo, que se usa como entrada para el bloque de control  $m_{x1}$ , por lo cual la estructura se ve de la siguiente forma:



**Figura 15. Modificación del SP2-SMCr.**

Como se ve en la figura anterior, se puede hacer una simplificación, dado que se tiene una realimentación de  $G_s \cdot m_{x1}$ , lo cual se hace con el siguiente comando:

```

%% Simplificación
C2=feedback(1,-mx1*Gs);

```

**Figura 16. Realimentación del SP2-SMCr.**

Y el diagrama de bloques se ve la anterior equivalencia multiplicando  $m_{e1}$ .





```

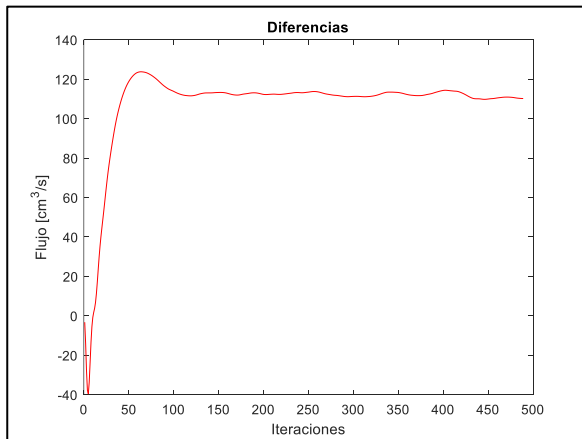
yR(10)=0;yR(11)=0;yR(12)=0;
yF(10)=0;yF(11)=0;yF(12)=0;

...
*(k-4)+B0ce(6)*k(k-5)+B0ce(7)*k(k-6)+B0ce(8)*k(k-7)+B0ce(9)*k(k-8)+B0ce(10)*k(k-9)+B0ce(11)*k(k-10)+B0ce(12)*k(k-11)+...
B0ce(6)*k(k-5)+B0ce(7)*k(k-6)+B0ce(8)*k(k-7)+B0ce(9)*k(k-8)+B0ce(10)*k(k-9)+B0ce(11)*k(k-10)+B0ce(12)*k(k-11)+
(k-3)+...

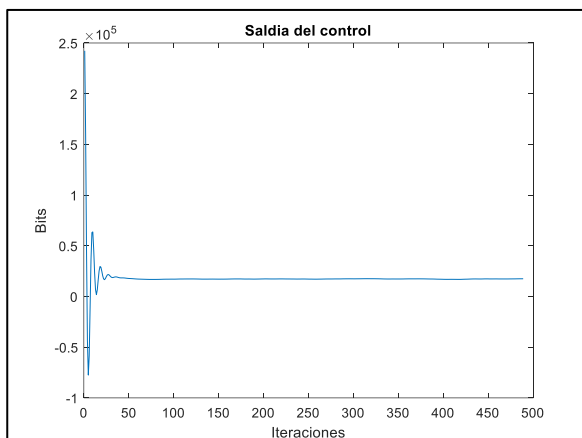
```

**Figura 23. Simulación en diferencias.**

Con la cual se obtiene la siguiente simulación:



**Figura 24. Salida de la simulación de las ecuaciones en diferencias.**



**Figura 25. Salida del controlador en ecuaciones en diferencias.**

## 6. Implementación en PLC.

Como ya se tiene el controlador y filtro en ecuaciones en diferencias, se puede implementar en el PLC, para lo cual se crean dos bloques FB con un lenguaje SCL, el cual permite introducir las ecuaciones y manejo de variables más sencillo, de tal forma que el bloque del filtro se ve por dentro de la siguiente manera:

```

1 //0.1145 z + 0.1145
2 //-----
3 // z - 0.771
4 //F2
5 // 0.05026 z + 0.05026
6 //-----
7 //z - 0.8995
8
9 #e := #Flujo;
10
11 //#u := 0.1145 * #e + 0.1145 * #e1 + 0.771 * #ul;
12 #u := 0.05026 * #e + 0.05026 * #e1 + 0.8995 * #ul;
13
14 #Flujo_Filtrado := #u;
15
16 #ul:=#u;
17 #e1 := #e;

```

**Figura 26. Filtro implementado en el PLC.**

Mientras que el bloque del controlador equivalente de 11vo orden se ve así (ver Anexo 2):

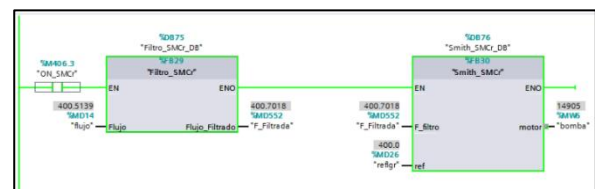
```

...
B0ce(6)*k(k-5)+B0ce(7)*k(k-6)+B0ce(8)*k(k-7)+B0ce(9)*k(k-8)+B0ce(10)*k(k-9)+B0ce(11)*k(k-10)+B0ce(12)*k(k-11)+
(k-3)+...
...

```

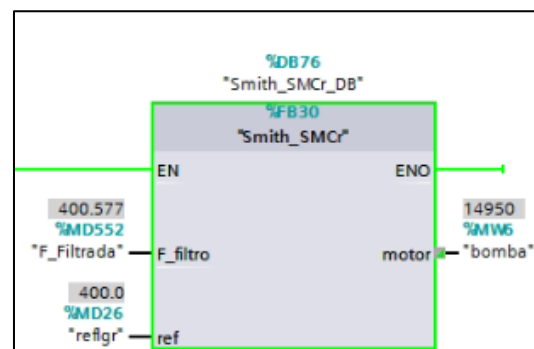
**Figura 27. Controlador implementado en el PLC.**

De esta forma se compila y carga el programa y se comprueba el correcto funcionamiento del controlador diseñado



**Figura 28. Comportamiento del controlador y filtro implementados.**

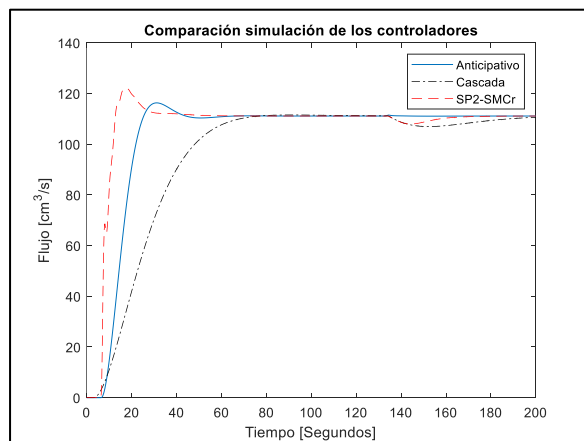
Y se ve como a una referencia de 111.1 cm³/s de flujo el controlador se estabiliza y genera la salida de a la bomba necesaria para garantizar el valor de flujo real censado igual al de la referencia.



**Figura 29. Bloque del controlador implementado.**

## 7. Análisis y Resultados.

Se compara en simulación el desempeño del control Anticipativo, control en Cascada y el control SP2-SMCr, para conocer el comportamiento de los controladores al ser implementados.



**Figura 30. Tipos de control.**

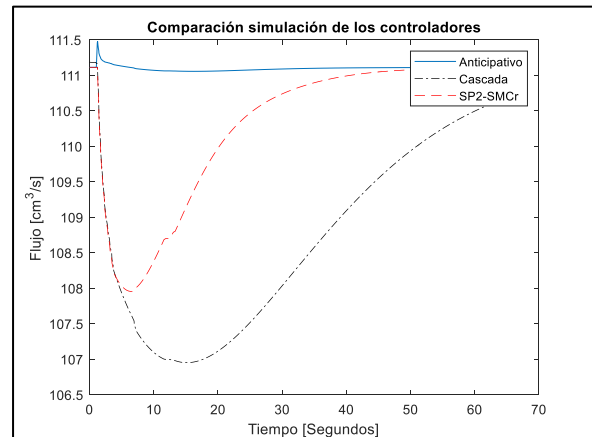
Como se observa en la figura anterior, cada una de estas metodologías de control sigue una referencia de buena forma, el control en Cascada evidencia un tiempo de establecimiento mas largo, en el caso del control SP2-SMCr presenta un menor tiempo de establecimiento acosta de un sobre impulso mayor (menor al 10%) que los otros esquemas de control, a continuación, se muestra una tabla con las características de la respuesta de cada uno de los desempeños.

Esquemas de Control	Sobre Impulso [%]	Tiempo de Establecimiento [Segundos]	Error estado estacionario [%]
SP2-SMCr	9.7	28.14	0
Anticipativo	4.7	38.57	0
Cascada	0	66.3	0

**Tabla 1. Características de desempeño.**

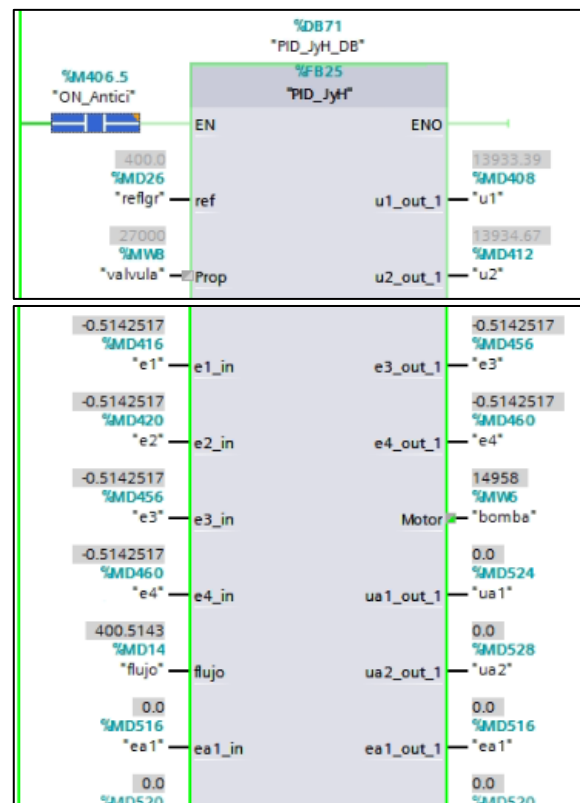
Como se ve en la simulación se realiza una perturbación a cada uno de los diferentes métodos, lo cual muestra el control SP2-SMCr realiza una recuperación más rápida ante una perturbación en comparación con el control en cascada, pero este no supera al control anticipativo, este control es mucho mejor ante

perturbaciones, cada uno se recupera y vuelve a su valor de referencia como se puede ver en la siguiente figura.



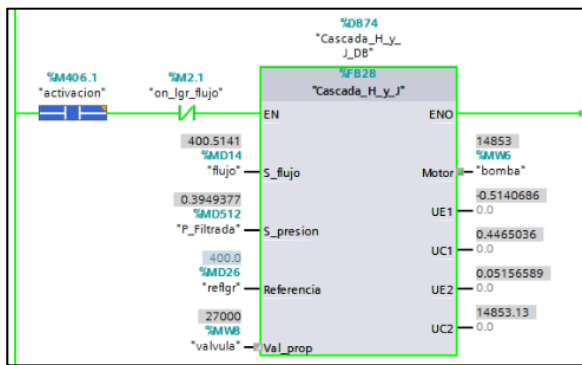
**Figura 31. Efecto de una perturbación en los distintos esquemas.**

Si bien se mostró un buen desempeño de cada una de las metodologías de control en la simulación, se realiza la implementación donde dio como resultado un correcto funcionamiento de estas, por lo cual se decide comparar ese buen comportamiento con respecto a otras metodologías de control.



**Figura 32. Bloque del controlador anticipativo.**

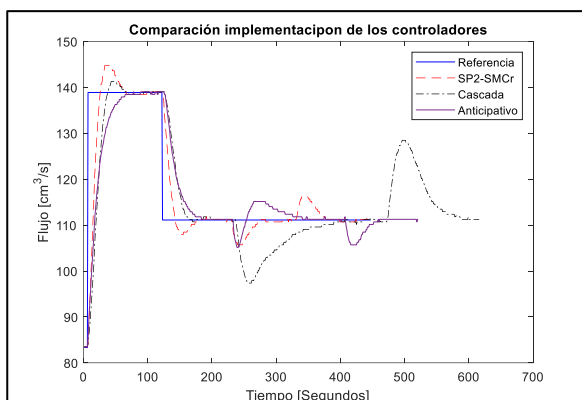




**Figura 33. Bloque del control en cascada.**

Para poder llevar a cabo esta comparación, se decidió empezar a tomar datos con el sistema estabilizado en una referencia de 83.3 cm<sup>3</sup>/s, después se incrementa la referencia a 138.8 cm<sup>3</sup>/s y en cuanto se estabiliza se baja está a 111.1 cm<sup>3</sup>/s, donde una vez estable se procede a hacer una perturbación en la válvula proporcional cerrándola un 20%.

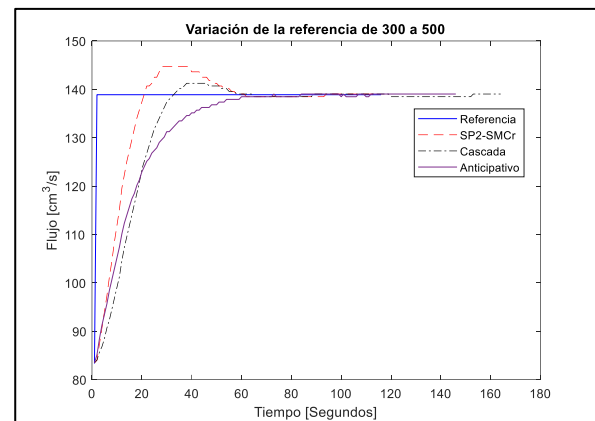
A continuación, se muestra el comportamiento del sistema en los diferentes métodos control:



**Figura 34. Comportamiento del sistema con el controlador cascada.**

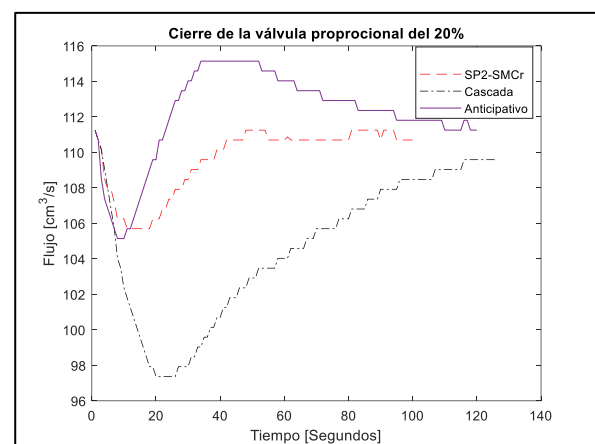
Viendo de forma general el comportamiento del sistema con los distintos controladores, se puede decir de forma generalizada, que el método del control anticipativo presenta menos sobreimpulso en la respuesta del sistema, además el control en cascada es muy sensible a las perturbaciones generando grandes picos en la salida del sistema. De esta forma se podría hacer un análisis de forma general, pero no sería muy veraz este, por lo tal a continuación se mostrarán tres gráficas en las que se verán específicamente los cambios de referencia de 83.3 cm<sup>3</sup>/s a 138.8 cm<sup>3</sup>/s y el cierre del 20% de la válvula proporcional en todos los casos se iniciaran las

gráficas en el inicio de la variación de la salida del sistema de flujo.



**Figura 35. Comparación del comportamiento de los tres métodos de control.**

De esta forma se puede comparar de forma más específica el desempeño de las tres estrategias de control, viendo como el SP2-SMCR es más rápido en su etapa de transición a diferencia de los otros métodos de control y de igual forma se aprecia como este método presenta mayor sobreimpulso, pero sin superar el 5% del valor de referencia con lo cual se toma como válido el funcionamiento.



**Figura 36. Comparación de los tres métodos de control ante una perturbación.**

Analizando el comportamiento del sistema con los distintos métodos de control, se puede apreciar que para el caso de una perturbación el método de control SP2-SMCR es mucho más eficaz para corregir el efecto de la perturbación, dado que es más rápido que los otros métodos, la variación es la menor de los tres casos y no presenta una oscilación en su comportamiento para la recuperación del sistema al valor de referencia introducido (de 111.1 cm<sup>3</sup>/s).

## 8. Conclusiones

La utilización de un modelo de segundo orden SOMTM en la identificación de la planta para el sistema de Flujo, este mostro un comportamiento igual con mínimo error, siendo posible un diseño de control por el método de ventana deslizante más robusto ante cualquier cambio en la planta.

El control SMCr muestra un excelente desempeño, el cual no posee error en estado estacionario, siendo un control rápido teniendo en cuenta que el sistema de Flujo posee un tiempo muerto elevado, además este tipo de control evidencia un comportamiento suave ante perturbaciones en la planta.

Al realizar la comparación de los 3 métodos de control ante la misma perturbación, se puede evidenciar que el método SCMr posee un buen comportamiento ante la perturbación siendo el mejor y el más rápido, donde el único que puede llegar a igualar este comportamiento es el control en cascada.

## Referencias

- [1] D. I. C. F. y C. O., «Controlador de Modos Deslizantes basado en Predictor de Smith y Modelo de Segundo Orden para Procesos con Elevado Retardo,» *Revista Politecnica*, vol. 35, n° 2, p. 7, 2015.

## Anexos

```
>> ceq

ceq =

3417 s^11 + 1.457e04 s^10 + 2.667e04 s^9 + 2.767e04 s^8 + 1.804e04 s^7 + 7725 s^6 + 2197 s^5 + 409.5 s^4 + 47.7 s^3 + 3.124 s^2 + 0.08661 s
-----
5.135 s^11 + 21.61 s^10 + 70.63 s^9 + 127.7 s^8 + 129.4 s^7 + 76.84 s^6 + 26.79 s^5 + 5.232 s^4 + 0.5138 s^3 + 0.0189 s^2 + 4.337e-19 s

Continuous-time transfer function.
```

### Anexo 1. Controlador equivalente SP2-SMCr.

```
#e := #ref - #F_filtro;
//#u := 493.1 * #e - 4365 * #e1 + 17510 * #e2 - 42060 * #e3 + 67140 * #e4 - 74810 * #e5 + 59360 * #e6 - 33540 * #e7 + 13220 * #e8 - 3464 * #e9 + 542.4 * #e10 - 38.46 * #e11 + 7.979 * #u1 - 29.11 * #u2 + 64.54 * #u3 - 97.
//#u := 451.6 * #e - 3834 * #e1 + 14740 * #e2 - 33870 * #e3 + 51670 * #e4 - 54960 * #e5 + 41570 * #e6 - 22350 * #e7 + 8374 * #e8 - 2080 * #e9 + 308.4 * #e10 - 20.66 * #e11 + 7.445 * #u1 - 25.25 * #u2 + 52.06 * #u3 - 73.
//#u := 514.8 * #e - 4652 * #e1 + 19070 * #e2 - 46790 * #e3 + 76370 * #e4 - 87050 * #e5 + 70710 * #e6 - 40920 * #e7 + 16540 * #e8 - 4443 * #e9 + 714.1 * #e10 - 52.02 * #e11 + 8.259 * #u1 - 31.22 * #u2 + 71.67 * #u3 - 13.
#u := 605.8 * #e - 5974 * #e1 + 26760 * #e2 - 71870 * #e3 + 128600 * #e4 - 161000 * #e5 + 143800 * #e6 - 91700 * #e7 + 40900 * #e8 - 12150 * #e9 + 2165 * #e10 - 175.2 * #e11 + 9.525 * #u1 - 41.45 * #u2 + 108.9 * #u3 - 19.
|

IF #u > 27000.0 THEN
    #motor := 27000.0;
ELSIF #u < 6500.0 THEN
    #motor := 6500.0;
ELSE
    #motor := #u;
END_IF;
// E
#e11 := #e10;
#e10 := #e9;
#e9 := #e8;
#e8 := #e7;
#e7 := #e6;
#e6 := #e5;
#e5 := #e4;
#e4 := #e3;
#e3 := #e2;
#e2 := #e1;
#e1 := #e;
// U
#u11 := #u10;
#u10 := #u9;
#u9 := #u8;
#u8 := #u7;
#u7 := #u6;
#u6 := #u5;
#u5 := #u4;
#u4 := #u3;
#u3 := #u2;
#u2 := #u1;
#u1 := #u;
```

```
#e11 + 7.979 * #u1 - 29.11 * #u2 + 64.54 * #u3 - 97.51 * #u4 + 106.4 * #u5 - 86.14 * #u6 + 51.96 * #u7 - 22.83 * #u8 + 6.917 * #u9 - 1.289 * #u10 + 0.1109 * #u11;  
#e11 + 7.445 * #u1 - 25.25 * #u2 + 52.06 * #u3 - 73.58 * #u4 + 76.13 * #u5 - 59.64 * #u6 + 35.52 * #u7 - 15.65 * #u8 + 4.785 * #u9 - 0.8995 * #u10 + 0.07774 * #u11;  
#e11 + 8.259 * #u1 - 31.22 * #u2 + 71.67 * #u3 - 111.8 * #u4 + 125.1 * #u5 - 103.2 * #u6 + 62.85 * #u7 - 27.68 * #u8 + 8.368 * #u9 - 1.553 * #u10 + 0.1332 * #u11;  
#e11 + 9.525 * #u1 - 41.45 * #u2 + 108.9 * #u3 - 191.9 * #u4 + 238.8 * #u5 - 214.1 * #u6 + 138.3 * #u7 - 63.18 * #u8 + 19.42 * #u9 - 3.615 * #u10 + 0.3086 * #u11;
```

## Anexo 2. Controlador implementado en el PLC.