

Proyecto de Bachelor

Control de la Presión en Redes de Distribución de Agua Potable

MEMORIA

Autor: Florian Eygonnet
Director: Vicenc Puig



Escuela Técnica Superior
de Ingeniería Industrial de Barcelona



Indice

1. INTRODUCCIÓN.....	3
2. ESTADO DEL ARTE.....	4
3. MODELO FISICO DE LA VÁLVULA.....	5
4. DISEÑO DE UN PRIMER CONTROLADOR.....	8
5. DISEÑO DEL CONTROLADOR HÍBRIDO.....	10
6. COMPARACIÓN CON OTROS CONTROLADORES.....	15
7. CONCLUSIÓN Y TRABAJOS FUTUROS.....	18
8. BIBLIOGRAFIA.....	19

1. Introducción

En este trabajo se estudiarán diferentes estrategias, y en particular una estrategia llamada híbrida, para el control de presión en redes de distribución.

Tenemos una canalización que contiene una válvula que se puede abrir y así disminuye la presión downstream, o cerrar y así aumenta la presión downstream. Queremos que esta presión sea la más cerca posible en todo momento a un valor de consigna que puede variar cada minuto. Esta regulación de la presión downstream es perturbada por la presión upstream y el caudal que también pueden variar cada minuto. Se precisará el modelo físico más adelante.

Se caracterizará una primera estrategia simple de control. Después se estudiará el controlador híbrido y se hará una comparación entre controladores.

2. Estado del arte

Se utiliza el control de presión para reducir las fugas y la incidencia de estallidos de tuberías. El control se puede implementar con Pressure Reducing Valves (PRV). Se adoptan single-feed PRV schemes para facilitar el control y monitoreo pero se arriesga a la interrupción del suministro en caso de falla. Multifeed sistemas mejoran la seguridad del suministro pero son más complejos y hay el riesgo de interacción de PRV que conduce a la inestabilidad.

En este proyecto se intenta pensar la válvula y su controlador como un sistema híbrido, o sea un sistema que tiene un comportamiento continuo en la relación entre la apertura de la válvula y la presión, y también un comportamiento discreto con las acciones de parar, cerrar, o abrir la válvula. En este sentido es interesante utilizar el Hybrid System Description Language (Hysdel) y luego simular el sistema en Simulink con la Hybrid Toolbox de Alberto Bemporad.

3. Modelo físico de la válvula

Modelo de la válvula según Brdys y Ulanicki, 1994:

$$q(t) = K f(\theta(t)) \sqrt{P_{ups}(t) - P_{dns}(t)}$$

donde $q(t)$ es el caudal en l/s, $f(\theta(t))$ una función que varía entre 0 (excluido) y 1 según la apertura $\theta(t)$ de la válvula, K es una constante en l/(s.m^{1/2}), y $P_{ups}(t)$ y $P_{dns}(t)$ las presiones upstream y downstream en metros de columna de agua. Se define el ángulo de la válvula $\theta(t)$, expresado en grados, tal que la posición cerrada es la inicial y corresponde a $\theta_0 = 90$ grados. La velocidad angular de la válvula es constante igual a $\frac{\pi/2}{250}$ rad/s y por lo tanto $\dot{\theta}(t) = \frac{90}{250}$ grados/s.

Descripción de la función de apertura $f(\theta)$:

Tenemos los datos siguientes:

Posición angular $\theta(t)$ en grados	Función $f(\theta(t))$
10 (abierta)	0
20	0.02
30	0.05
40	0.10
50	0.18
60	0.30
70	0.50
80	0.90
90 (cerrada)	1

Haciendo una interpolación polinomial de grado 4 con Matlab, tenemos:

$$f(\theta(t)) = -1,32 \cdot 10^{-7} \theta(t)^4 + 2,66 \cdot 10^{-5} \theta(t)^3 - 0,0016 \theta(t)^2 + 0,0370 \theta(t) - 0,2506$$

Determinación de la constante de la constante de la válvula K:

Utilizando datos para el caudal y para las presiones downstream y upstream, podemos dibujar en un mismo grafico el caudal $q(t)$ y el factor de conversión presión-caudal $C_{pc}(t)$ que se define por:

$$C_{pc}(t) = \frac{q(t)}{\sqrt{P_{ups}(t) - P_{dns}(t)}} = K f(\theta(t))$$

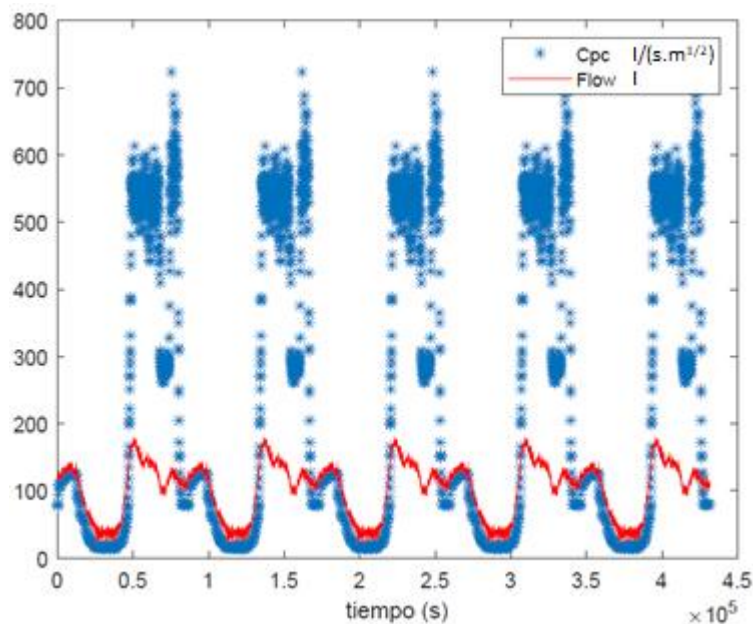


Figura1: Evolución del factor de conversión presión-caudal y del caudal. Los datos que tenemos son para un día. Se presenta aquí un ciclo de 5 días

Vemos que tenemos un caudal máximo en $t=505000s$ y un C_{pc} correspondiente de $600 \text{ l}/(\text{s} \cdot \text{m}^{1/2})$. Si hacemos la aproximación que el término $\sqrt{P_{ups}(t) - P_{dns}(t)}$ sea constante (en realidad varía entre 0.2 y 3), el caudal es máximo cuando $f=1$, o sea $C_{pc}=K$. Así $K=600 \text{ l}/(\text{s} \cdot \text{m}^{1/2})$.

Después para mejorar nuestra aproximación, probando experimentalmente con Simulink, añadimos al caudal un factor de corrección constante igual a 60 metros de columna de agua.

Nuestro nuevo modelo:

$$q(t) + C = K f(\theta(t)) \sqrt{P_{ups}(t) - P_{dns}(t)}$$

donde C es el factor de corrección.

Sacamos la variable $P_{dns}(t)$ que queremos controlar.

$$P_{dns}(t) = P_{ups}(t) - \frac{(q(t) + C)^2}{K^2 f(\theta(t))^2}$$

con:

$$K = 600$$

$$C = 60$$

$$f(\theta(t)) = -1,32 \cdot 10^{-7} \theta(t)^4 + 2,66 \cdot 10^{-5} \theta(t)^3 - 0,0016 \theta(t)^2 + 0,0370 \theta(t) - 0,2506$$

$$\theta(t) = u \cdot \frac{90}{250} t + 90, \quad \theta(t) \in [10, 90]$$

$$u = \begin{cases} -1 & \text{para abrir la válvula} \\ 0 & \text{para parar la válvula} \\ 1 & \text{para cerrar la válvula} \end{cases}$$

La planta tiene una salida $P_{dns}(t)$ y tres entradas: el comando u , y las dos perturbaciones $P_{ups}(t)$ y $q(t)$.

4. Diseño de un primer controlador

Se puede simular el sistema descrito en el capítulo 3 con el programa Simulink.

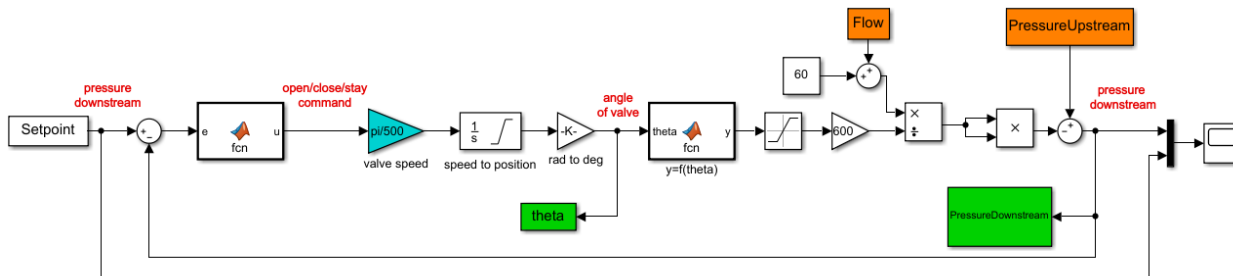


Figura2: Diagrama de bloques Simulink con un controlador Matlab Function

Setpoint es la consigna de presión downstream. Esta consigna es variable en el tiempo (por ejemplo no se necesita la misma presión durante el día y la noche cuando el consumo de agua es menor). Flow y PressureUpstream contienen las perturbaciones que son también variables en el tiempo, como se puede ver en la figura siguiente.

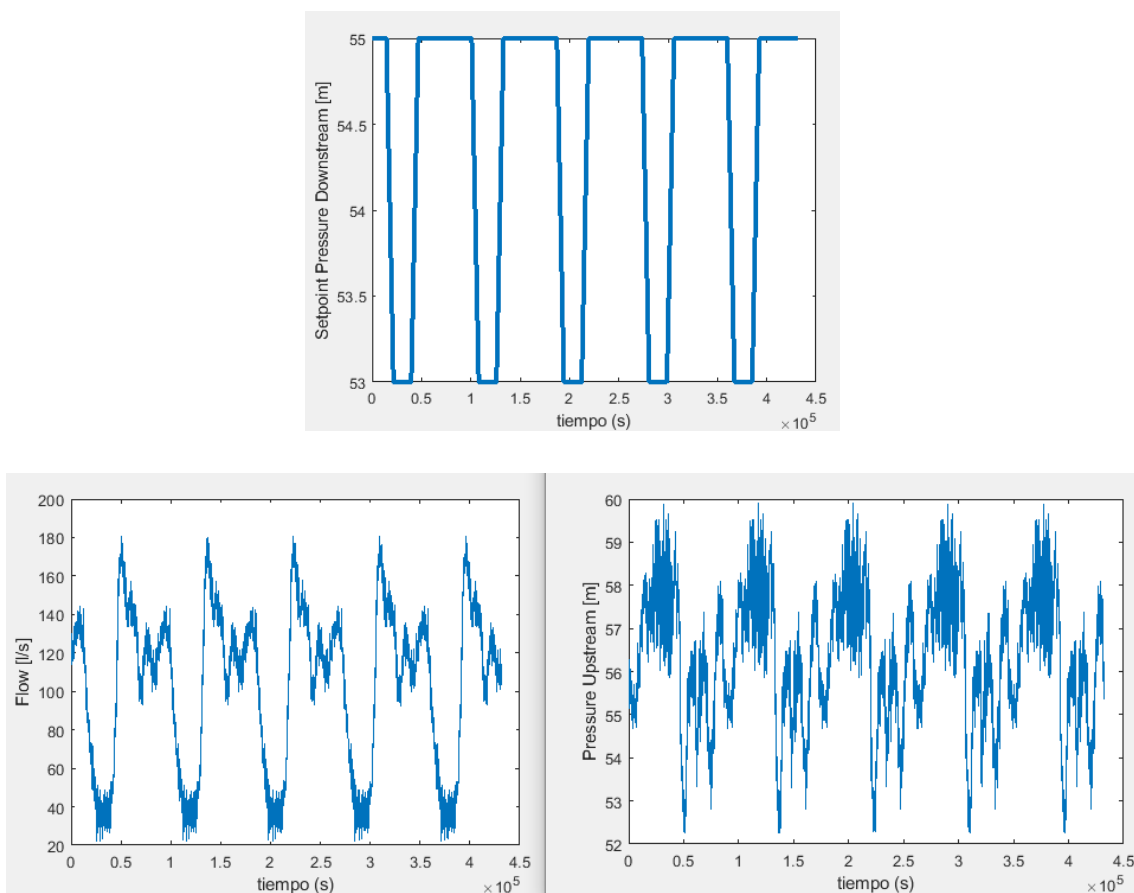


Figura3: Datos utilizados por el diagrama de bloques Simulink. Las presiones están expresadas en metros de columna de agua

El controlador es una Matlab Function cuya entrada es el error e , y la salida es el comando u .

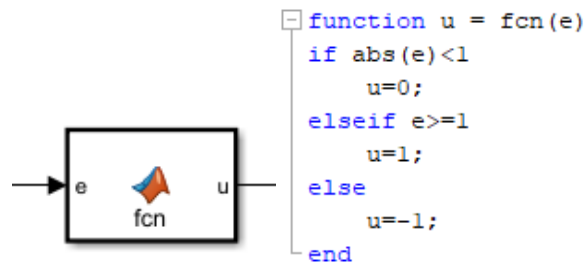


Figura4: Bloque del controlador y el código que contiene

Comentamos que en la Figura 2, hay otra Matlab Function $y=f(\theta)$ que es la función de apertura de la válvula que ya hemos visto al principio del capítulo 3. Hemos limitado la variable de salida del integrador entre 0.18 y $\pi/2$ rad (10 y 90 grados como lo queremos) y también la salida de la función de apertura entre 0.001 (para que nunca valga cero porque después hay una división) y 1. La condición inicial de integración es $\pi/2$.

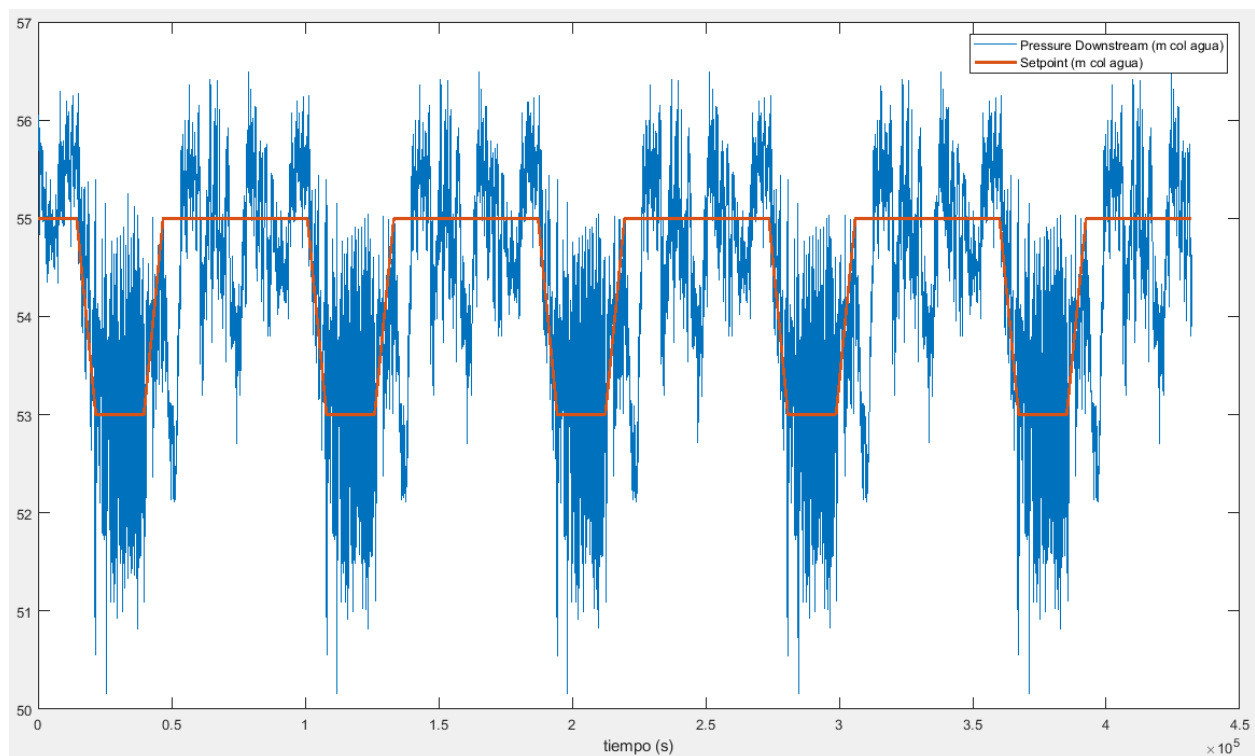


Figura5: Resultado del control de la presión downstream con un controlador Matlab Function

El resultado ya es bueno porque las oscilaciones de presión están casi todo el tiempo alrededor de la consigna. Vamos a intentar hacer lo mismo (o mejor) con un controlador híbrido en el capítulo que viene.

5. Diseño del controlador híbrido

En este capítulo se intenta diseñar un controlador híbrido usando el lenguaje de programación Hysdel y la Simulink Hybrid Toolbox de Alberto Bemporad.

Este lenguaje nos permite usar solo ecuaciones lineales y por lo tanto vamos a linealizar el modelo físico del capítulo 3. Por eso tenemos los datos de las perturbaciones y los datos de salida Theta y PressureDownstream de la simulación del capítulo 4. Así buscamos los mejores coeficientes α, β, γ para que $\text{PressureDownstream} \approx \alpha \text{ PressureUpstream} + \beta \text{ Flow} + \gamma \text{ Theta}$ (con Theta en grados). La función regress de Matlab da los coeficientes $\alpha = 0.878$, $\beta = 0.0284$, $\gamma = 0.0309$.

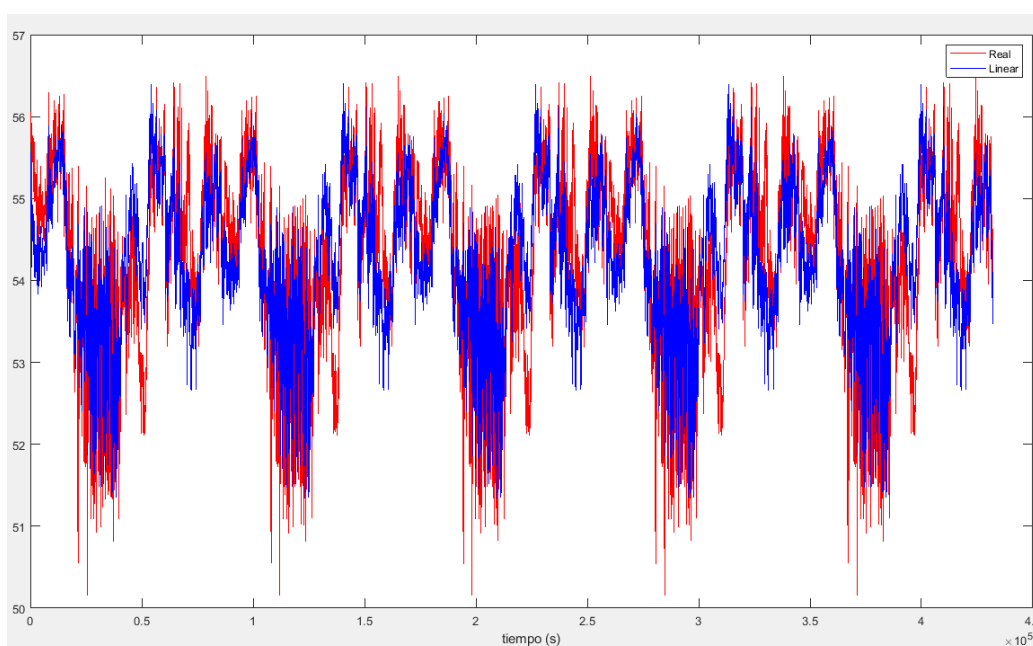


Figura6: Comparación de la presión downstream (m col agua) con y sin aproximación lineal

La aproximación lineal en azul es buena.

Ahora, usando el lenguaje Hysdel, escribimos un programa que describe nuestro sistema en un editor de texto. Se guarda el fichero con la extensión ".hys". Contiene el código siguiente:

```
SYSTEM ctrl {
INTERFACE {
STATE { REAL THETA [10,90];}

INPUT { REAL u [-1,1];}
```

```
    OUTPUT {REAL y;}

    PARAMETER {REAL Ts,vopen, vclose;}

}

IMPLEMENTATION {

    AUX { REAL u1,u2;

        BOOL open, close;

    }

    AD { open = u<=-1;

        close = u>=1;

    }

    DA { u1 = {IF open THEN vopen ELSE 0};

        u2 = {IF close THEN vclose ELSE 0};

    }

    CONTINUOUS {THETA=THETA+Ts*(u1+u2);}

    OUTPUT {y=THETA;}

}

}
```

Se construye en Matlab un modelo MLD llamando al fichero ctrl.hys que ya hemos creado. Luego Matlab puede generar el controlador híbrido a partir de este modelo MLD. La estrategia de control es el Model Predictive Control (MPC). El código es:

```
Ts=60;

vopen=-90/250;

vclose=90/250;

S=mld('ctrl',Ts);
```

```
clear Q refs limits

refs.y=1; % just weight state

Q.y=1;

Q.rho=Inf; % hard constraints

% Q.norm=2; % Quadratic costs

Q.norm=Inf; % Infinity norm costs

N=2;

limits.xmin=[10];

limits.xmax=[90];

limits.umin=-1;

limits.umax=1;

C=hybcon(S,Q,N,limits,refs);

C.mipsolver='glpk'; % used for MILP

%C.mipsolver='cplex'; % used for MIQP

Setpointc=Setpoint;

for i=1:7200

    if Setpoint(i,2)==55

        Setpointc(i,2)=Setpoint(i,2)+0.5;

    end

end

end
```

Comentamos que en la última parte de este código se hace una pequeña corrección experimental en el Setpoint para que luego la simulación con Simulink sea mejor.

Ahora se puede simular nuestro sistema con Simulink.

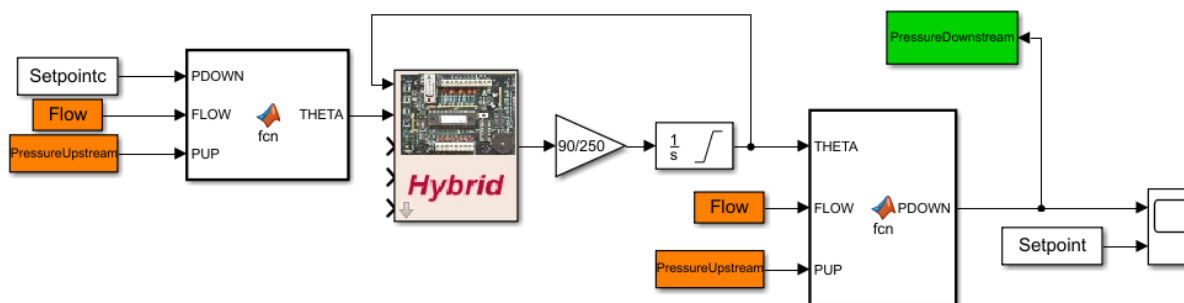


Figura7: Diagrama de bloques Simulink con un controlador híbrido

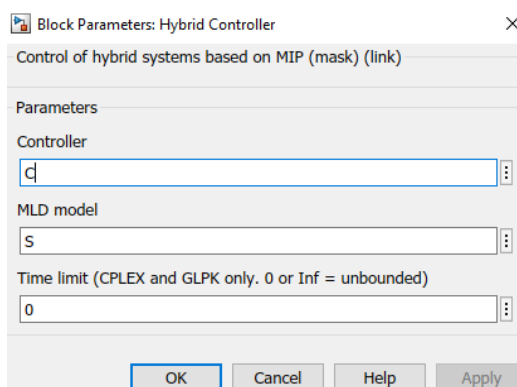


Figura8: Contenido del bloque del controlador híbrido

En la figura 7 doble clicando en el bloque del controlador Hybrid se abre la ventana que se ve en la figura 8. Se puede así cargar el modelo MLD y el controlador que ya hemos creado con el código Matlab antes.

En el diagrama de bloques de la figura 7, la constante de 90/250 es la velocidad de la válvula en grados por segundos. Justo después se integra la velocidad para obtener el ángulo con una condición inicial de 90 grados. Se limite el ángulo entre 10 y 90 grados.

Hay dos bloques Matlab Function : el que está a la derecha calcula la presión downstream con la formula lineal que ya hemos visto al principio de este capítulo. La Matlab Function a la izquierda convierte la consigna de presión downstream en una consigna del ángulo theta para que el controlador híbrido pueda hacer el control solo en theta sin preocuparse de las perturbaciones y del valor de la presión downstream (para que sea más simple en la parte de programación con el lenguaje Hysdel).

La linealización de la ecuación de la presión downstream, ya hecha al principio de este capítulo, en realidad no es necesaria porque la ecuación de la presión downstream no interviene en la programación Hysdel sino fuera en los dos bloques Matlab Function de la figura 7. Sin embargo, nos quedamos con esta versión lineal porque ya hemos visto que es una buena aproximación y es más fácil manipularla (inversión de la formula para sacar el ángulo theta en el bloque Matab Function a la izquierda en la figura 7). Además, si ahora queremos que la planta sea también de tipo híbrido en un trabajo futuro, esta linealización es necesaria ya que el lenguaje Hysdel necesita ecuaciones lineares.

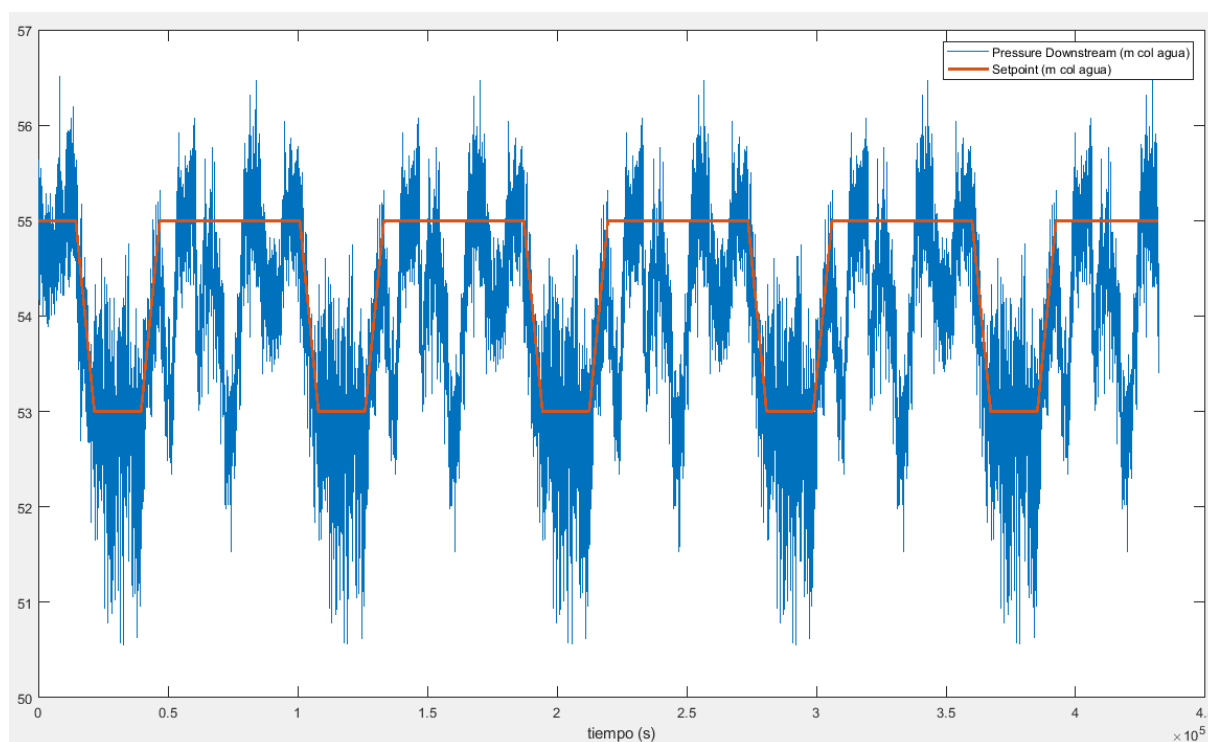


Figura9: Resultado del control de la presión downstream con un controlador híbrido

6. Comparación con otros controladores

En este capítulo se compara diferentes controladores.

En primer lugar, comentamos que un controlador muy parecido al del capítulo 4 es un controlador con switch y relay.

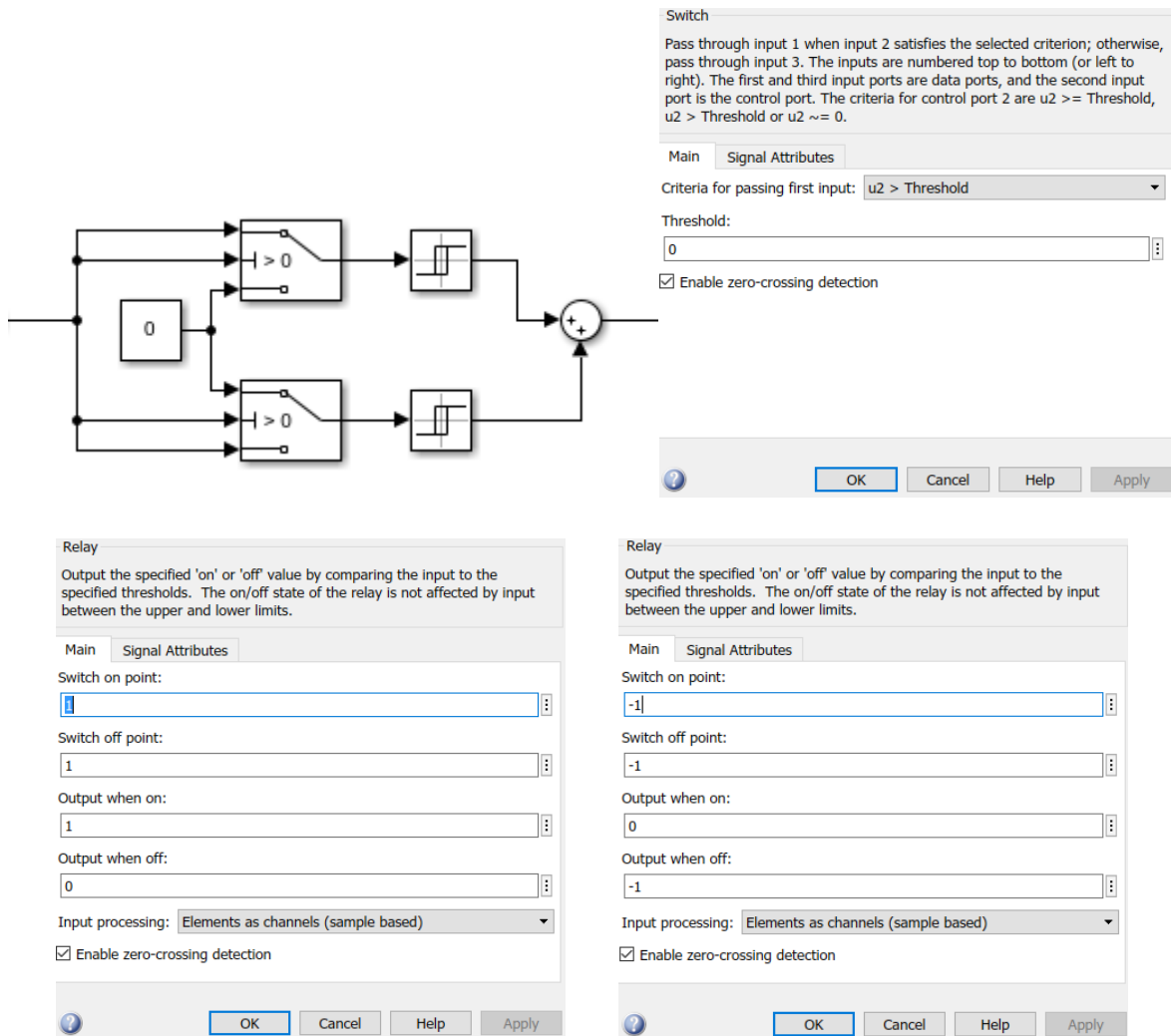


Figura10: Controlador con 2 switch idénticos y 2 relay

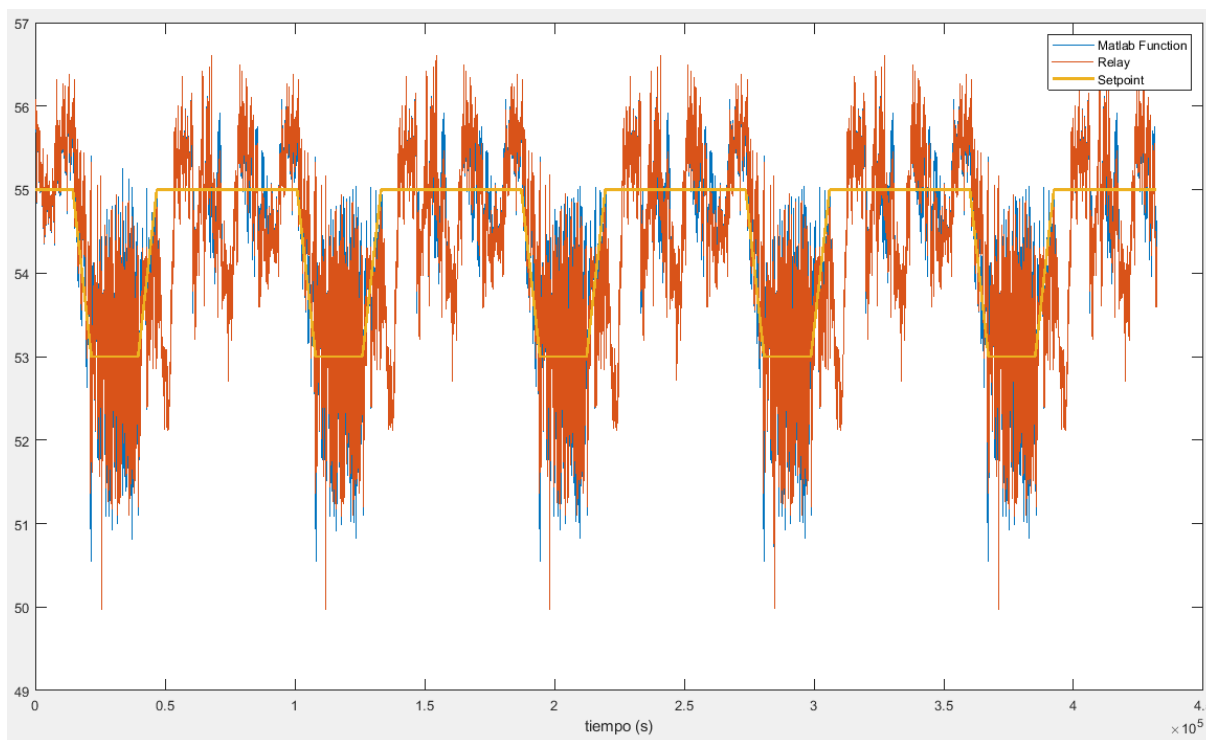


Figura 11: Comparación de la evolución de la presión downstream (m col agua) controlada por la Matlab Function del capítulo 4 o por los switch y relay

Entonces la Matlab Function del capítulo 4 y este nuevo controlador switch y relay dan resultados muy parecidos.

Ahora se compara nuestro controlador híbrido con el controlador del capítulo 4 (o con los switch y relay, da igual porque son casi equivalentes).

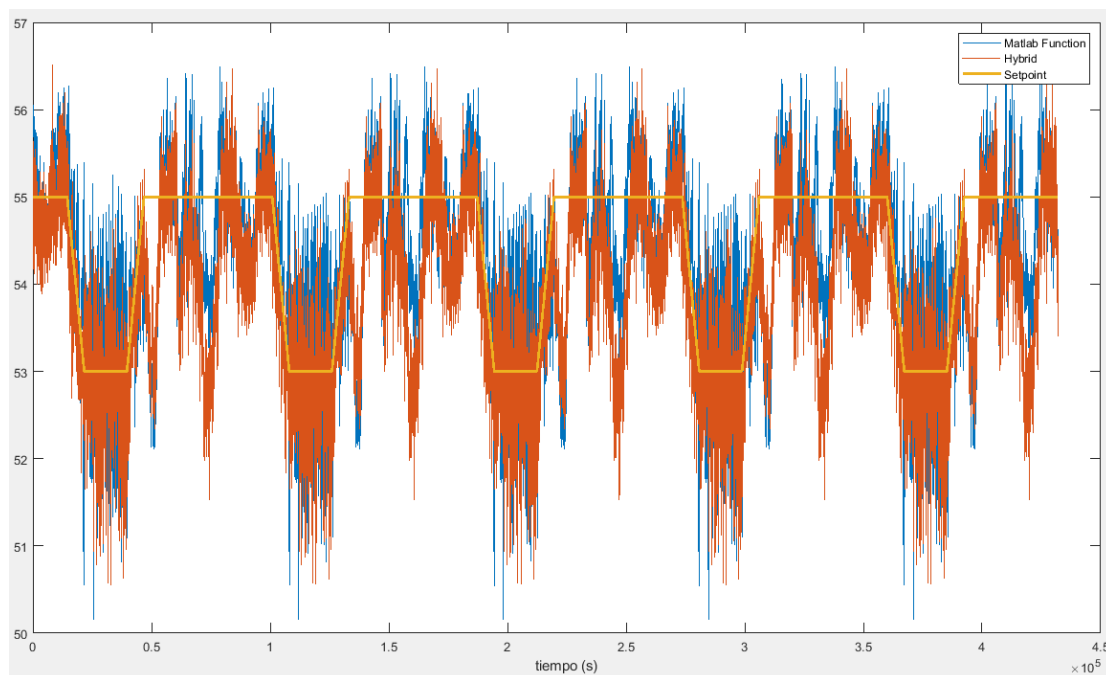


Figura 12: Comparación de la evolución de la presión downstream (m col agua) controlada por la Matlab Function o por el controlador híbrido

Entonces el controlador híbrido da oscilaciones de menor amplitud, pero en algunos puntos en el caso híbrido la presión se aleja más de la consigna, alrededor de $t= 74340s$, $160700s$, etc.

Al final también se puede comparar el controlador híbrido con un controlador PID hecho por Jonas Perolini (ver la bibliografía para más detalle). Los resultados son parecidos, un poco mejor para el PID.

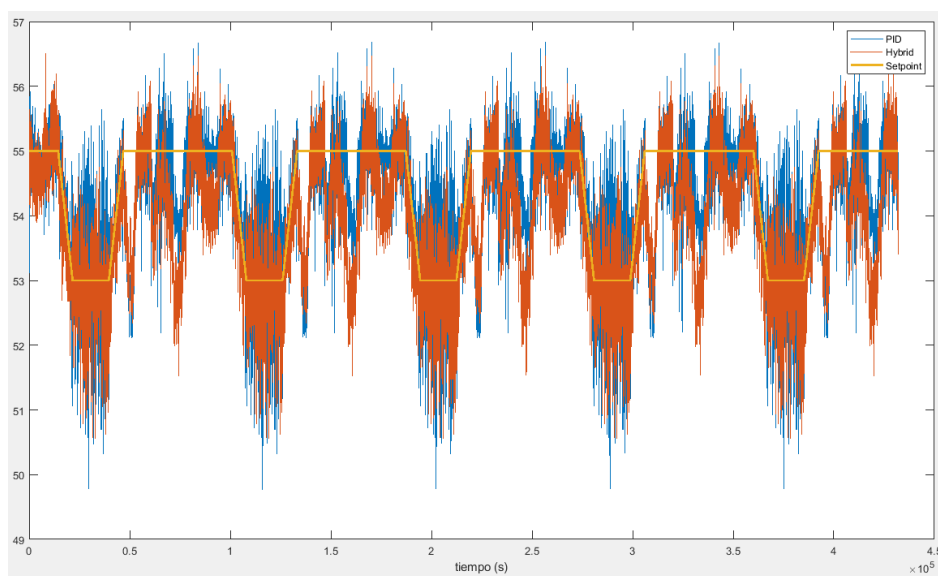


Figura 13: Comparación de la evolución de la presión downstream (m col agua) controlada por el PID o por el controlador híbrido

7. Conclusión y trabajos futuros

En este trabajo se ha estudiado y comparado estrategias de control como relay, híbrido, PID. Los resultados de control de la presión downstream por estas diferentes estrategias son parecidos y aceptables.

Para mejorar el resultado se debería intentar reducir las oscilaciones y estar menos lejos de la consigna en algunos puntos.

Se podría trabajar de nuevo con el controlador híbrido y quitar la planta que teníamos para poner una planta híbrida utilizando el bloque de la Hybrid Toolbox correspondiente. Se debería cambiar el código Hysdel para intentar que el control no se haga en el ángulo theta como en el capítulo 4 sino en la presión downstream.

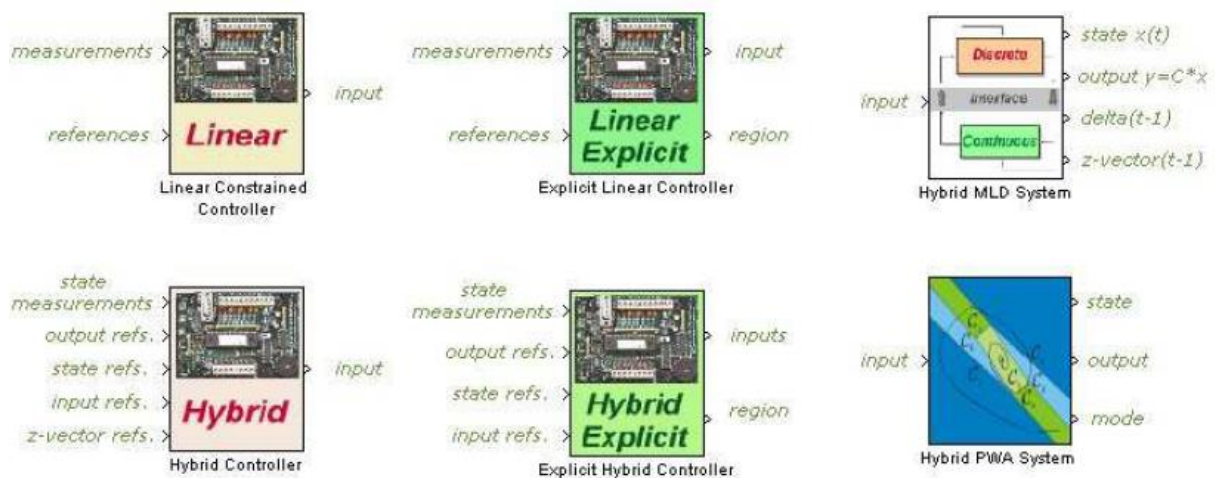


Figura14: Bloques de la Simulink Hybrid Toolbox.

El bloque de planta híbrida está a la derecha (MLD o PWA)

8. Bibliografía

Wide, *Modelo de valvula*

ETH, *Hysdel 3.0 Manual*

Fabio Danilo Torrisi and Alberto Bemporad, Member, IEEE, *Hysdel - A Tool for Generating Computational Hybrid Models for Analysis and Synthesis Problems*

Alberto Bemporad, *Hybrid Toolbox Manual*

Jonas Perolini, *PID para el control de la presión en redes de distribución de agua potable*

Simon L. Prescott and Bogumil Ulanicki, *Dynamic Modeling of Pressure Reducing Valves*