

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ**  
**ESCUELA DE POSGRADO**



**SINTONIZACIÓN DE UN CONTROLADOR PID UTILIZANDO  
ALGORITMOS GENÉTICOS APLICADA A UNA PLANTA  
CONCENTRADORA DE COBRE**

**TESIS PARA OPTAR EL GRADO ACADÉMICO DE MAGÍSTER EN  
INGENIERÍA DE CONTROL Y AUTOMATIZACIÓN**

**AUTOR**

Renato Javier Martínez Ordoñez

**ASESOR:**

Dr. Antonio Manuel Morán Cárdenas

Febrero, 2021

## RESUMEN

El objetivo principal de esta tesis es establecer un método para el modelamiento del proceso en un lazo cerrado de control PID (Controlador Proporcional integrativo y derivativo) y con este poder encontrar los parámetros óptimos usando sintonización basada en algoritmos genéticos.

En primer lugar, se explica cuál es la problemática que actualmente se tiene en la industria para realizar la sintonización de los lazos de control PID y se detalla el estado del arte de la sintonización de los controladores PID.

En segundo lugar, se evalúa cual es el método de identificación en lazo cerrado que mejor representa la respuesta real del proceso de inyección de agua cruda al cajón de alimentación de las bombas de ciclones.

En tercer lugar, se realiza la sintonía basada en algoritmos genéticos con el modelo obtenido con la identificación y se evalúa cual es la función de aptitud más adecuada para poder encontrar los parámetros del controlador PID.

Finalmente, se presenta los resultados de la sintonización del controlador PID obtenidos para el proceso de inyección de agua cruda al cajón de alimentación de las bombas de ciclones y el proceso de control de nivel de espuma de una celda de flotación Rougher.

## DEDICATORIA

A mi padre Cesar Martinez que en paz descansa quien siempre me apoyo en mis estudios y me enseñó a siempre superarme en mi vida familiar y laboral.

A mi madre Martha Ordoñez por su amor, esfuerzo y dedicación de toda la vida.

A mi esposa, el amor de mi vida, Yulisa Garcia por alentarme y apoyarme de forma incondicional para lograr mis metas.

A mis hijos Leonela, Javier y Berenice que son el motor de mi vida y la razón principal para siempre buscar superarme.

AGRADECIMIENTOS

Al Dr. Antonio Morán por sus consejos y apoyo para la elaboración de esta tesis.



## CONTENIDO

INTRODUCCIÓN.....	1
CAPÍTULO 1: ESTADO DEL ARTE DE LA SINTONIZACION DE LOS CONTROLADORES PID EN LA INDUSTRIA .....	3
1.1    Introducción.....	3
1.2    Estado del arte de los métodos de sintonización .....	3
1.2.1    Métodos empíricos.....	3
1.2.1.1    Método de Ziegler-Nichols .....	3
1.2.1.2    Método del relé.....	6
1.2.2    Métodos analíticos.....	7
1.2.2.1    Método de algoritmos genéticos.....	8
1.3    Modelamiento de los procesos.....	11
1.3.1    Modelo teórico.....	12
1.3.2    Identificación del sistema.....	12
1.3.2.1    Modelo ARX (exógeno autorregresivo).....	14
1.3.2.2    Modelo ARMAX (exógeno autorregresivo de media móvil).....	14
1.3.2.3    Modelo OE (error de salida) .....	14
1.3.2.4    Modelo BJ (Box Jenkins).....	14
1.3.3    Identificación de sistemas en lazo cerrado .....	15
1.3.3.1    Identificación de un sistema desconocido en lazo cerrado con un controlador conocido.....	16
1.4    Estado del arte de los índices de desempeño.....	17
1.5    Características genéricas del proceso de concentrado de cobre.....	19
1.5.1    Proceso de Chancado .....	19
1.5.2    Proceso de Molienda.....	20
1.5.3    Proceso de Flotación.....	23
1.5.4    Proceso de Espesamiento y Filtrado.....	25
1.6    Objetivos de la tesis.....	26
1.6.1    Objetivo general.....	27
1.6.2    Objetivos específicos .....	27
CAPÍTULO 2: MODELAMIENTO DE UN PROCESO EN UN SISTEMA DE LAZO CERRADO.....	28
2.1    Introducción.....	28

2.2	Método de identificación en lazo cerrado sobre un modelo teórico.....	28
2.2.1	Modelo teórico de un proceso control de flujo con válvula de diafragma 29	
2.2.2	Identificación en lazo cerrado sobre un proceso teórico .....	32
2.2.2.1	Identificación con modelo ARX .....	33
2.2.2.2	Identificación con modelo ARMAX .....	35
2.2.2.3	Identificación de proceso con controlador conocido .....	38
2.3	Método de identificación en lazo cerrado sobre un proceso real.....	43
2.3.1	Identificación con modelo ARX.....	45
2.3.2	Identificación con modelo ARMAX .....	47
2.3.3	Identificación de proceso con controlador conocido .....	50
2.4	Conclusiones del capítulo .....	54
<b>CAPÍTULO 3: SINTONIZACION DE UN CONTROLADOR PID UTILIZANDO</b>		
<b>ALGORITMOS GENETICOS.....</b>		
3.1	Introducción.....	55
3.2	Método de algoritmos genéticos .....	55
3.2.1	Modelamiento en tiempo discreto .....	55
3.2.2	Programación del método de algoritmos genéticos .....	56
3.2.2.1	Etapas 1-Parámetros .....	57
3.2.2.2	Etapas 2-Primera generación .....	58
3.2.2.3	Etapas 3-Evaluación de aptitud.....	59
3.2.2.4	Etapas 4-Padres más aptos.....	60
3.2.2.5	Etapas 5-Descendencia .....	60
3.2.2.6	Etapas 6-Mejor individuo .....	62
3.3	Sintonización del controlador PID.....	62
3.3.1	Índice de desempeño como función de aptitud.....	62
3.3.2	Elección de la función de aptitud .....	65
3.3.2.1	Elección del índice de desempeño.....	65
3.3.2.2	Elección de las especificaciones de la respuesta transitoria.....	68
3.4	Conclusiones del capítulo .....	73
<b>CAPÍTULO 4: RESULTADOS DE LA SINTONIZACION UTILIZANDO</b>		
<b>ALGORITMOS GENETICOS .....</b>		
4.1	Introducción.....	74

4.2	Resultado de la sintonización en un proceso real de control de flujo .....	74
4.2.1	Respuesta transitoria de la PV con el PID inicial .....	74
4.2.2	Respuesta transitoria de la PV con el PID Sintonizado.....	77
4.2.2.1	Modelamiento en tiempo discreto .....	77
4.2.2.2	Sintonización del controlador PID .....	78
4.2.2.3	Análisis de la repuesta transitoria.....	82
4.3	Resultado de la sintonización en un proceso real de control de nivel.....	85
4.3.1	Modelamiento del proceso real de control de nivel .....	85
4.3.2	Respuesta transitoria de la PV con el PID inicial .....	88
4.3.3	Sintonización del proceso real de control de nivel.....	89
4.3.4	Análisis de la repuesta transitoria.....	90
4.4	Conclusiones del capítulo.....	92
CONCLUSIONES GENERALES .....		94
BIBLIOGRAFIA.....		96
ANEXOS .....		1
Anexo A: Programas .....		1
A.1. Modelamiento y Sintonización.....		1
A.2. Función de aptitud .....		4

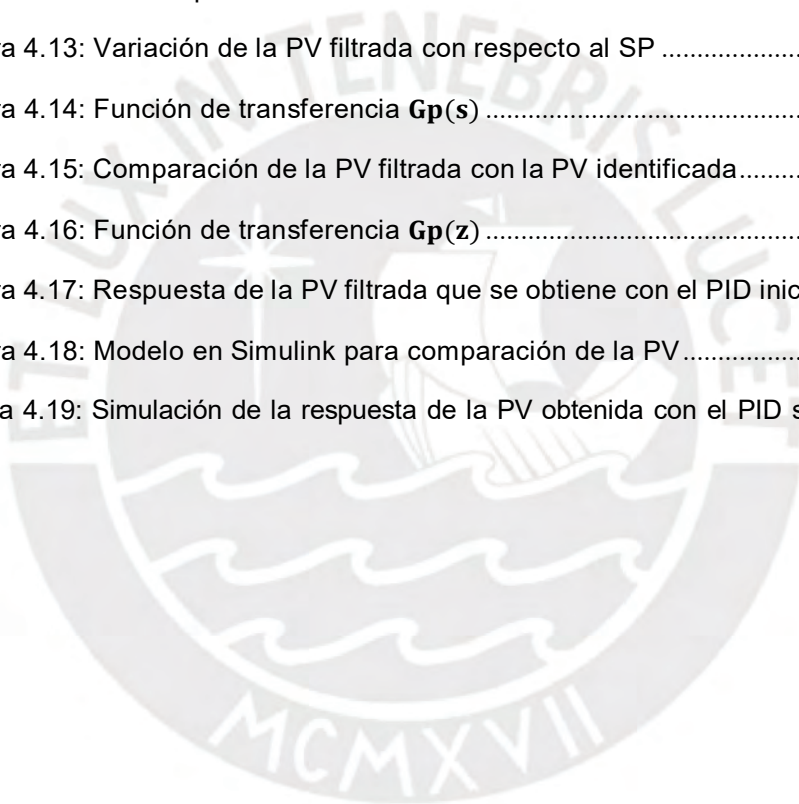
## ÍNDICE DE FIGURAS

Figura 1.1: Método de respuesta a un escalón (Kok Kion, 1999) .....	4
Figura 1.2: Sintonía en lazo cerrado de Ziegler-Nichols (Boiko, 2013) .....	5
Figura 1.3: Método relé en lazo cerrado (Gomery Pinto, 2014) .....	6
Figura 1.4: Salida del proceso (Gomery Pinto, 2014) .....	6
Figura 1.5: Crossover (Unal y otros, 2013) .....	9
Figura 1.6: Diagrama de flujo de la técnica de algoritmos genéticos .....	10
Figura 1.7: Diagrama de bloques de un control PID .....	11
Figura 1.8: Sistema que debe identificarse (Tangirala, 2015) .....	11
Figura 1.9: Diagrama de bloques para identificación de sistemas .....	13
Figura 1.10: Diagrama de bloques en lazo cerrado (Johnson y Moradi, 2005) .....	15
Figura 1.11: Identificación en lazo cerrado (Johnson y Moradi, 2005) .....	17
Figura 1.12: Respuesta a escalón unitario (Ogata, 2010) .....	18
Figura 1.13: Diagrama de flujo de un proceso genérico de Chancado .....	20
Figura 1.14: Diagrama de flujo de la primera etapa de un proceso genérico de Molienda .....	21
Figura 1.15: Diagrama de flujo de la segunda etapa de un proceso genérico de Molienda Secundaria .....	22
Figura 1.16: Funcionamiento del Hidrociclón .....	23
Figura 1.17: Diagrama de flujo de un proceso genérico de Flotación .....	24
Figura 1.18: Funcionamiento de la celda de Flotación .....	25
Figura 1.19: Diagrama de flujo de un proceso genérico de Espesamiento y Filtrado. ....	26
Figura 2.1: Diagrama de bloques de un lazo cerrado de control .....	29
Figura 2.2: Mecánica del actuador de una válvula de diafragma (Boiko, 2013) .....	29
Figura 2.3: Diagrama de bloques en lazo cerrado del modelo teórico .....	32
Figura 2.4: Muestreo de SP, MV y PV con tiempo de 0.1 segundos .....	32

Figura 2.5: Proceso $G(S)$ que se debe identificar .....	33
Figura 2.6: Representación esquemática de la estructura ARX (Tangirala, 2015)..	34
Figura 2.7: Comparación de la respuesta teórica con el modelo ARX .....	35
Figura 2.8: Representación esquemática de la estructura ARMAX (Tangirala, 2015) .....	36
Figura 2.9: Comparación de la respuesta teórica con el modelo ARMAX .....	37
Figura 2.10: Proceso $GP(s)$ que se debe identificar .....	38
Figura 2.11: Ejecución del comando “tfest” de Matlab.....	40
Figura 2.12: Comparación de la respuesta $GCL(s)$ muestreado con su identificación .....	40
Figura 2.13: Función de transferencia $Gp(s)$ calculado en Matlab .....	41
Figura 2.14: Modelo en Simulink para validar la identificación de $Gp(s)$ .....	42
Figura 2.15: Comparación entre la salida $GCL(s)$ muestreada y la salida de la identificación de $Gp(s)$ .....	42
Figura 2.16: Diagrama de bloques del proceso de inyección de agua cruda al cajón de alimentación de las bombas de ciclones.....	43
Figura 2.17: Variación de la PV con respecto a la MV .....	44
Figura 2.18: Variación de la PV con respecto al SP .....	45
Figura 2.19: Comparación del proceso muestreado con el modelo ARX del ensayo 1 .....	46
Figura 2.20: Máxima estimación de datos encontrada con el modelo ARX.....	47
Figura 2.21: Comparación del proceso muestreado con el modelo ARMAX ensayo 1 .....	48
Figura 2.22: Máxima estimación de datos encontrada con el modelo ARMAX.....	49
Figura 2.23: Proceso $GP(s)$ a identificar .....	50
Figura 2.24: Código en Matlab para obtener $GCL(s)$ .....	50
Figura 2.25: Comparación de la respuesta $GCL(s)$ muestreado e identificado .....	51
Figura 2.26: Función de transferencia $Gp(s)$ calculado en Matlab .....	52
Figura 2.27: Modelo en Simulink para validar la identificación de $Gp(s)$ .....	52

Figura 2.28: Comparación en Simulink de los datos reales muestreados con la respuesta de la identificación de $G_p(s)$ .....	52
Figura 2.29: Comparación de la PV muestreada con la PV identificada .....	53
Figura 3.1: Diagrama de bloques del lazo cerrado de control de flujo en tiempo discreto .....	55
Figura 3.2: Función de transferencia $G_p(z)$ calculado en Matlab .....	56
Figura 3.3: Diagrama de flujo de la técnica de algoritmos genéticos .....	57
Figura 3.4: Definición de los parámetros del algoritmo genético.....	58
Figura 3.5: Definición de los parámetros del algoritmo genético.....	58
Figura 3.6: Cálculo de la función de aptitud por cada individuo .....	59
Figura 3.7: Obtención de los mejores individuos.....	60
Figura 3.8: Descendencia de la generación anterior .....	61
Figura 3.9: Parámetros sintonizados del controlador PID.....	62
Figura 3.10: Modelo en Simulink para comparación de la PV y la MV .....	63
Figura 3.11: Respuesta de la PV sintonizada y la PV inicial.....	63
Figura 3.12: Respuesta de la MV sintonizada y la MV inicial .....	64
Figura 3.13: Calculo del error integral.....	65
Figura 3.14: Respuesta de la variable de proceso por cada índice de desempeño.66	
Figura 3.15: Respuesta de la variable manipulada por cada índice de desempeño 67	
Figura 3.16: Condición de máximo sobreimpulso .....	69
Figura 3.17: Función de aptitud .....	69
Figura 3.18: Respuesta de la variable de proceso por cada función de aptitud	70
Figura 3.19: Respuesta de la variable manipulada por cada función de aptitud	71
Figura 4.1: Respuesta de la PV con el PID inicial .....	75
Figura 4.2: Comportamiento de la PV con respecto a la MV .....	76
Figura 4.3: Diagrama de bloques del proceso real de control de flujo en tiempo discreto .....	77
Figura 4.4: Función de transferencia $G_p(z)$ calculado en Matlab .....	78

Figura 4.5: Especificaciones de la respuesta transitoria .....	79
Figura 4.6: Primera generación.....	79
Figura 4.7: Descendencia (crossover y mutación) .....	80
Figura 4.8: Comparación de la PV Sintonizada y la PV inicial .....	81
Figura 4.9: Respuesta de la PV con el PID sintonizado .....	82
Figura 4.10: Comportamiento de la PV con respecto a la MV .....	83
Figura 4.11: Variación de la PV con respecto al SP .....	85
Figura 4.12: Método para filtrar la PV.....	86
Figura 4.13: Variación de la PV filtrada con respecto al SP .....	86
Figura 4.14: Función de transferencia $G_p(s)$ .....	87
Figura 4.15: Comparación de la PV filtrada con la PV identificada.....	87
Figura 4.16: Función de transferencia $G_p(z)$ .....	88
Figura 4.17: Respuesta de la PV filtrada que se obtiene con el PID inicial.....	89
Figura 4.18: Modelo en Simulink para comparación de la PV .....	90
Figura 4.19: Simulación de la respuesta de la PV obtenida con el PID sintonizado	91



## INTRODUCCION

El controlador PID es el que más se utiliza en las plantas concentradoras de cobre ya que estas son diseñadas en su gran mayoría para trabajar en rangos de operación donde estos controladores son suficientes para poder controlar el proceso, claro está que hay algunas industrias que tienen procesos donde el control PID no es suficiente y se necesita usar algoritmos de control avanzado, pero esta tesis se centra en industrias de gran minería donde en general no se cuenta con algoritmos de control avanzado y en Perú no se tiene estas implementaciones, en general en gran minería se tiene entre 100 y 200 lazos de control PID, por tal motivo es muy importante la sintonización de los lazos de control ya que esto permitirá una óptima operación y menor desgaste en los elementos de control.

Para poder realizar la sintonización en la mayoría de los casos los ingenieros de control usan su experiencia, pero no logran una óptima sintonización y además que debido al desgaste de equipos y a los cambios en el proceso es necesario que cambien sus parámetros, pero no lo hacen, por este motivo los ingenieros de la sala de control en algunos casos trabajan el lazo de control de forma manual es decir en lazo abierto. Unos pocos ingenieros de control quieren usar métodos de sintonización como Ziegler-Nichols o método de relé, pero estos son métodos empíricos que generan inestabilidad al proceso por lo que no es posible realizarlo ya que las plantas concentradoras trabajan las 24 horas del día y los 7 días de la semana.

Los métodos de sintonización empíricos generan inestabilidad en el proceso y en la práctica no se pueden realizar, por este motivo se debe usar un método de sintonización que no genere inestabilidad en el proceso por lo que es necesario realizar un modelamiento del proceso fuera de línea y así poder usar un algoritmo que permita encontrar los parámetros óptimos del controlador PID.

Debido a los cambios en el proceso (por ejemplo una alta reducción de inyección de agua a la descarga de los molinos de bolas para la alimentación a los hidrociclones debido a una reducción de alimentación en el Molino de SAG y mayor humedad en el

mineral por lluvias), desgastes de los equipos (por ejemplo reducción del rendimiento de las bombas por desgaste mecánico) y elementos de control (por ejemplo desgaste en las válvulas de control) un modelo teórico del proceso no puede realizarse ya que en la práctica no se puede calcular estos cambios físicos para poder actualizar el modelo, por este motivo es necesario encontrar un método de modelamiento del proceso que pueda cambiar de acuerdo al histórico de los datos del proceso, es decir se requiere realizar una identificación.

Para la sintonía del controlador PID se ha usado algoritmos genéticos el cual ya ha sido probado anteriormente para otros estudios, en la presente tesis se enfoca para una planta real donde se debe considerar estrictamente las especificaciones de máximo sobreimpulso, tiempo de subida y error en estado estacionario.

El documento se divide en 4 capítulos: Capítulo 1, se presenta el estado del arte de algunos de los métodos de sintonización y modelamiento, también se muestran los índices de desempeño, se describe en la práctica cual es la problemática que se tienen en las industrias para realizar la sintonización de un lazo PID y se describe las características genéricas del proceso de concentrado de cobre. Capítulo 2, se evalúan algunos métodos existentes de modelamiento sobre un proceso teórico de segundo orden que es controlado por un PID y estos métodos son aplicados a los datos reales del proceso de inyección de agua cruda al cajón de alimentación de las bombas de ciclones para escoger cual es el método apropiado para el modelamiento. Capítulo 3, se escoge la función de aptitud y se presenta como se realiza la sintonización basado en algoritmos genéticos. Capítulo 4, se muestra los resultados de sintonización sobre datos reales tomados del proceso de inyección de agua cruda al cajón de alimentación de las bombas de ciclones y el proceso de control de nivel de espuma de una celda de flotación Rougher.

# **CAPÍTULO 1: ESTADO DEL ARTE DE LA SINTONIZACION DE LOS CONTROLADORES PID EN LA INDUSTRIA**

## **1.1 Introducción**

En el presente capítulo se expone el estado del arte de algunos de los métodos de sintonización e índices de desempeño más usados por los ingenieros de control, además se indica los problemas que tienen los ingenieros de control al realizar las sintonizaciones de los lazos de control en las industrias. También se presenta las características genéricas de una planta de concentrado y los objetivos de la tesis.

## **1.2 Estado del arte de los métodos de sintonización**

Existen varios métodos de sintonización de los cuales solo se mostrarán algunos, los dividiré en métodos empíricos y analíticos.

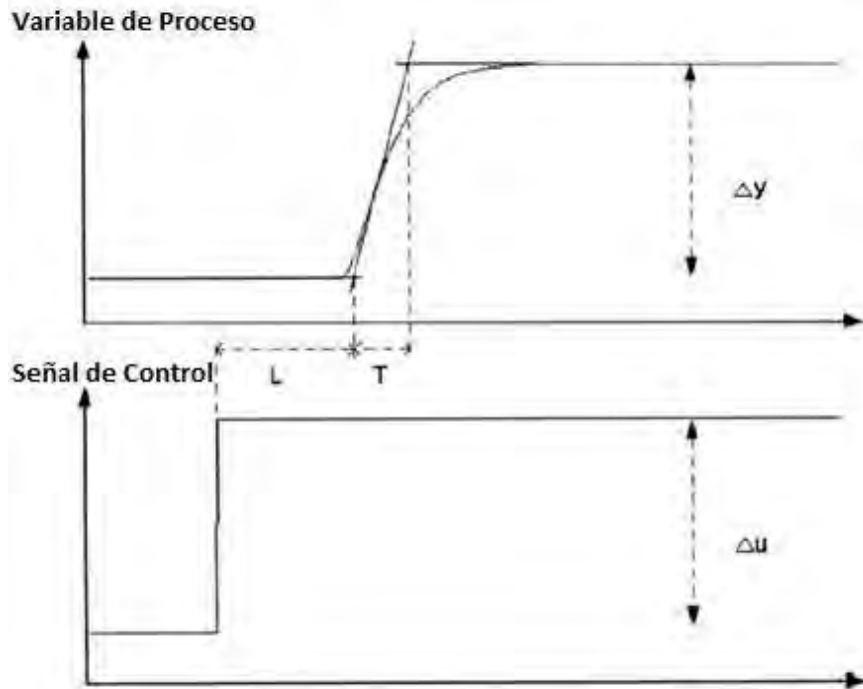
### **1.2.1 Métodos empíricos**

Son los aquellos que requieren de alteraciones en la entrada del proceso para poder llegar a una condición en la salida que permita obtener los parámetros del controlador PID.

#### **1.2.1.1 Método de Ziegler-Nichols**

Este método es ampliamente conocido por los ingenieros de control y se tienen 2 reglas propuestas por Ziegler-Nichols en 1942 y se presentan a continuación.

Método de respuesta al escalón (Kok Kiong, 1999), Este método se realiza para procesos no controlados, es decir el controlador está en manual, se genera una entrada escalonada al proceso y se obtendrá una respuesta el cual es analizada para encontrar el retardo en el tiempo (L), la constante de tiempo (T) y la ganancia estática del proceso (Kp) (figura 1.1).



**Figura 1.1:** Método de respuesta a un escalón (Kok Kion, 1999)

De esta gráfica se obtiene:

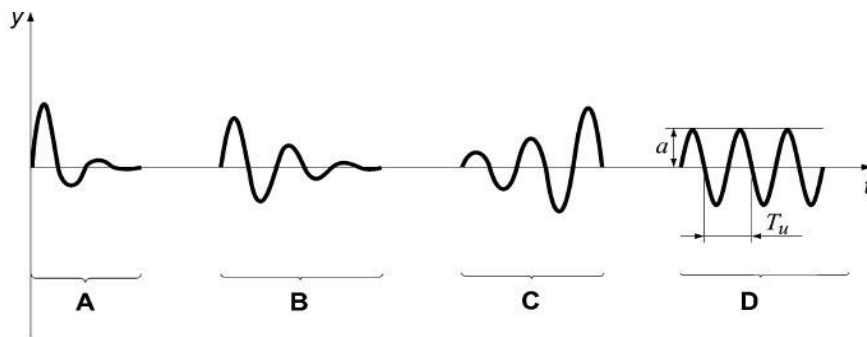
$$\theta = \frac{L}{T} \quad y \quad K_p = \frac{\Delta y}{\Delta u} \quad (1.1)$$

Con estas constantes se obtiene los parámetros del controlador PID (tabla 1.1)

Controlador	K	$T_i$	$T_d$
P	$\frac{1}{K_p \theta}$	-	-
PI	$\frac{0.9}{K_p \theta}$	3L	-
PID (Paralelo)	$\frac{1.2}{K_p \theta}$	2L	L/2
PID (Serie)	$\frac{0.6}{K_p \theta}$	L	L

**Tabla 1.1:** Parámetros PID de acuerdo a Ziegler-Nichols (Kok Kion, 1999)

Método de lazo cerrado (Boiko, 2013), Este método es puramente no paramétrico, primero el lazo cerrado debe llevarse a un estado estable, ya sea manualmente con el ajuste del elemento de control (por ejemplo, una válvula) o modificando los parámetros del controlador PID, luego se necesita quitar la acción integrativa y derivativa para poder iniciar el incremento progresivo de la constante proporcional hasta llegar a tener oscilaciones sostenidas en la salida como se puede observar en la señal D de la figura 1.2.



**Figura 1.2:** Sintonía en lazo cerrado de Ziegler-Nichols (Boiko, 2013)

Cuando el sistema tiene oscilaciones sostenidas se encuentra la ganancia crítica ( $K_u$ ) el cual es el último valor que se encuentra de  $K$  ( $K_{\text{ultimo}}$ ) y el período crítico ( $T_u$ ) que se encuentra a través de la frecuencia crítica ( $\Omega_u$ ).

$$K_u = K_{\text{ultimo}} \quad \text{y} \quad T_u = \frac{2\pi}{\Omega_u} \quad (1.2)$$

Además, Ziegler-Nichols establecieron coeficientes de sintonización para su método el cual se ve en la tabla 1.2.

Controlador	$c_1$	$c_2$	$c_3$
P	0.50	-	-
PI	0.45	0.83	-
PID	0.60	0.50	0.12

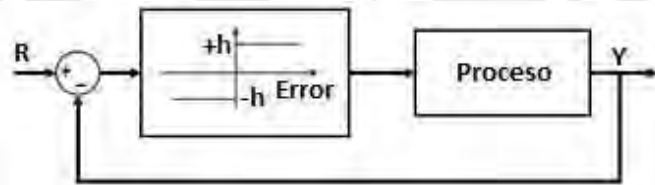
**Tabla 1.2:** Coeficientes de la regla de sintonía (Boiko, 2013)

Con todo lo ya encontrado se puede obtener los parámetros del controlador PID con las siguientes formulas.

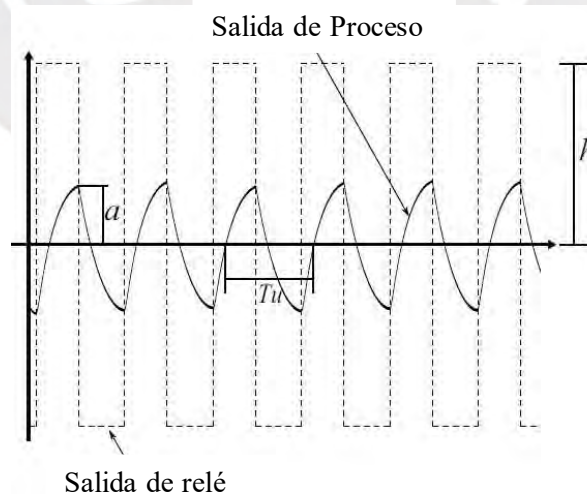
$$K_c = c_1 K_u, \quad T_i = c_2 \frac{2\pi}{\Omega_u} \quad y \quad T_d = c_3 \frac{2\pi}{\Omega_u} \quad (1.3)$$

### 1.2.1.2 Método del relé

Este método fue presentado por Astrom y Hagglund en 1984, consiste en reemplazar el controlador PID por un control todo o nada (relé ideal o un relé con histéresis) como se muestra en la figura 1.3, de esta manera pueden llevar la salida a tener oscilaciones sostenidas limitadas y a partir de esta respuesta obtener la amplitud de la salida del proceso (a) y el periodo crítico ( $T_u$ ) como se muestra en la figura 1.4.



**Figura 1.3:** Método relé en lazo cerrado (Gomery Pinto, 2014)



**Figura 1.4:** Salida del proceso (Gomery Pinto, 2014)

Con los valores obtenidos en la figura 1.4 se puede obtener la ganancia crítica ( $K_u$ ).

$$K_u = \frac{4h}{\pi a} \quad (1.4)$$

Con los valores de  $K_u$  y  $T_u$  se puede encontrar los parámetros del controlador PID usando la tabla 1.2 y la ecuación 1.3.

Los métodos de Ziegler-Nichols y de Astrom y Hagglund son ampliamente conocidos por los ingenieros de control, pero en una industria que trabaja las 24 horas del día y los 7 días de la semana no se puede aplicar estos métodos ya que generan cambios en la salida del proceso por lo que los Gerentes de las empresas no permiten realizarlo ya que puede generar pérdidas en el producto final.

### 1.2.2 Métodos analíticos

Son aquellos que requieren tener un modelo del proceso para poder calcular los parámetros del controlador PID.

Teniendo el modelo del proceso, existen varios métodos para sintonizar un controlador PID como por ejemplo el diseño por asignación de polos (Bobál y otros, 2005), diseño en el dominio de la frecuencia, enfoque basado en modelos inversos (Johnson y Moradi, 2005), algoritmos genéticos (Unal y otros, 2013), etc.

Todos los métodos de sintonización son buenos, pero no todos son adaptables a todos los tipos de sistemas sobre todo cuando no son estables en lazo abierto, uno de los métodos que si es adaptable es el basado en algoritmos genéticos debido a que es heurístico y esto lo hace robusto, además es fácil de implementar y en la práctica se requiere de la experiencia del ingeniero de control para escoger un conjunto adecuado de variables aleatorias para así encontrar la solución más óptima, esto hace que el método de algoritmos genéticos pueda ser más aceptado por los ingenieros de control, por estos motivos escogeré este método para realizar la sintonía.

### 1.2.2.1 Método de algoritmos genéticos

Los algoritmos genéticos se han utilizado para resolver una amplia gama de problemas de optimización durante los últimos treinta y siete años. Sus aplicaciones incluyen varios tipos de problemas como diseñar las redes de comunicación, optimizar la consulta de la base de datos y controlar los sistemas físicos. Así, los algoritmos genéticos se han convertido en una robusta herramienta de optimización para la resolución de problemas relacionados con diferentes campos de las ciencias técnicas y sociales (Unal y otros, 2013).

Existen varios tipos de algoritmos genéticos como:

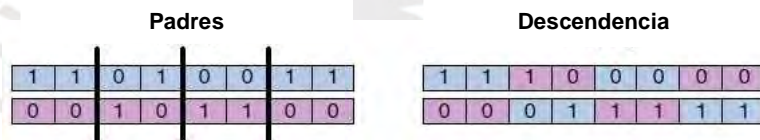
- Algoritmo genético simple.
- Algoritmo micro genético.
- Algoritmo genético de estado estable.
- Algoritmo genético jerárquico.
- Algoritmo genético desordenado.

En la presente tesis se ha trabajado con el Algoritmo genético simple el cual ya se conoce que es muy útil para la minimización de funciones y no toma mucho tiempo en la evaluación debido a que no se trabaja con una gran población de valores.

Para entender mejor la técnica de los algoritmos genéticos primero se tiene que tomar en cuenta definición de los siguientes términos:

- Función de aptitud, es la función que debe de ser optimizada ya sea maximizándolo o minimizándolo para encontrar la respuesta esperada en un proceso.
- Población de individuos, es el conjunto de posibles soluciones que podrían minimizar o maximizar la función de aptitud.
- Tamaño de la población, es la cantidad de individuos que utilizaran para buscar la solución óptima.

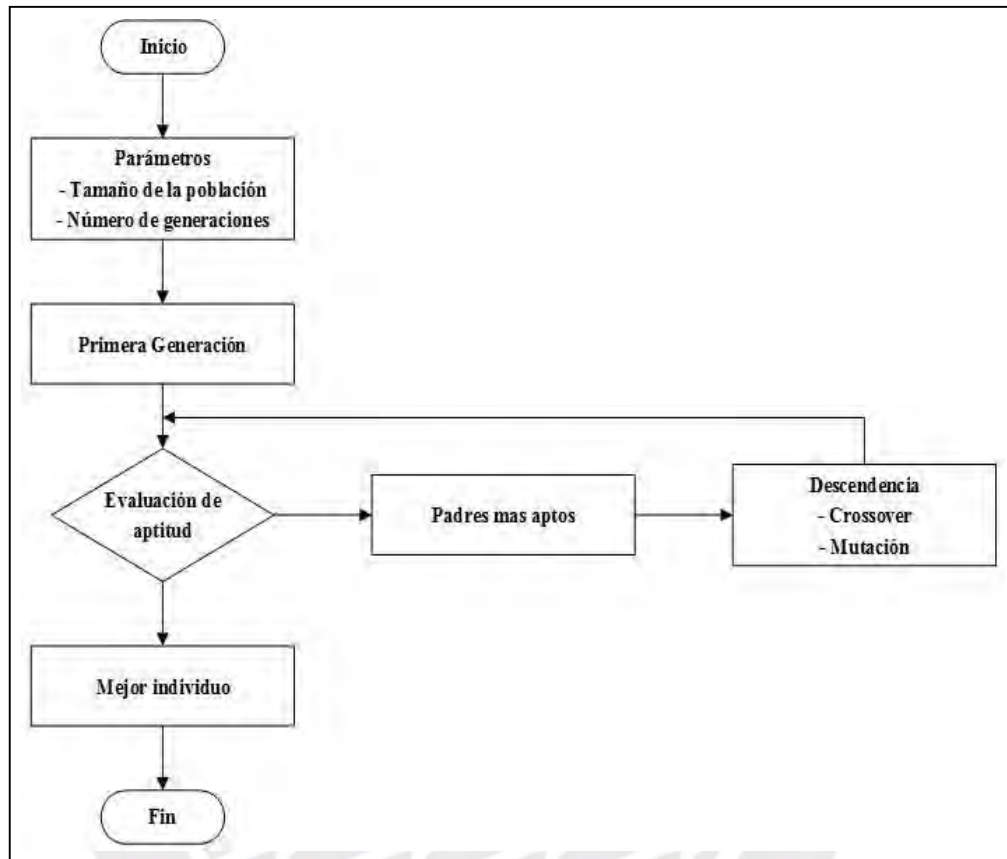
- Primera generación, es el primer conjunto de soluciones que podrían llevar a la optimización de la función de aptitud.
- Evaluación de aptitud, son los resultados de la función de aptitud por cada individuo.
- Padres más aptos, son los individuos que consiguieron los mejores resultados en la evaluación de aptitud.
- Descendencia, es la siguiente generación que está conformado por el conjunto total de crossover y mutaciones, estos vuelven a ejecutar la evaluación de aptitud.
  - Crossover, es la descendencia que se obtienen al realizar combinaciones de los padres, por ejemplo, si tenemos 2 padres y cada uno es de 8 bits la descendencia puede ser una combinación como se aprecia en la figura 1.5.



**Figura 1.5:** Crossover (Unal y otros, 2013)

- Mutación, es la descendencia que se obtiene de valores aleatorios del conjunto de posibles soluciones.
- Número de generaciones, es la cantidad máxima de descendencias que se tendrán, cada descendencia debe ejecutar la evaluación de aptitud, también se le conoce como número de interacciones.
- Criterio de finalización, es la condición que se debe cumplir para dejar de generar descendencias es decir para dejar de buscar al individuo que presente la aptitud deseada, esta condición puede ser cuando se llegue a la solución óptima o cuando se cumpla con el número de generaciones.
- Mejor individuo, es la solución que logro el mejor resultado para la función de aptitud la cual se encuentra luego que se ejecuta el criterio de finalización.

Teniendo en cuenta las definiciones anteriores, se presenta en la figura 1.6 el diagrama de flujo de la técnica de algoritmos genéticos que se usara en la presente tesis.

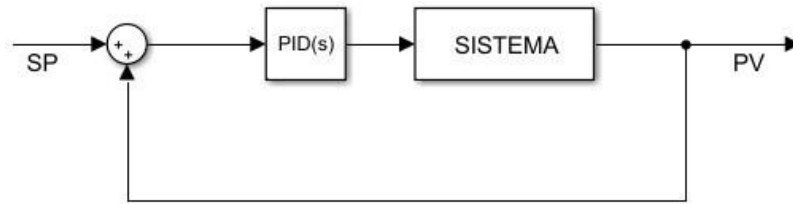


**Figura 1.6:** Diagrama de flujo de la técnica de algoritmos genéticos

Esta técnica de algoritmos genéticos se usa para varios tipos de problemas de optimización por lo que también puede usarse para realizar la sintonización de los parámetros de un controlador PID, esto ya ha sido estudiado en muchas ocasiones por ejemplo para el control de un tanque de nivel de presión ajustando el flujo de aire de entrada (Unal y otros, 2013), para el control de un Ball and Beam (Ortiz, 2014), para el control de un servo electro hidráulico de accionamiento directo (Cao, 2018) y para un Cuadrícóptero (Alkamachi, 2017). Lo que se puede encontrar en estos estudios es que es muy importante contar con un buen modelo del proceso y también una adecuada función de aptitud la cual refleje las especificaciones del lazo de control SISO como el máximo sobreimpulso, tiempo de subida y error en estado estacionario.

### 1.3 Modelamiento de los procesos

Para poder realizar la sintonización de un controlador PID con el método de algoritmos genéticos es de vital importancia tener el modelo del proceso, pero ¿exactamente qué es lo debemos de modelar? En la figura 1.7 se muestra el diagrama de bloques clásico de un control PID.



**Figura 1.7:** Diagrama de bloques de un control PID

El bloque sistema es lo que se debe modelar y este se compone por el actuador, el proceso, las perturbaciones, el sensor y ruidos en el sensor como se puede ver en la figura 1.8.



**Figura 1.8:** Sistema que debe identificarse (Tangirala, 2015)

Existen muchos métodos para poder modelar este sistema, pero como en la práctica, en una planta concentradora de cobre, no es posible abrir sus lazos de control, para realizar el modelamiento es necesario que todo se realice en lazo cerrado, a continuación, se explican los métodos que se utilizarán para la presente tesis.

### 1.3.1 Modelo teórico

Esta clasificación se basa en la naturaleza matemática de la suposición hecha sobre un sistema (Tangirala, 2015), es necesario tener los parámetros de diseño y conocimiento de las leyes de la física que describen el proceso para poder realizar el modelamiento.

Por ejemplo, para un proceso común como el control de nivel de agua en un tanque se tiene que la salida es la altura y la entrada es el flujo de agua el cual es controlado por una válvula. Para este proceso el modelo teórico está representado por la siguiente ecuación.

$$A_c \frac{\partial h(t)}{\partial t} = F_i(t) - C_v \sqrt{h(t)} \quad (1.5)$$

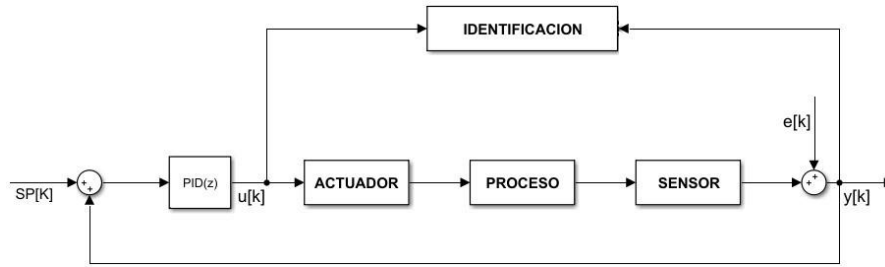
Donde:

- $h(t)$  → Nivel del líquido
- $F_i(t)$  → Flujo de agua de ingreso
- $A_c$  → Área de la sección transversal del tanque
- $C_v$  → Coeficiente de la válvula

### 1.3.2 Identificación del sistema

Es la obtención del modelo matemático del sistema de forma experimental para lo cual se usa los datos reales de una planta, como estos datos son obtenidos por muestreo el modelo se presenta en el dominio discreto.

Los datos que se requieren muestrear para obtener el modelo son la entrada y la salida del sistema. Como ya se indicó es necesario hacer la identificación en lazo cerrado y además se debe considerar las perturbaciones en el proceso los cuales están incluidos dentro del muestreo de la salida, el diagrama de bloques para esta identificación se muestra en la figura 1.9.



**Figura 1.9:** Diagrama de bloques para identificación de sistemas

Para el bloque de identificación existen varias estructuras de modelo que se pueden utilizar, estos se llaman modelos de errores de ecuación (Tangirala, 2015), la siguiente es la estructura genérica de estos modelos.

$$A(q^{-1})y[k] = \frac{B(q^{-1})}{F(q^{-1})} u[k] + \frac{C(q^{-1})}{D(q^{-1})} e[k] \quad (1.6)$$

Donde:

- $A(q^{-1}) = 1 + a_1 q^{-1} + \dots + a_{n_a} q^{-n_a}$
- $B(q^{-1}) = b_{n_k} q^{-n_k} + b_{n_k+1} q^{-n_k-1} + \dots + b_{n_k+n_b-1} q^{-(n_b+n_k-1)}$
- $C(q^{-1}) = 1 + c_1 q^{-1} + \dots + c_{n_c} q^{-n_c}$
- $D(q^{-1}) = 1 + d_1 q^{-1} + \dots + d_{n_d} q^{-n_d}$
- $F(q^{-1}) = 1 + f_1 q^{-1} + \dots + f_{n_f} q^{-n_f}$

Para poder encontrar una estructura que refleje el comportamiento dinámico del proceso que se muestra en la figura 1.9, es necesario determinar adecuadamente cual es el orden de los polinomios ( $n_a$ ,  $n_b$ ,  $n_c$ ,  $n_d$ ,  $n_f$ ), el retardo  $n_k$  y los coeficientes de los polinomios ( $a_i$ ,  $b_i$ ,  $c_i$ ,  $d_i$ ,  $f_i$ ). Estos parámetros son sencillos de encontrar si se tiene un conocimiento del diseño del proceso, pero en la práctica no se cuenta con esta información, por tal motivo encontrar los parámetros de la estructura de modelo podría convertirse en una tarea tediosa ya que es necesario probar con distintos valores para los parámetros. Existen estructuras que son una simplificación de la estructura genérica, a continuación, se detallan algunas.

### 1.3.2.1 Modelo ARX (exógeno autorregresivo)

Se obtiene cuando  $C=D=F=1$ , reemplazando en la ecuación 1.6 se obtiene.

$$y[k] = \frac{B(q^{-1})}{A(q^{-1})}u[k] + \frac{1}{A(q^{-1})}e[k] \quad (1.7)$$

### 1.3.2.2 Modelo ARMAX (exógeno autorregresivo de media móvil)

Es una extensión de la estructura ARX por la inclusión del término de media móvil en el modelo (Tangirala, 2015). Este se obtiene cuando  $D=F=1$ , reemplazando en la ecuación 1.6 se obtiene.

$$y[k] = \frac{B(q^{-1})}{A(q^{-1})}u[k] + \frac{C(q^{-1})}{A(q^{-1})}e[k] \quad (1.8)$$

### 1.3.2.3 Modelo OE (error de salida)

Como sugiere su nombre, asume que el ruido blanco afecta directamente a la salida, es decir no intenta modelar las características del ruido (Tangirala, 2015). Este se obtiene cuando  $A=C=D=1$ , reemplazando en la ecuación 1.6 se obtiene.

$$y[k] = \frac{B(q^{-1})}{F(q^{-1})}u[k] + e[k] \quad (1.9)$$

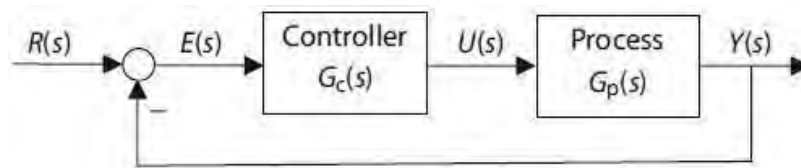
### 1.3.2.4 Modelo BJ (Box Jenkins)

Es una de las estructuras más generales (Tangirala, 2015). Este se obtiene cuando  $A=1$ , reemplazando en la ecuación 1.6 se obtiene.

$$y[k] = \frac{B(q^{-1})}{F(q^{-1})}u[k] + \frac{C(q^{-1})}{D(q^{-1})}e[k] \quad (1.10)$$

### 1.3.3 Identificación de sistemas en lazo cerrado

Para poder realizar esta identificación se tomará en cuenta el diagrama de bloques en lazo cerrado de control, esta se muestra en la figura 1.10.



**Figura 1.10:** Diagrama de bloques en lazo cerrado (Johnson y Moradi, 2005).

Donde:

- $G_c(s)$  → Función de transferencia del controlador
- $G_p(s)$  → Función de transferencia del proceso
- $R(s)$  → Entrada del sistema en lazo cerrado
- $E(s)$  → Entrada del controlador, también llamado error
- $U(s)$  → Entrada del proceso, también llamado variable manipulada
- $Y(s)$  → Salida del proceso y del sistema en lazo cerrado

Se debe tomar en cuenta que el bloque proceso que se muestra en la figura 1.10 corresponde al sistema que debe identificarse y se muestra en la figura 1.8.

Hay 2 métodos que serán discutidos a continuación (Johnson y Moradi, 2005).

- Identificación de un sistema desconocido en lazo cerrado con un controlador desconocido, este método es una identificación de una caja negra y se utiliza cuando no se tiene ningún conocimiento del proceso ni del controlador.
- Identificación de un sistema desconocido en lazo cerrado con un controlador conocido, este método se utiliza cuando se conoce la función de transferencia del controlador y es necesario identificar el proceso.

En la práctica, en los lazos de control de una planta concentradora de cobre y en general en toda industria, siempre se conoce la función de transferencia del controlador PID que viene configurado en el DCS o PLC, por tal motivo solo se detallará el segundo método.

### 1.3.3.1 Identificación de un sistema desconocido en lazo cerrado con un controlador conocido

Tomando en consideración el diagrama de bloques en lazo cerrado mostrado en la figura 1.10, se tiene que la función de transferencia en lazo cerrado  $G_{CL}(s)$  viene dada por la siguiente ecuación.

$$G_{CL}(s) = \frac{Y(s)}{R(s)} = \frac{G_c(s)G_p(s)}{1 + G_c(s)G_p(s)} \quad (1.11)$$

De donde se tiene que la función de transferencia del controlador  $G_c(s)$  es conocido y la función de transferencia en lazo cerrado  $G_{CL}(s)$  se puede identificar ya que se puede muestrear los datos de entrada y salida del sistema en lazo cerrado del DCS o del PLC, tomando esto en consideración se puede despejar la función de transferencia del proceso  $G_p(s)$ .

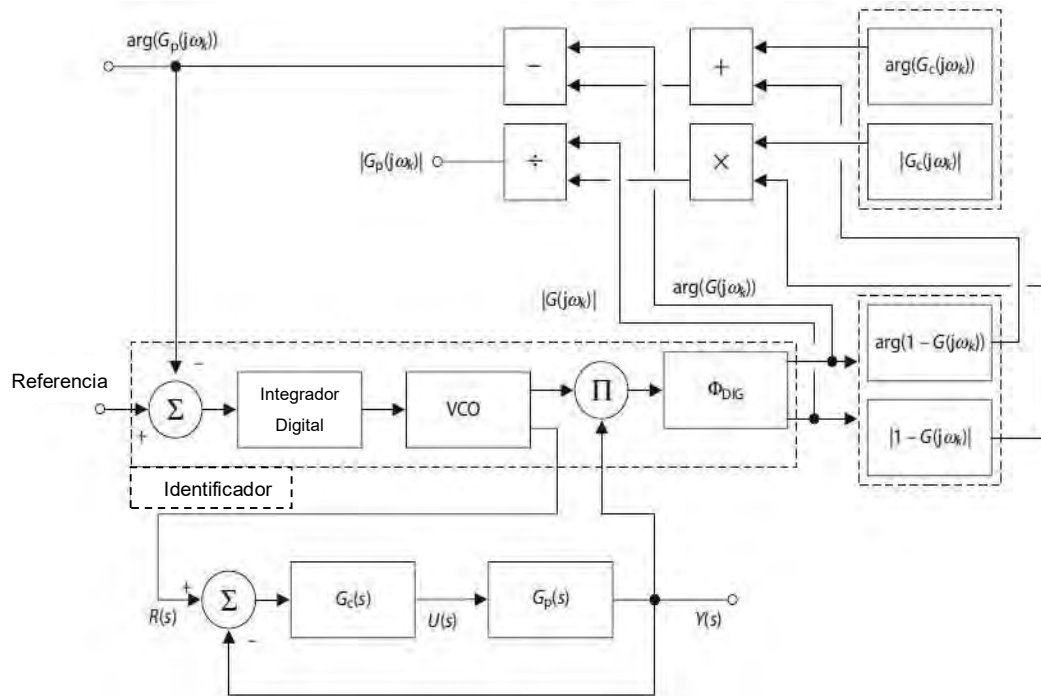
$$G_p(s) = \frac{G_{CL}(s)}{G_c(s)(1 - G_{CL}(s))} \quad (1.12)$$

Considerando que  $s = j\omega$  y que se conoce  $G_c(s)$  y  $G_{CL}(s)$ , se puede obtener el margen de ganancia y de fase de  $G_p(s)$  los cuales vienen dado por las siguientes ecuaciones.

$$|G_p(j\omega)| = \frac{|G_{CL}(j\omega)|}{|G_c(j\omega)|(1 - G_{CL}(j\omega))|} \quad (1.13)$$

$$\arg(G_p(j\omega)) = \arg(G_{CL}(j\omega)) - \arg(G_c(j\omega)) - \arg(1 - G_{CL}(j\omega)) \quad (1.14)$$

En la figura 1.11 se muestra cómo se realiza la identificación considerando las ecuaciones 1.13 y 1.14.

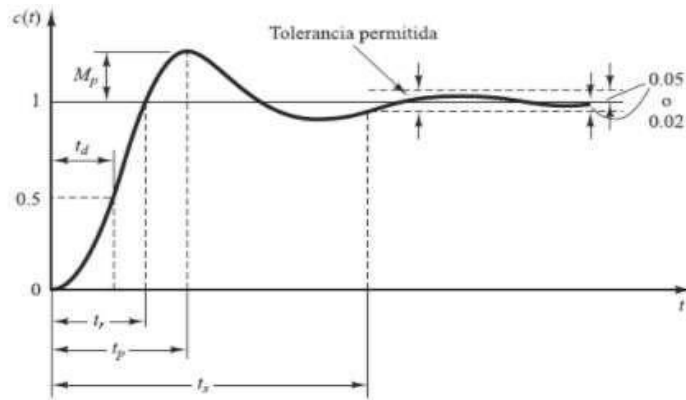


**Figura 1.11:** Identificación en lazo cerrado (Johnson y Moradi, 2005).

Una vez que se tiene identificado el margen de ganancia y de fase ya se puede diseñar el controlador PID de acuerdo con las especificaciones de diseño en el dominio de la frecuencia. En el libro “PID Control New Identification and Design Methods” (Johnson y Moradi, 2005) se puede encontrar algunos métodos de diseño de un controlador PID en el dominio de la frecuencia.

#### 1.4 Estado del arte de los índices de desempeño

Para poder saber si un lazo cerrado de control como el mostrado en la figura 1.10 tiene en su salida una respuesta transitoria adecuada cuando ha cambiado la entrada del sistema de lazo cerrado (set point), es necesario establecer por cada lazo de control de la planta cuáles son los límites que se permiten para sus especificaciones de la respuesta transitoria, estas especificaciones se muestran en la figura 1.12



**Figura 1.12:** Respuesta a escalón unitario (Ogata, 2010).

Donde:

- $M_p \rightarrow$  Máximo sobreimpulso
- $t_r \rightarrow$  Tiempo de subida
- $t_s \rightarrow$  Tiempo de establecimiento
- $e_{es} \rightarrow$  Error en estado estacionario (error cuando  $t > t_s$ )

También se puede usar los índices de desempeño para evaluar si el lazo de control trabaja de forma adecuada, la forma clásica de la evaluación de desempeño es el basado en el error integral (Juliani, 2012), estos índices son los siguientes.

$$\text{(Error cuadrático integral) } ISE = \int_{t_0}^{t_F} e(t)^2 dt \quad (1.15)$$

$$\text{(Error absoluto integral) } IAE = \int_{t_0}^{t_F} |e(t)| dt \quad (1.16)$$

$$\text{(Error cuadrático de tiempo integrado) } ITSE = \int_{t_0}^{t_F} t \cdot e(t)^2 dt \quad (1.17)$$

$$\text{(Error absoluto de tiempo integrado) } ITAE = \int_{t_0}^{t_F} t \cdot |e(t)| dt \quad (1.18)$$

Otra forma de evaluar el desempeño del lazo de control es usando el índice de desempeño normalizado (Desborough y Harris, 1992), cuando su valor es más cercano a 0 quiere decir que el lazo tiene una buena sintonización. Este índice se representa por la siguiente ecuación.

$$\hat{n}(b) = 1 - \frac{n - b - m + 1}{n - b - 2m + 1} \frac{(\hat{\mathbf{y}} - \bar{\mathbf{X}}\hat{\boldsymbol{\alpha}})^T (\hat{\mathbf{y}} - \bar{\mathbf{X}}\hat{\boldsymbol{\alpha}})}{\hat{\mathbf{y}}^T \hat{\mathbf{y}} + \hat{\mathbf{y}}^2} \quad (1.19)$$

El detalle de cada variable y vector se encuentra en el paper “Performance Assessment Measures for Univariate Feedback Control” (Desborough y Harris, 1992).

Las especificaciones de la respuesta transitoria y los índices de performance que se han detallado serán evaluadas en el capítulo 2.

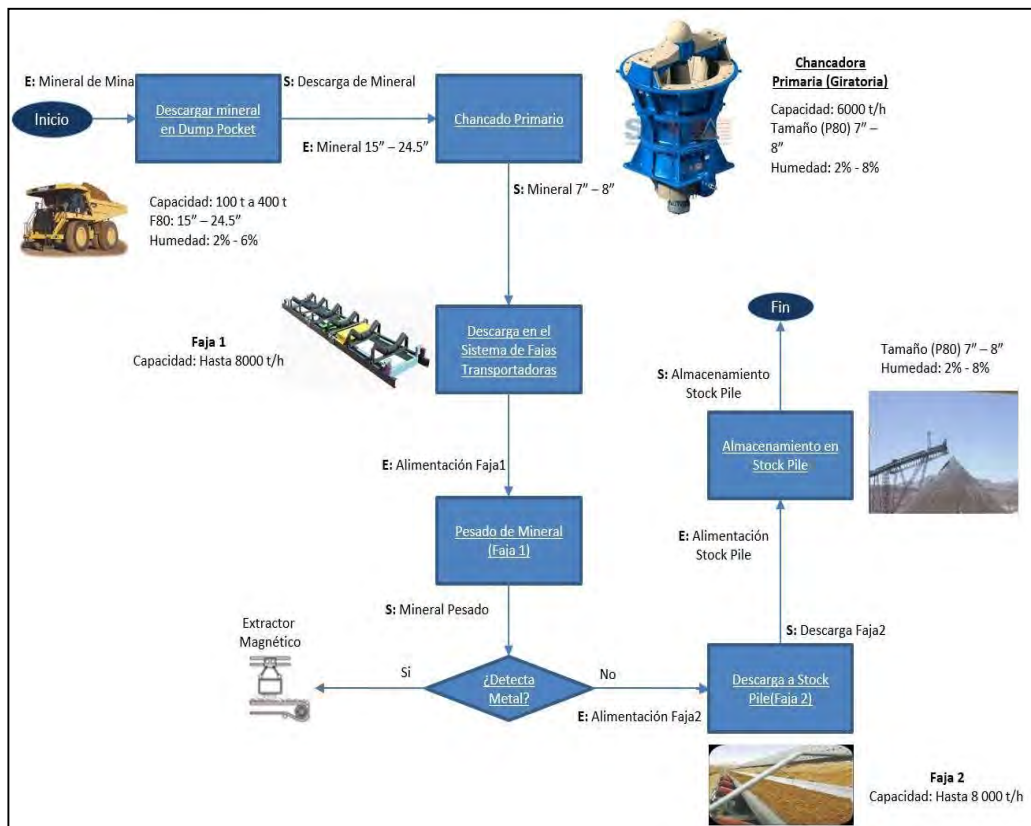
## 1.5 Características genéricas del proceso de concentrado de cobre

En la presente tesis se va a elaborar un código basado en el método de sintonización por algoritmos genéticos y se va a probar si este puede sintonizar lazos de control de una planta concentradora de cobre, para esto se ha conseguido datos reales de 2 lazos de control y es necesario saber la importancia de estos lazos de control dentro del proceso de concentrado de cobre. A continuación, se describirá de forma genérica los procesos de una planta de concentrado de cobre y se indicará dónde están ubicados estos lazos de control.

### 1.5.1 Proceso de Chancado

Después del fracturamiento del mineral con el proceso de voladura este es llevado hacia la cancha de almacenamiento para iniciar el proceso de chancado, el cual permite reducir de tamaño el mineral mediante equipos de trituración hasta un tamaño deseado para el ingreso al proceso de molienda. El proceso de chancado puede tener varias etapas y el tamaño final varía de acuerdo al tipo de mineral que se tiene en cada proceso minero y se define por el P80 (tamaño del tamiz que deja pasar el 80% de las partículas

del producto). En la figura 1.13 se muestra el diagrama de flujo de un proceso genérico de Chancado indicando las variables más importantes, en donde se ha considerado una chancadora primaria y un sistema de 2 fajas para transportar el mineral al Stock Pile. La función principal de esta etapa es la de generar mineral con tamaños inferiores a 8 o 7 pulgadas y almacenar masa mineral que pueda abastecer constantemente a los procesos posteriores.

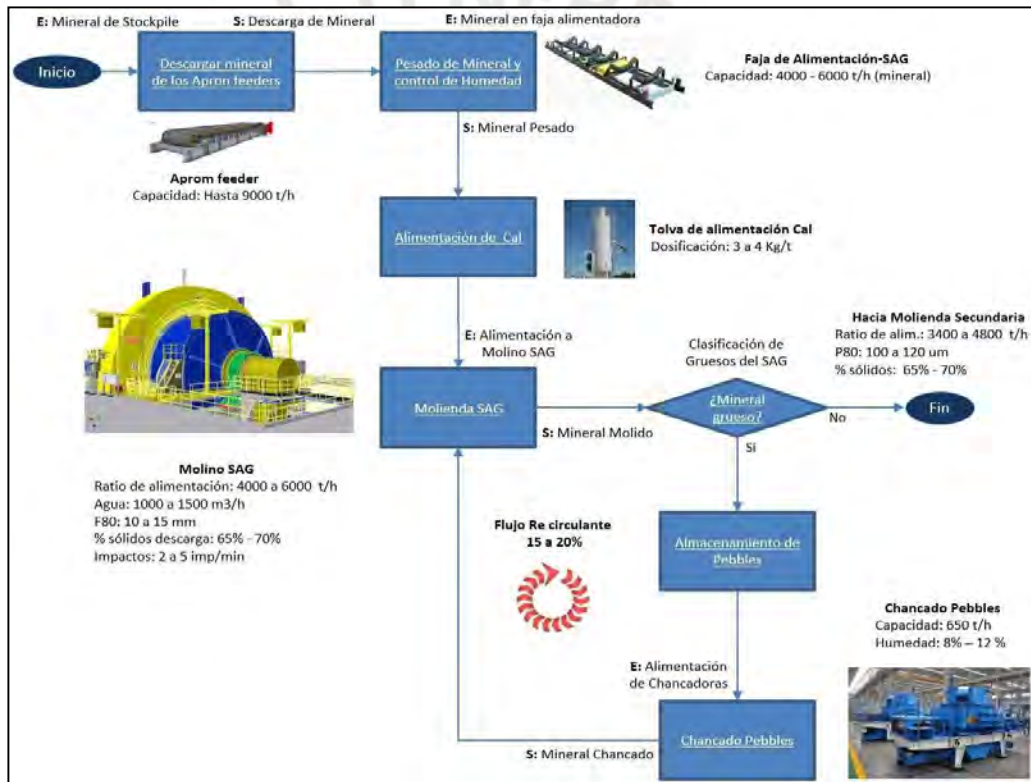


**Figura 1.13:** Diagrama de flujo de un proceso genérico de Chancado.

## 1.5.2 Proceso de Molienda

El mineral que se encuentra en el stock pile es ingresado al proceso de Molienda el cual permite reducir el tamaño del mineral mediante Molinos, Chancadoras y clasificadores hasta la granulometría que se desea ingresar al proceso de Flotación. En el proceso de Molienda se pueden identificar 2 etapas las cuales se explicarán a continuación.

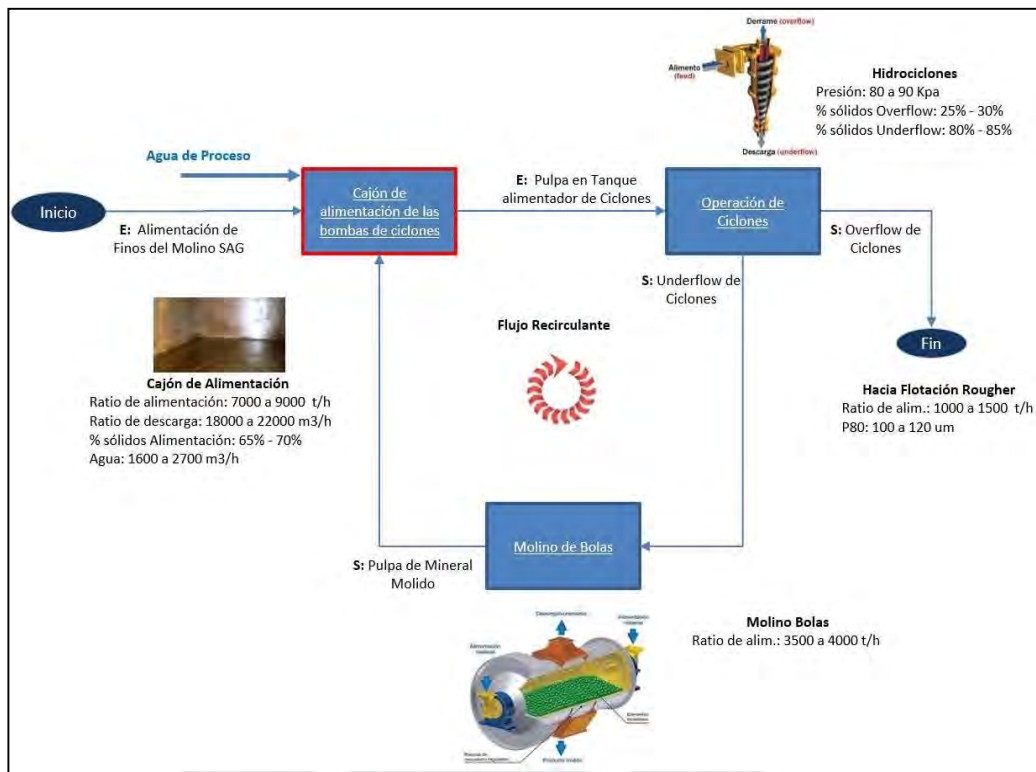
**Molienda primaria (etapa 1)**, abarca desde la alimentación del molino SAG (Molienda semiautógena) hasta la clasificación seca. En la figura 1.14 se muestra el diagrama de flujo de la primera etapa de un proceso genérico de molienda indicando las variables más importantes, en donde se ha considerado un molino SAG, una chancadora de Pebbles para el flujo recirculante y una zaranda vibratoria para la clasificación seca. Tomar en cuenta que las variables que determinan una mayor capacidad de molienda son la alimentación de mineral, la alimentación de agua y la carga moledora (bolas de acero y Pebbles), por lo que es necesario un correcto control de estas variables para aumentar la eficiencia del grado de molienda.



**Figura 1.14:** Diagrama de flujo de la primera etapa de un proceso genérico de Molienda.

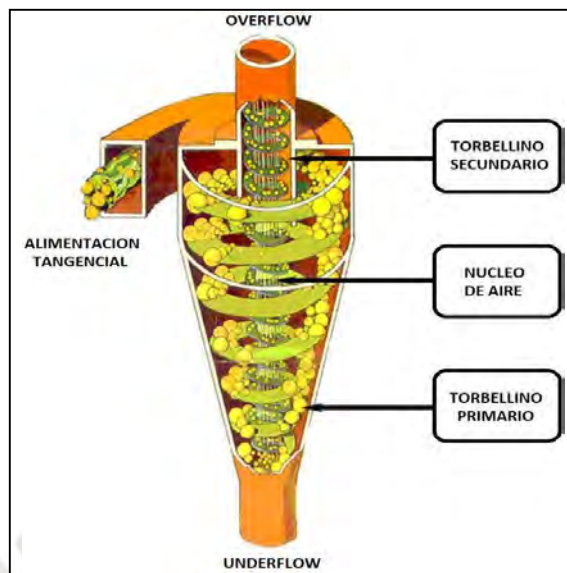
**Molienda secundaria (etapa 2)**, abarca desde el cajón de alimentación de las bombas de ciclones hasta la clasificación húmeda. En la figura 1.15 se muestra el diagrama de flujo de la segunda etapa de un proceso genérico de molienda secundaria indicando las

variables más importantes, en donde se ha considerado hidrociclones para la clasificación húmeda y un molino de bolas para la remolienda del flujo recirculante. Además, se ha marcado en un rectángulo rojo el “cajón de alimentación de las bombas de ciclones” en el cual se encuentra el primer lazo de control en el que se realizará pruebas de sintonía con datos reales, este lazo controla el flujo de agua de ingreso al cajón.



**Figura 1.15:** Diagrama de flujo de la segunda etapa de un proceso genérico de Molienda Secundaria.

En la figura 1.16 se muestra el principio de funcionamiento de los hidrociclones el cual consiste en realizar la separación de las partículas gruesas y finas por suspensión en rotación, la fuerza centrífuga que se genera al interior del hidrociclón realiza una sedimentación de las partículas gruesas las cuales son arrastradas por un torbellino primario a lo largo de la superficie cilíndrica y cónica hacia la descarga (Underflow) y las partículas finas salen por la parte superior (Overflow) formando un torbellino secundario.



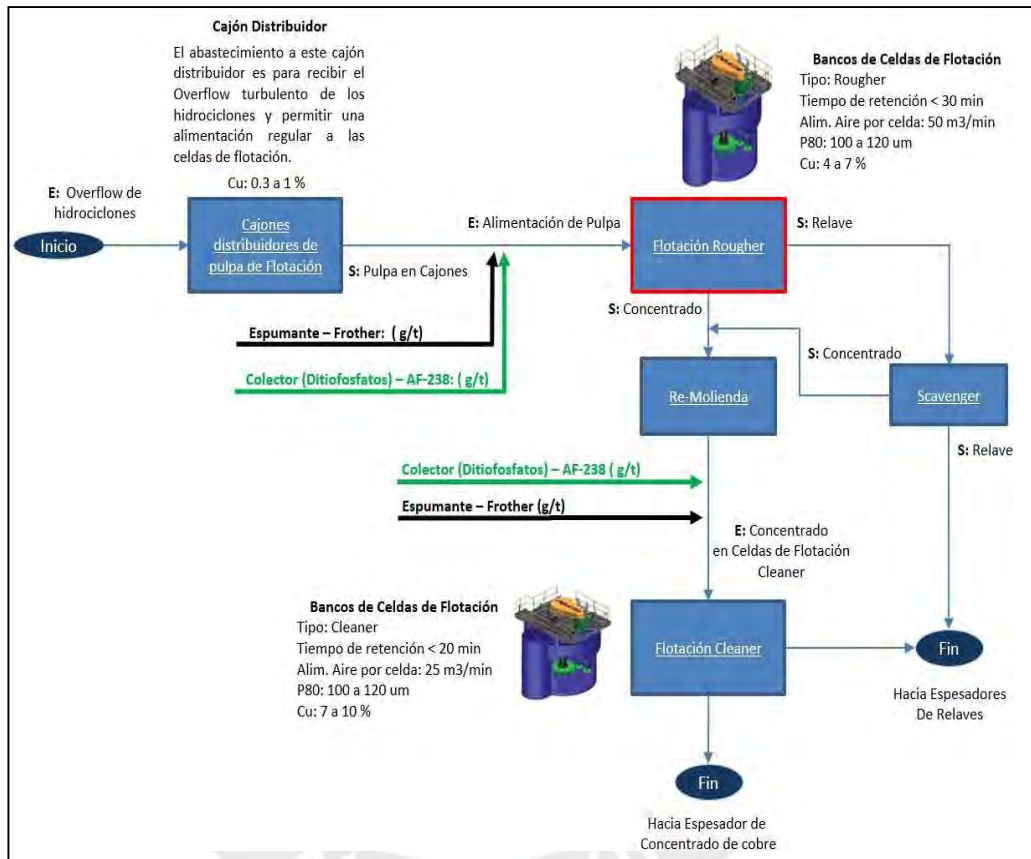
**Figura 1.16:** Funcionamiento del Hidrociclón

Para poder obtener un adecuado porcentaje de sólidos en el Overflow del hidrociclón es necesario controlar la adición de agua, el nivel del cajón y la densidad de la pulpa en el “cajón de alimentación de las bombas de ciclones”. Por este motivo se ha obtenido los datos históricos de estas 3 variables encontrando que el control de adición de agua no era el adecuado ya que demora aproximadamente 3 minutos en llegar al valor deseado cuando es necesario aumentar o disminuir el flujo de ingreso de agua para mantener un valor entre el 25% y 30% del porcentaje de sólidos en el Overflow del hidrociclón. Por lo indicado se ha considerado realizar en este lazo de control la sintonización del controlador PID basado en algoritmos genéticos lo cual es el objeto de estudio de la presente tesis.

### 1.5.3 Proceso de Flotación

La pulpa de mineral que sale del Overflow de los hidrociclones ingresa al proceso de Flotación el cual separa el mineral de cobre de los materiales sin valor mediante celdas de Flotación Rougher, Cleaner y Scavenger hasta obtener una pulpa con mayor ley de cobre que ingresará al proceso de Espesamiento y Filtrado. En la figura 1.17 se muestra el diagrama de flujo de un proceso genérico de Flotación indicando las variables más

importantes, en donde se ha considerado las 3 etapas de flotación (Rougher, Cleaner y Scavenger) y un proceso de remolienda. Además, se ha marcado en un rectángulo rojo el “proceso de Flotación Rougher” en el cual se encuentra el segundo lazo de control en el que se realizara pruebas de sintonía con datos reales, este lazo controla el nivel de espuma de la primera celda de flotación.



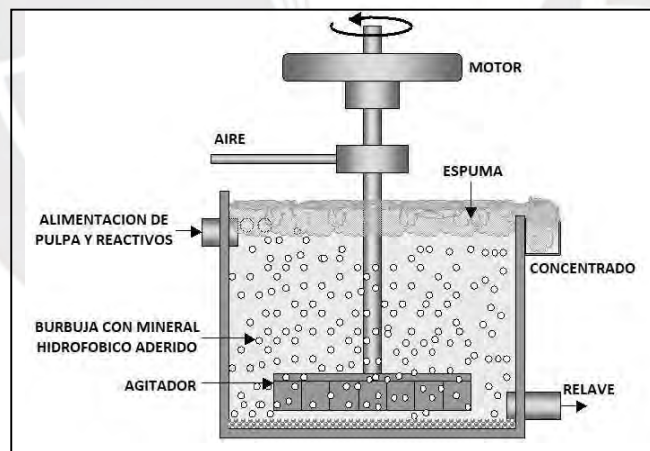
**Figura 1.17:** Diagrama de flujo de un proceso genérico de Flotación.

En la figura 1.18 se muestra el principio de funcionamiento de la flotación Rougher el cual consiste en obtener una alta recuperación del concentrado de cobre separándolo del relave, para esto es importante controlar los siguientes elementos:

- Adición de reactivos, es el componente más importante que permite realizar una adecuada flotación del mineral el cual se clasifica en colectores (compuestos químicos que vuelven al mineral hidrofóbico para que pueda adherirse a las

burbujas de aire y así puedan ascender formando la espuma), espumantes (sustancias orgánicas que permiten formar una espuma estable de aire y agua) y agentes reguladores (sustancias inorgánicas que facilitan la absorción de los reactivos sobre las partículas minerales).

- Ingreso de aire, permite la formación de burbujas de aire con ayuda del agitador.
- Nivel de espuma, ayuda a obtener una alta recuperación de cobre y esto se logra manteniendo un determinado rango de nivel del colchón de espuma en el cual se debe trabajar, este rango es calculado de acuerdo al diseño de la planta y a la composición del mineral que ingresa a la chancadora primaria. Mantener este rango de nivel es muy importante ya que si supera el límite superior disminuye el tiempo de permanencia de la espuma en la celda lo que ocasiona que el concentrado de cobre arrastre material estéril y si baja del límite inferior aumenta el tiempo de permanencia de la espuma en la celda lo que ocasiona que se pierda mineral ya que la espuma empieza a caer. Por lo indicado se ha considerado realizar en este lazo de control la sintonización del controlador PID basado en algoritmos genéticos lo cual también es el objeto de estudio de la presente tesis.

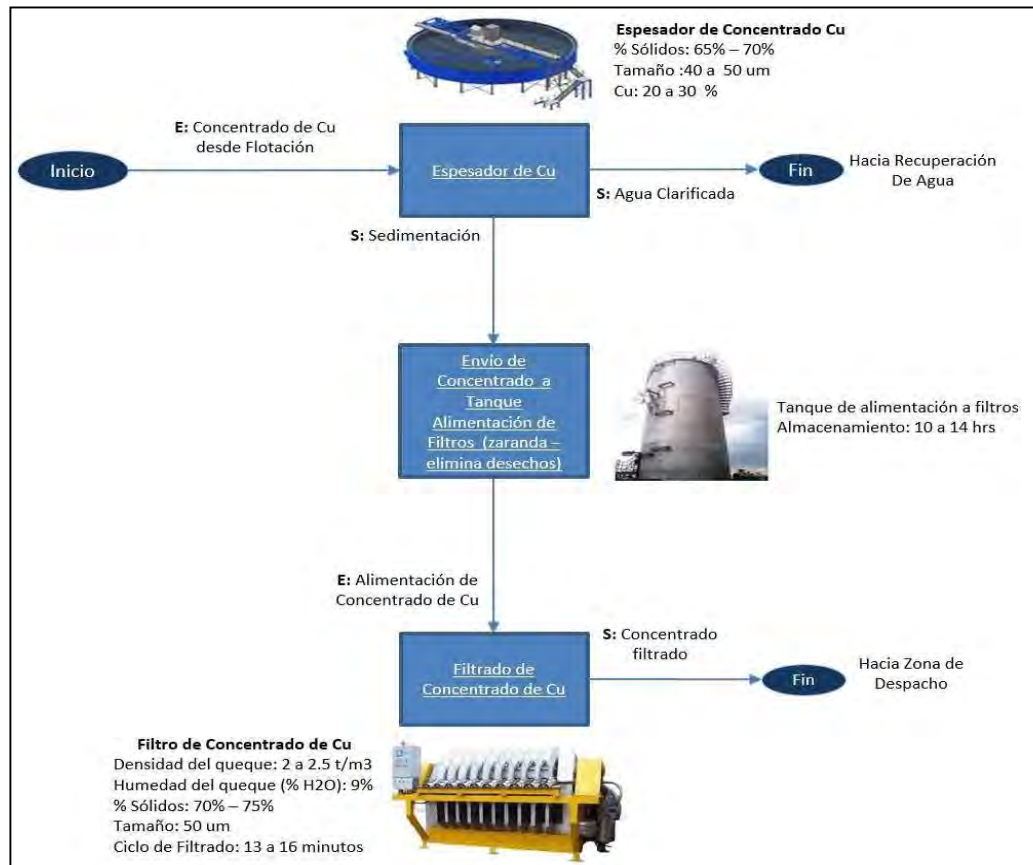


**Figura 1.18:** Funcionamiento de la celda de Flotación

#### 1.5.4 Proceso de Espesamiento y Filtrado

El concentrado que se obtiene del proceso de flotación ingresa al proceso de espesamiento y filtrado el cual realiza la separación de sólido/líquido hasta obtener un

concentrado de baja humedad que se almacenara para poder ser despachado. En la figura 1.19 se muestra el diagrama de flujo de un proceso genérico de Espesamiento y Filtrado indicando las variables más importantes, en donde se ha considerado un espesador de cobre y filtros prensa.



**Figura 1.19:** Diagrama de flujo de un proceso genérico de Espesamiento y Filtrado.

## 1.6 Objetivos de la tesis

En una planta concentradora se tiene que en general los controladores PID no están bien sintonizados, además como las plantas trabajan las 24 horas del día y los 7 días de la semana no se tiene el permiso de los Gerentes de las empresas para poder realizar cambios si es que no se demuestra que el control del proceso va a mejorar, debido a esto los ingenieros de control no realizan cambios en los parámetros de los controladores PID, incluso se encuentra que varios de estos no han sido modificados

desde que la planta empezó a operar. Por tal motivo es necesario implementar un método de sintonización que no altere el proceso mientras se busca los parámetros óptimos del controlador PID, para que de esta manera los Gerentes de las empresas no tengan temor en realizar cambios que permitan mejorar el proceso. por este motivo se ha establecido los siguientes objetivos.

### **1.6.1 Objetivo general**

“Sintonización de un controlador PID utilizando algoritmos genéticos aplicada a una planta concentradora de cobre”.

### **1.6.2 Objetivos específicos**

Para poder alcanzar el objetivo general se requiere los siguientes objetivos específicos:

- Escoger un método de modelamiento que pueda implementarse para varios tipos de lazos cerrados de control y permita representar adecuadamente un proceso.
- Evaluar los distintos índices de desempeño de los lazos de control.
- Diseñar y sintonizar un controlador PID usando el método de algoritmos genéticos.
- Analizar las condiciones actuales en el cajón de alimentación de las bombas de ciclones y en la primera celda de flotación Rougher.
- Simulación y aplicación del método de modelamiento y sintonización por algoritmos genéticos del proceso real de inyección de agua cruda al cajón de alimentación de las bombas de ciclones.
- Simulación del método de modelamiento y sintonización por algoritmos genéticos del proceso real de control de nivel de espuma de la primera celda de flotación Rougher.

## **CAPÍTULO 2: MODELAMIENTO DE UN PROCESO EN UN SISTEMA DE LAZO CERRADO**

### **2.1 Introducción**

Como ya se explicó en el capítulo 1 en la práctica, en una planta concentradora de cobre, no es posible abrir sus lazos de control, para realizar el modelamiento es necesario que todo se realice en lazo cerrado. Adicionalmente se debe tomar en cuenta que realizar un modelo teórico no es recomendable ya que generalmente no se cuenta con los parámetros de diseño y, no es posible medir fácilmente el desgaste de los equipos y el cambio en los procesos que afectan dicho modelo.

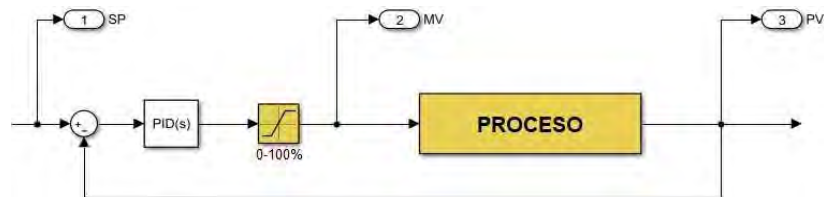
En el presente capítulo se realizará lo siguiente. Primero, se evaluará 3 métodos de identificación sobre un proceso teórico de flujo de segundo orden que trabaja con una válvula de diafragma y es controlado por un PID, se realiza las simulaciones para verificar la aproximación del modelo. Segundo, los métodos de identificación son aplicados a los datos reales del proceso de inyección de agua cruda al cajón de alimentación de las bombas de ciclones, se realiza las simulaciones para escoger el método de identificación. Por último, se presenta las conclusiones del capítulo 2.

### **2.2 Método de identificación en lazo cerrado sobre un modelo teórico**

Para poder evaluar los métodos de identificación, se debe tener en cuenta que en la práctica se cuenta en general con la siguiente información por cada lazo de control.

- SP → Set point del lazo de control
- MV → Variable manipulada del lazo de control
- PV → Variable de proceso del lazo de control
- $K_p$  → Constante proporcional del controlador PID
- $K_I$  → Constante integrativa del controlador PID
- $K_D$  → Constante derivativa del controlador PID

En la figura 2.1 se muestra el diagrama de bloques de un lazo cerrado de control marcando de amarillo la información con la que no se cuenta. Se debe tomar en cuenta que el bloque “PROCESO” está conformado por el transductor, actuador, válvula de control y el sensor,

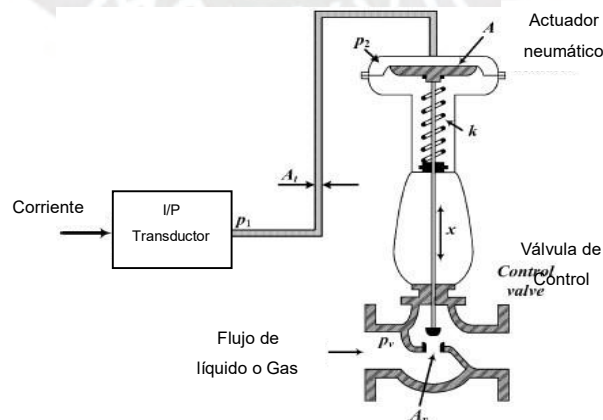


**Figura 2.1:** Diagrama de bloques de un lazo cerrado de control

El proceso de control de flujo que trabaja con una válvula de diafragma es común en las plantas concentradoras por lo que se va a encontrar el modelo teórico de este proceso para evaluar los métodos de identificación.

### 2.2.1 Modelo teórico de un proceso control de flujo con válvula de diafragma

Se debe encontrar la función de transferencia del bloque “PROCESO” el cual ya se indicó que está conformado por el transductor, actuador, válvula de control y el sensor. En la figura 2.2 se puede observar cual es la mecánica del actuador de una válvula de diafragma.



**Figura 2.2:** Mecánica del actuador de una válvula de diafragma (Boiko, 2013)

En la figura 2.1 se muestra que la variable MV es la salida de 0 a 100% del PID y esto es lo que ingresa al transductor pero ya convertido en corriente de 4 a 20 mA a través de una tarjeta de salida del PLC o DCS, por tal motivo para el transductor represento con  $MV(t)$  la entrada de 0-100% y con  $P_1(t)$  la salida de presión, entonces la función de transferencia que representa al transductor se puede observar en la ecuación 2.1.

$$T(s) = \frac{P_1(s)}{MV(s)} \quad (2.1)$$

En la ecuación 2.2 se muestra el modelo linealizado de la dinámica del actuador el cual es ampliamente utilizado (Boiko, 2013).

$$W(s) = \frac{X(s)}{P_1(s)} = \frac{A}{ms^2 + bs + k} \quad (2.2)$$

Donde:

- $A$  → Es el área de superficie efectiva del diafragma
- $m$  → Masa del vástago y partes móviles asociadas
- $b$  → Fricción viscosa
- $k$  → Relación carga de deformación del resorte de una válvula de diafragma

si se considera que la válvula tiene una característica lineal donde  $q(t)$  es el caudal de salida y  $x(t)$  es la posición del vástago, entonces la función de transferencia que representa la válvula se puede observar en la ecuación 2.3.

$$V(s) = \frac{Q(s)}{X(s)} = C \quad (2.3)$$

Donde:

- $C$  → Constante lineal de la válvula

Para poder medir el caudal  $q(t)$  es necesario agregar un sensor que permita al controlador obtener la medida del caudal el cual será representado por  $PV(t)$ , considerando un sensor de flujo de inserción el cual cumple con la ley de Faraday, la función de transferencia que representa la válvula se puede observar en la ecuación 2.4.

$$M(s) = \frac{PV(s)}{Q(s)} \quad (2.4)$$

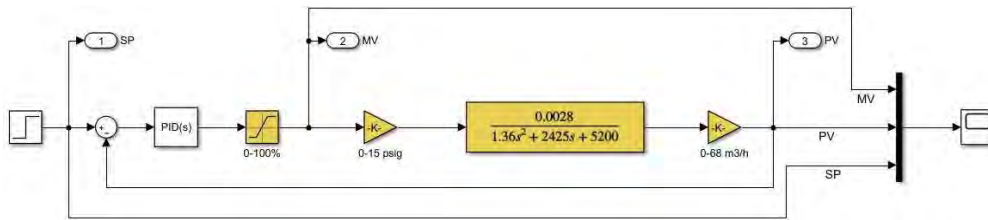
Finalmente, la función de transferencia del bloque “PROCESO” que se muestra en la figura 2.1 estaría dado por el producto de las funciones de transferencia del transductor, actuador, válvula de control y el sensor, es decir el producto de las ecuaciones 2.1, 2.2, 2.3 y 2.4.

$$G(s) = T(s)W(s)V(s)M(s) = \frac{PV(s)}{MV(s)} = C \frac{P_1(s)}{MV(s)} \frac{PV(s)}{Q(s)} \frac{A}{ms^2 + bs + k} \quad (2.5)$$

Una vez que ya se tiene la función de transferencia es necesario tener los valores de diseño de un proceso de control de flujo para poder iniciar la identificación, para esto ha sido tomados los siguientes valores del libro “Non-parametric Tuning of PID Controllers” (Boiko, 2013).

- $P_1 \rightarrow 3$  a 15 psig
- $A \rightarrow 0.0028 \text{ m}^2$
- $m \rightarrow 1.36 \text{ kg}$
- $b \rightarrow 2425 \text{ kg/s}$
- $k \rightarrow 5200 \text{ N/m}$
- $PV \rightarrow$  Máximo flujo  $68 \text{ m}^3/\text{h}$
- La característica de la válvula es lineal

En la figura 2.3 se muestra el diagrama de bloques que se ha construido con la función de transferencia y los valores de diseño.



**Figura 2.3:** Diagrama de bloques en lazo cerrado del modelo teórico

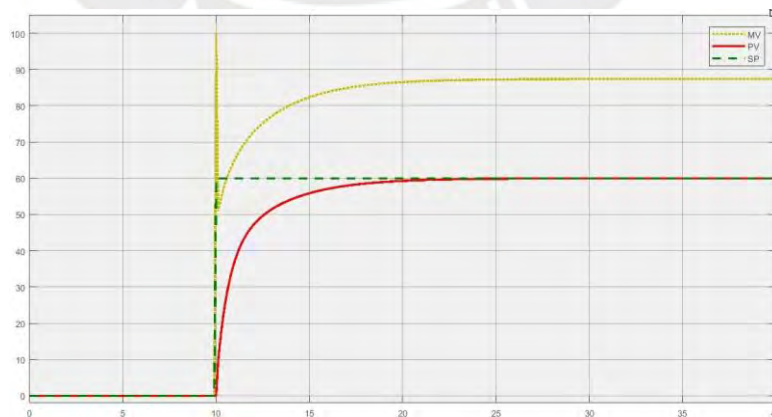
## 2.2.2 Identificación en lazo cerrado sobre un proceso teórico

Para poder realizar la identificación en lazo cerrado del diagrama de bloques mostrado en la figura 2.3, es necesario primero tener un muestreo de los valores de SP, MV y PV, así como también los parámetros del controlador PID.

Se va a considerar un controlador PID paralelo con parámetros de  $K_p$  de 2,  $K_I$  y  $K_D$  de 1, la función de transferencia del controlador se muestra en la ecuación 2.6.

$$PID(s) = 2 + \frac{1}{s} + s \quad (2.6)$$

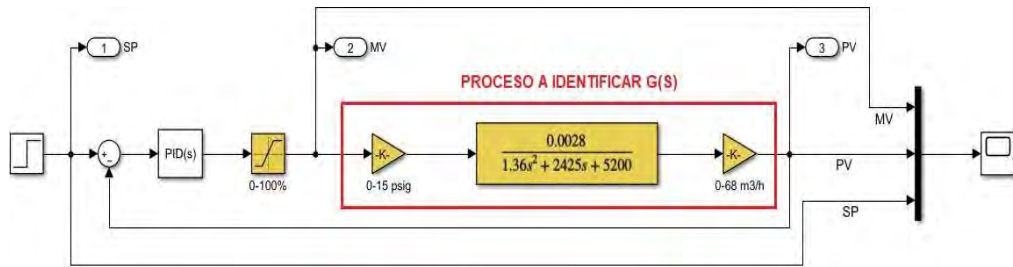
Se va a considerar un cambio en el SP de 0 a 60 m<sup>3</sup>/h en el segundo 10 y un rango de muestreo de 40 segundos, en la figura 2.4 se muestra la respuesta del SP, MV y PV que han sido muestreados con tiempo de 0.1 segundos.



**Figura 2.4:** Muestreo de SP, MV y PV con tiempo de 0.1 segundos

### 2.2.2.1 Identificación con modelo ARX

Lo que se va a identificar es el proceso que tiene como entrada a la MV y como salida a la PV, en la figura 2.5 se encierra en un rectángulo rojo el proceso a identificar.



**Figura 2.5:** Proceso G(S) que se debe identificar

El modelo teórico de la función de transferencia  $G(s)$  se puede observar en la ecuación 2.7.

$$G(s) = \frac{3570}{1.36s^2 + 2425s + 5200} \quad (2.7)$$

Esta función de transferencia en “s” podemos convertirla a una función de transferencia en “z” con un tiempo de muestreo de 0.1 segundos y se puede observar en la ecuación 2.8.

$$G(z) = \frac{0.132z + 0.0006685}{z^2 - 0.8068z} \quad (2.8)$$

La función  $G(z)$  es la que va a identificar usando un modelo ARX, en la ecuación 2.9 se muestra la estructura del modelo.

$$y[k] = \frac{B(q^{-1})}{A(q^{-1})}u[k] + \frac{1}{A(q^{-1})}e[k] \quad (2.9)$$

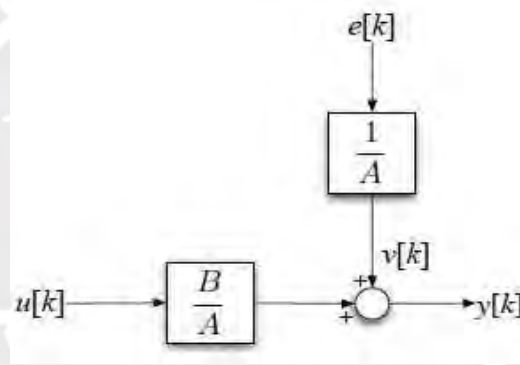
Donde los polinomios A y B tienen la siguiente forma:

- $A(q^{-1}) = 1 + a_1q^{-1} + \dots + a_{n_a}q^{-n_a}$
- $B(q^{-1}) = b_{n_k}q^{-n_k} + b_{n_k+1}q^{-n_k-1} + \dots + b_{n_k+n_b-1}q^{-(n_b+n_k-1)}$

Las señales son las siguientes:

- $y[k]$  → Salida del sistema
- $x[k]$  → Entrada del sistema
- $e[k]$  → Ruido de entrada al sistema

Y su estructura se presenta gráficamente en la figura 2.6.



**Figura 2.6:** Representación esquemática de la estructura ARX (Tangirala, 2015)

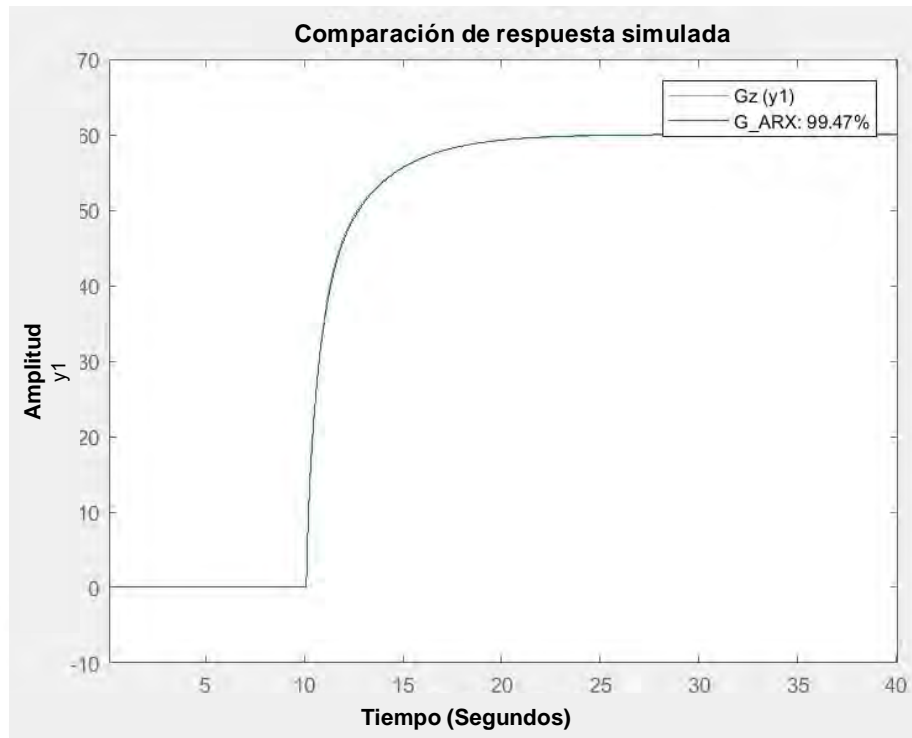
Como se cuenta con el modelo teórico de  $G(z)$  mostrado en la ecuación 2.8, los polinomios A y B del modelo ARX deben coincidir con este modelo, por tal motivo se tiene que el orden de cada polinomio debe ser el siguiente.

$$n_a = 2, n_b = 1, n_k = 1 \quad (2.10)$$

Con el orden de cada polinomio y el muestreo de las señales MV y PV (ver figura 2.4) se obtiene el siguiente el modelo ARX.

$$y[k] = \frac{0.09388q^{-1}}{1 - 1.096q^{-1} + 0.2322q^{-2}}u[k] + \frac{1}{1 - 1.096q^{-1} + 0.2322q^{-2}}e[k] \quad (2.11)$$

Comparando el modelo teórico encontrado en la ecuación 2.8 con el modelo ARX encontrado en la ecuación 2.11 se obtiene una estimación de los datos de un 99.47% como se aprecia en la figura 2.7



**Figura 2.7:** Comparación de la respuesta teórica con el modelo ARX

Con un modelo teórico se obtiene una excelente estimación de los datos, pero se debe tomar en cuenta que en este caso se cuenta con la información del sistema por lo que fue sencillo encontrar el orden de los polinomios, en la práctica no se cuenta con esta información por lo que se debe realizar varios ensayos hasta encontrar el modelo adecuado.

### 2.2.2.2 Identificación con modelo ARMAX

La función  $G(z)$  que se va a identificar usando un modelo ARMAX es la misma que para la identificación por ARX, en la ecuación 2.12 se muestra la estructura del modelo.

$$y[k] = \frac{B(q^{-1})}{A(q^{-1})}u[k] + \frac{C(q^{-1})}{A(q^{-1})}e[k] \quad (2.12)$$

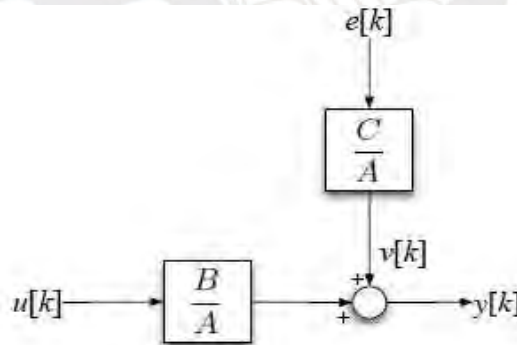
Donde los polinomios A, B y C tienen la siguiente forma:

- $A(q^{-1}) = 1 + a_1q^{-1} + \dots + a_{n_a}q^{-n_a}$
- $B(q^{-1}) = b_{n_k}q^{-n_k} + b_{n_k+1}q^{-n_k-1} + \dots + b_{n_k+n_b-1}q^{-(n_b+n_k-1)}$
- $C(q^{-1}) = 1 + c_1q^{-1} + \dots + c_{n_c}q^{-n_c}$

Las señales son las siguientes:

- $y[k] \rightarrow$  Salida del sistema
- $x[k] \rightarrow$  Entrada del sistema
- $e[k] \rightarrow$  Ruido de entrada al sistema

Y su estructura se presenta gráficamente en la figura 2.8.



**Figura 2.8:** Representación esquemática de la estructura ARMAX (Tangirala, 2015)

Como se cuenta con el modelo teórico de  $G(z)$  mostrado en la ecuación 2.8, los polinomios A y B del modelo ARMAX deben coincidir con este modelo, por tal motivo se tiene que el orden de cada polinomio debe ser el siguiente.

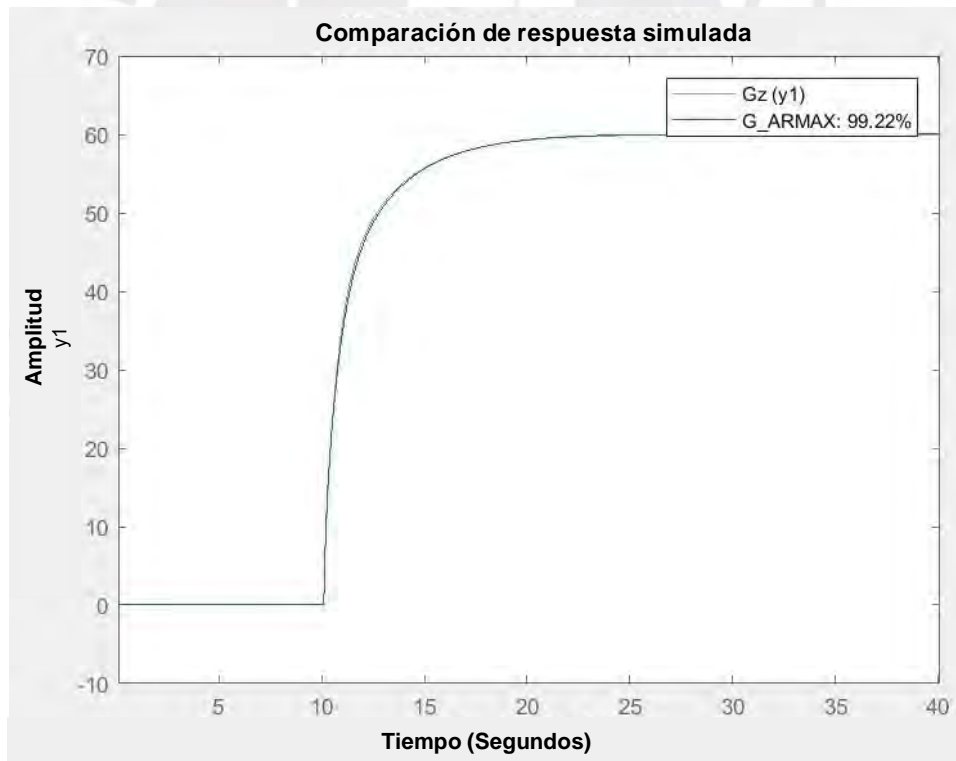
$$n_a = 2, n_b = 1, n_k = 1 \quad (2.13)$$

El modelo ARMAX a diferencia del ARX agrega un polinomio “C” que afecta directamente al ruido  $e[k]$ , como el modelo teórico no contempla el ruido se va a probar con un polinomio “C” de orden 1, es decir  $n_c = 1$ , lo cual es suficiente para el modelo teórico.

Con el orden de cada polinomio y el muestreo de las señales MV y PV (ver figura 2.4) se obtiene el siguiente modelo ARMAX.

$$y[k] = \frac{0.09684q^{-1}}{1 - 1.046q^{-1} + 0.1871q^{-2}}u[k] + \frac{1 + 0.9993q^{-1}}{1 - 1.046q^{-1} + 0.1871q^{-2}}e[k] \quad (2.14)$$

Comparando el modelo teórico encontrado en la ecuación 2.8 con el modelo ARMAX encontrado en la ecuación 2.14 se obtiene una estimación de los datos de un 99.22% como se aprecia en la figura 2.9.

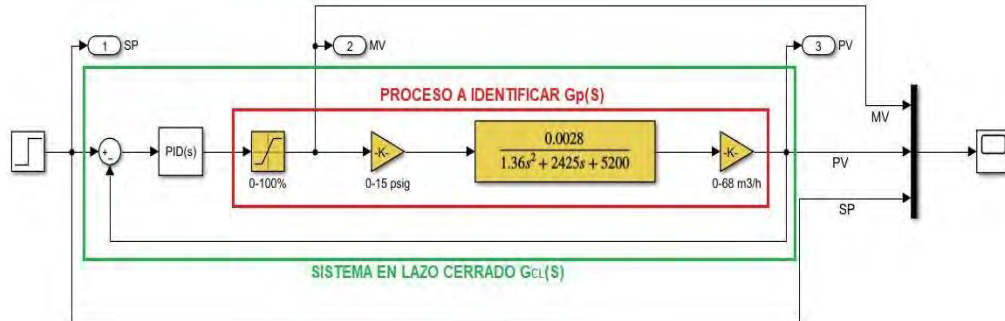


**Figura 2.9:** Comparación de la respuesta teórica con el modelo ARMAX

Al igual que con ARX se obtiene una excelente estimación de los datos, aunque con un valor ligeramente menor, 99.22% de ARMAX respecto al 99.47% de ARX, con un modelo teórico se puede usar tanto ARX como ARMAX. También se debe tomar en cuenta que al contar con la información del sistema es sencillo encontrar el orden de los polinomios y al no considerar el ruido se puede probar el valor de 1 para el orden del polinomio “C”, además se debe recalcar que en la práctica no se cuenta con esta información por lo que se debe realizar varios ensayos hasta encontrar el modelo adecuado.

### 2.2.2.3 Identificación de proceso con controlador conocido

Tomando en consideración que en la práctica siempre se conoce los parámetros del controlador PID, en la figura 2.10 se encierra en un rectángulo rojo el proceso  $G_p(s)$  a identificar y, en un rectángulo verde el sistema en lazo cerrado  $G_{CL}(s)$  que tiene como entrada al SP y salida a la PV.



**Figura 2.10:** Proceso  $G_p(s)$  que se debe identificar

Tomando en consideración el diagrama de bloques de la figura 2.10, se tiene que la función de transferencia del sistema en lazo cerrado  $G_{CL}(s)$  viene dada por la siguiente ecuación.

$$G_{CL}(s) = \frac{PV(s)}{SP(s)} = \frac{PID(s)G_p(s)}{1 + PID(s)G_p(s)} \quad (2.15)$$

De la ecuación 2.15 se puede despejar la función de transferencia del proceso  $G_p(s)$ , se obtiene lo siguiente.

$$G_p(s) = \frac{G_{CL}(s)}{PID(s)(1 - G_{CL}(s))} \quad (2.16)$$

Como en la ecuación 2.6 se tiene la función de transferencia de  $PID(s)$ , ahora es necesario identificar la función de transferencia del sistema en lazo cerrado  $G_{CL}(s)$  que tiene como entrada al SP y como salida a la PV los cuales han sido muestreados con un tiempo de 0.1 segundos (ver figura 2.4), se puede utilizar ARX o ARMAX para poder identificarlo y luego transformar la función de transferencia de “z” a “s” pero estos modelos agregan la variable del ruido y complicaría el cálculo de la función de transferencia en “s”, por lo que para identificar  $G_{CL}(s)$  se va a usar la estimación de la función de transferencia de tiempo continuo usando los datos en el dominio del tiempo discreto. Esto se puede realizar rápidamente usando el comando “tfest” de Matlab, pero es necesario colocar la cantidad de ceros y polos que tendría la función de transferencia  $G_{CL}(s)$ , para estimarlo se va a calcular la función de transferencia teórica.

Tomando en cuenta que se tiene la función del controlador PID en la ecuación 2.6, se tiene la función de transferencia del proceso en la ecuación 2.7 y si consideramos que el limitador de 0 a 100% del PID que se puede ver en la figura 2.10 tiene un valor constante K, estos pueden ser reemplazados en la ecuación 2.17 y se obtiene la función de transferencia teórica del sistema en lazo cerrado  $G_{CL}(s)$ .

$$G_{CL}(s) = \frac{3570K(s^2 + 2s + 1)}{1.36s^3 + 5995s^2 + 12340s + 3570} \quad (2.17)$$

Se puede observar que  $G_{CL}(s)$  tiene 2 ceros y 3 polos. En la figura 2.11 se muestra cómo se ejecuta el comando “tfest” de Matlab considerando como entrada al SP, salida a la PV, 2 ceros y 3 polos.

```

load('Datos_Modelo_Teorico_Flujo.mat')
T=0.1; % tiempo de muestreo
SP = Datos_Modelo_Teorico_Flujo(:,1);
PV = Datos_Modelo_Teorico_Flujo(:,3);
Ceros=2;
Polos=3;

GCL_Muestreado = iddata(PV,SP,T);
GCL_Identicado = tfest(GCL_Muestreado,Polos,Ceros);

compare(GCL_Muestreado,GCL_Identicado)

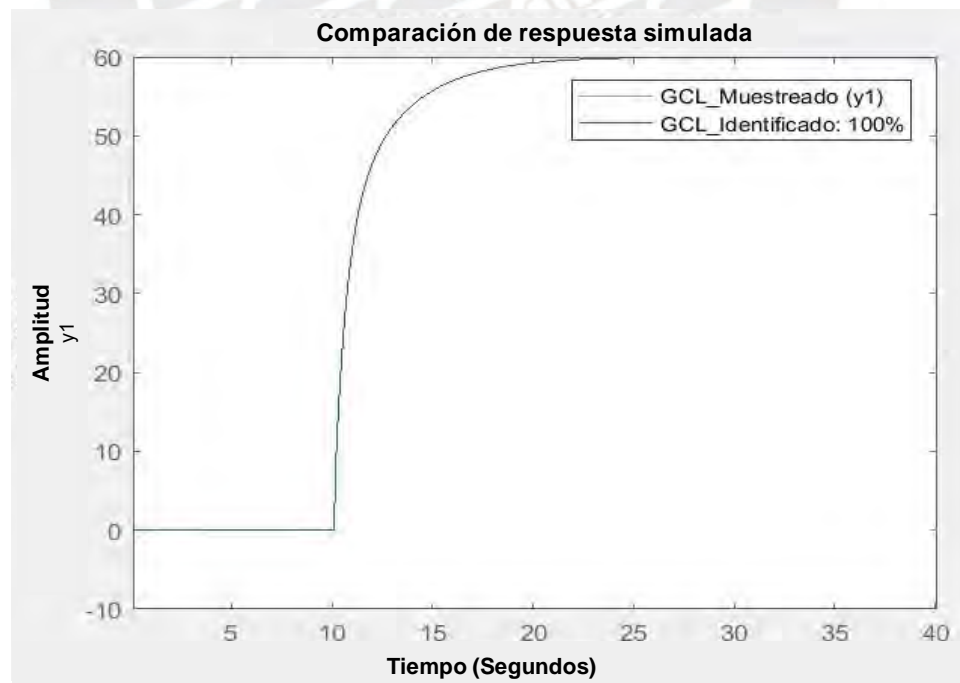
```

**Figura 2.11:** Ejecución del comando “tfest” de Matlab

La función de transferencia que se obtiene se puede observar en la ecuación 2.18.

$$G_{CL}(s) = \frac{5.063s^2 + 85.71s + 42.43}{s^3 + 74.13s^2 + 147.5s + 42.43} \quad (2.18)$$

Comparando los valores muestreados con el modelo obtenido en la ecuación 2.18 se obtiene una estimación de los datos del 100% para  $G_{CL}(s)$  como se aprecia en la figura 2.12.



**Figura 2.12:** Comparación de la respuesta  $G_{CL}(s)$  muestreado con su identificación

Como ya se tiene las funciones de transferencia PID(s) en la ecuación 2.6 y  $G_{CL}(s)$  en la ecuación 2.18, estas se reemplazan en la ecuación 2.16 y, se puede obtener el margen de ganancia y margen de fase para poder diseñar el controlador (Johnson y Moradi, 2005), pero en la presente tesis realizo la sintonización usando el método de algoritmos genéticos por lo que en realidad lo que se necesita es la función de transferencia de  $G_p(s)$ , entonces solo se requiere la función de transferencia que se encuentra al reemplazar las ecuaciones 2.6 y 2.18 en la ecuación 2.16, esta función de transferencia se puede ver en la ecuación 2.19.

$$G_p(s) = \frac{a_6s^6 + a_5s^5 + a_4s^4 + a_3s^3 + a_2s^2 + a_1s}{s^8 + b_7s^7 + b_6s^6 + b_5s^5 + b_4s^4 + b_3s^3 + b_2s^2 + b_1s + b_0} \quad (2.19)$$

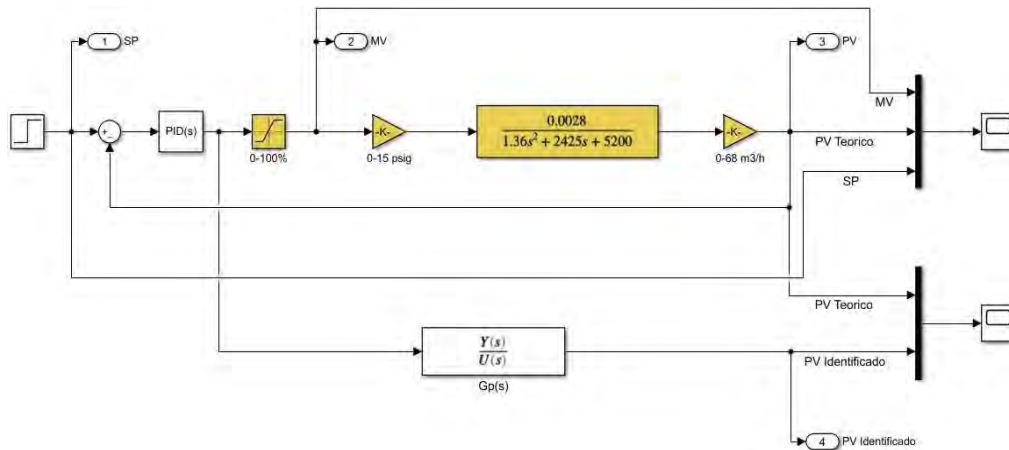
Donde los valores de los coeficientes del numerador y denominador se pueden ver en la figura 2.13.

```
Gp =
From input to output "y1":
      5.063 s^6 + 461.1 s^5 + 7143 s^4 + 1.6e04 s^3 + 9894 s^2 + 1800 s
-----
s^8 + 145.2 s^7 + 5617 s^6 + 2.561e04 s^5 + 4.699e04 s^4 + 4.151e04 s^3 + 1.728e04 s^2 + 2621 s + 8.429e-05
Continuous-time transfer function.
```

**Figura 2.13:** Función de transferencia  $G_p(s)$  calculado en Matlab

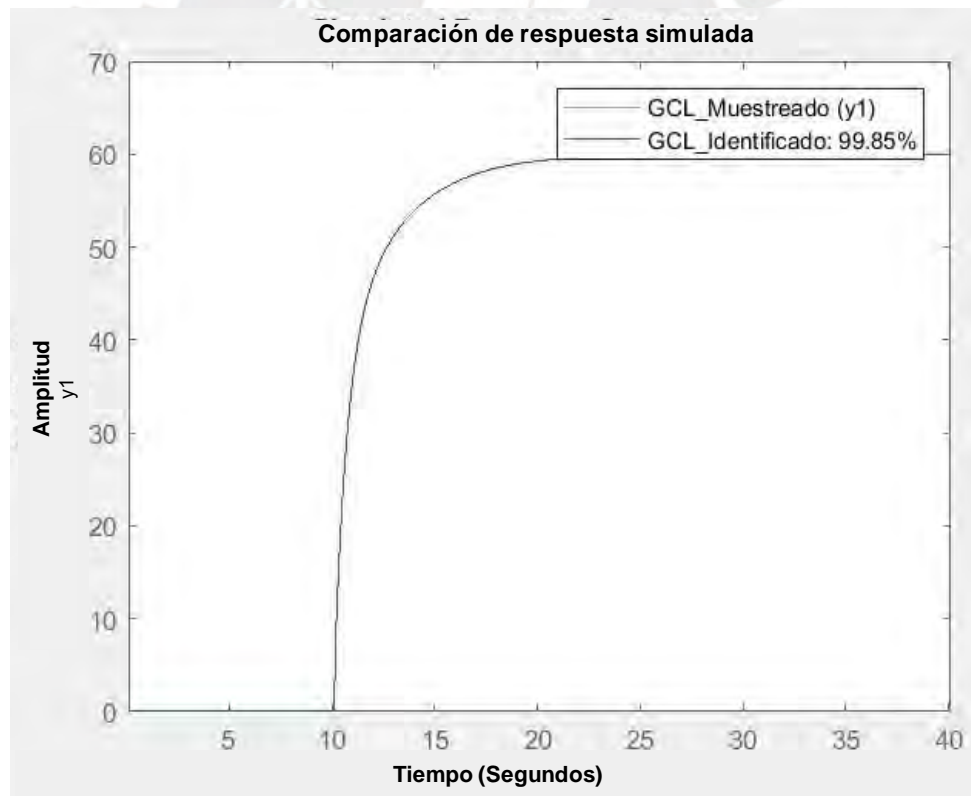
Como se puede observar la identificación de la función de transferencia de  $G_p(s)$  se obtuvo solo operando matemáticamente la ecuación 2.16, por este motivo la cantidad 6 ceros y 8 polos obtenidos es correcto.

Ahora que ya se cuenta con la función de transferencia identificada de  $G_p(s)$ , en la figura 2.14 se muestra un modelo en Simulink que permitirá poder comparar la respuesta del sistema teórico en lazo cerrado con el modelo obtenido.



**Figura 2.14:** Modelo en Simulink para validar la identificación de  $G_p(s)$

Se obtiene una estimación de los datos del 99.85% al comparar el PV teórico y PV identificado como se aprecia en la figura 2.15.

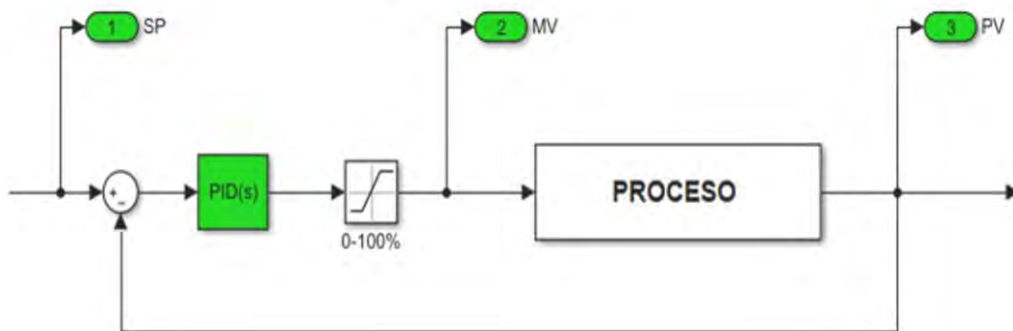


**Figura 2.15:** Comparación entre la salida  $G_{CL}(s)$  muestreada y la salida de la identificación de  $G_p(s)$

Este método de identificación tiene una excelente estimación de datos del 99.85% ligeramente superior a la identificación usando ARX (99.47%) y usando ARMAX (99.22%), se debe tomar en cuenta que también se necesitó de la información del sistema para poder saber cuántos ceros y polos se necesitaban para la identificación en lazo cerrado del sistema que tiene como entrada al SP y salida a la PV (ver figura 2.10), como se sabe en la práctica no se cuenta con esta información pero al tener que la entrada SP es de tipo escalón es fácil poder identificar y asumir que cantidad de ceros y de polos tiene el sistema solo con ver la respuesta, además en la práctica no es necesario trabajar con un sistema que sea mayor a uno de tercer orden.

### 2.3 Método de identificación en lazo cerrado sobre un proceso real

En la subsección 2.2.2 se analizó 3 métodos de identificación sobre un modelo teórico linealizado de segundo orden que representa un proceso de control de flujo con válvula de diafragma, este tipo de proceso es común en una planta concentradora de cobre, por este motivo para poder probar adecuadamente los 3 métodos de identificación se usa los datos reales que se han obtenido del proceso de inyección de agua cruda al cajón de alimentación de las bombas de ciclones el cual también trabaja con una válvula de diafragma. En la figura 2.16 se muestra el diagrama de bloques del lazo cerrado de control del proceso mencionado marcando de verde la información con la que se cuenta.



**Figura 2.16:** Diagrama de bloques del proceso de inyección de agua cruda al cajón de alimentación de las bombas de ciclones

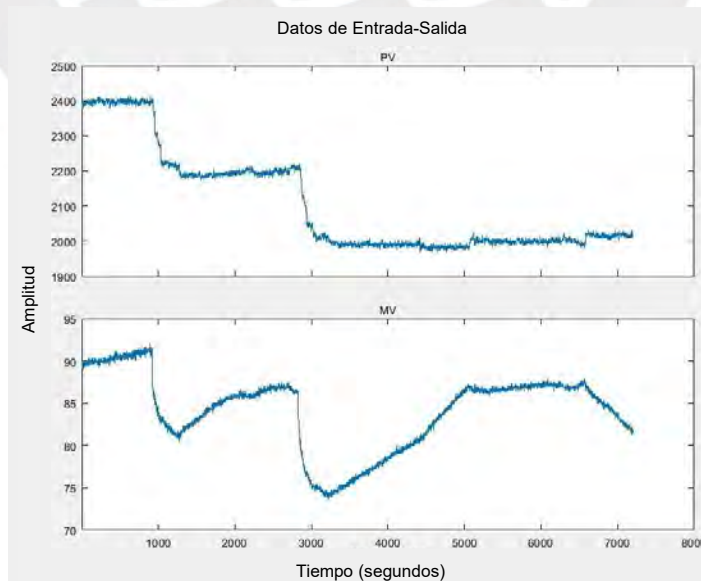
En el lazo de control mostrado en la figura 2.16 se conoce la siguiente información que ha sido extraída del sistema de control:

- $K_p$  → Constante proporcional del controlador PID con valor de 0.5
- $T_i$  → Tiempo integrativo del controlador PID con valor de 30, donde  $K_I = K_p/T_i$
- $T_d$  → Tiempo derivativo del controlador PID con valor de 1.1, donde  $K_D = K_p T_d$
- PID → Tipo paralelo
- SP → Set point del lazo de control muestreado a 1 segundo
- MV → Variable manipulada del lazo de control muestreado a 1 segundo
- PV → Variable de proceso del lazo de control muestreado a 1 segundo

Conociendo el tipo de controlador PID y sus parámetros  $K_p$ ,  $T_i$  y  $T_d$  se puede obtener su función de transferencia el cual se muestra en la ecuación 2.20.

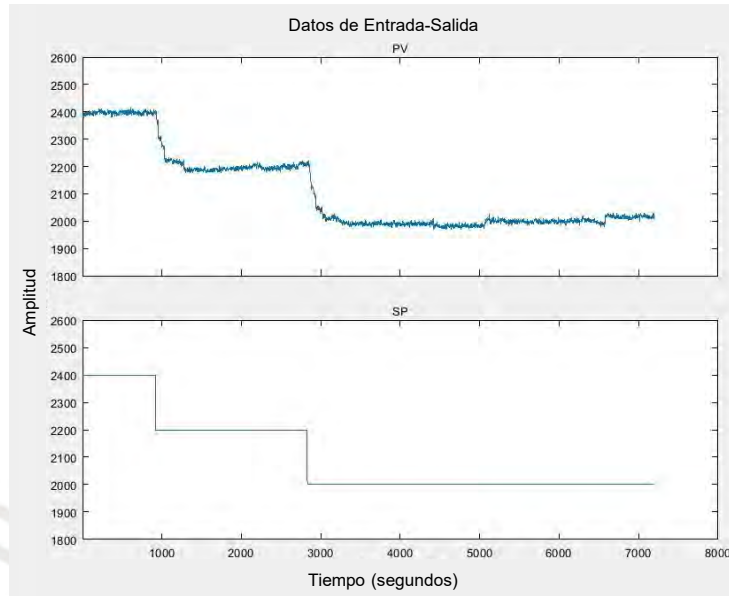
$$\text{PID}(s) = 0.5 + 0.01667 \frac{1}{s} + 0.55s \quad (2.20)$$

De la base de datos se obtiene la variación de la PV con respecto a la MV en un rango de 7200 segundos, este se muestra en la figura 2.17.



**Figura 2.17:** Variación de la PV con respecto a la MV

De la base de datos se obtiene la variación de la PV con respecto al SP en un rango de 7200 segundos, este se muestra en la figura 2.18.



**Figura 2.18:** Variación de la PV con respecto al SP

Ahora que ya se tiene toda la información que puede ser extraída del sistema de control se analizan los 3 métodos de identificación que se probaron en la subsección 2.2.2.

### 2.3.1 Identificación con modelo ARX

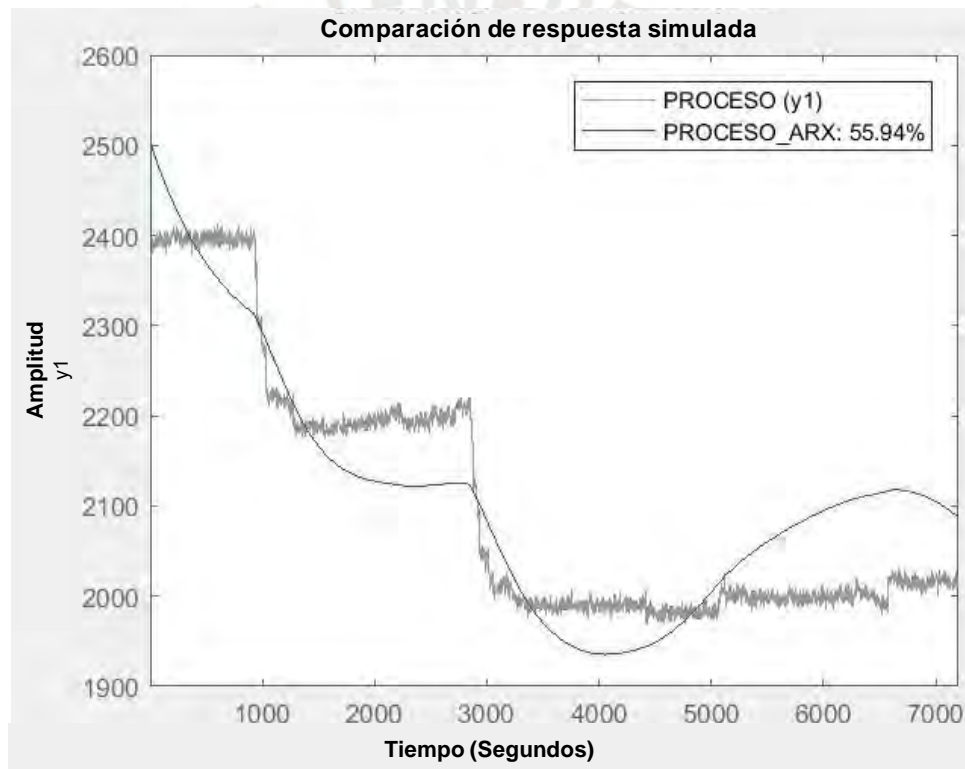
Lo que se va a identificar es el bloque “PROCESO” que se muestra en la figura 2.16 donde el muestreo de la MV y la PV se puede ver en la figura 2.17. Al observar con detalle la variación de la PV con respecto a la MV se puede apreciar que la PV no responde de acuerdo con la variación de la MV, esto se debe a que la válvula de diafragma está trabajando en un rango no lineal pero además que la PV debe depender también de otras variables las cuales no conocemos, por este motivo se puede esperar que el modelo obtenido por ARX tenga una baja estimación de los datos.

Como se ha escogido un proceso de flujo similar al analizado de manera teórica en la subsección 2.2.2 se puede tomar para el primer ensayo del modelamiento con ARX el mismo orden de los polinomios que se encontraron en la ecuación 2.10.

Con el orden de cada polinomio y el muestreo de las señales MV y PV se obtiene el siguiente modelo ARX.

$$y[k] = \frac{0.02891q^{-1}}{1 - 1.04q^{-1} + 0.04113q^{-2}}u[k] + \frac{1}{1 - 1.04q^{-1} + 0.04113q^{-2}}e[k] \quad (2.21)$$

Comparando el proceso muestreado (ver figura 2.17) con el modelo ARX encontrado en la ecuación 2.21 se obtiene una estimación de los datos de un 55.94% como se aprecia en la figura 2.19

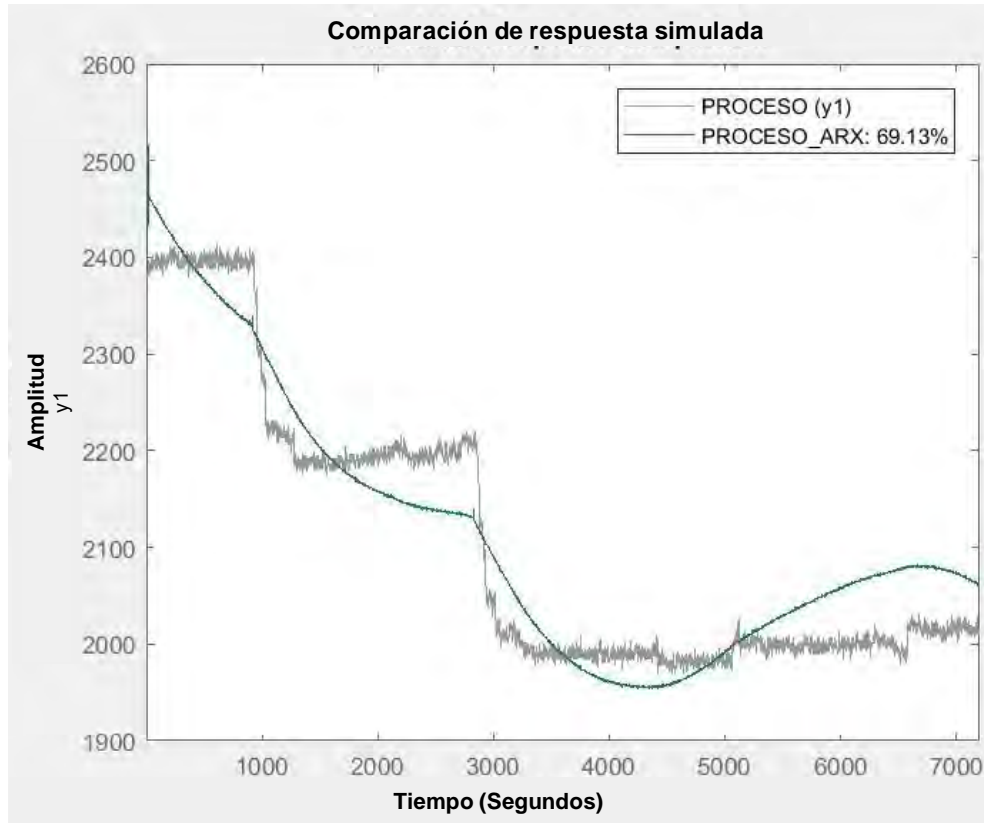


**Figura 2.19:** Comparación del proceso muestreado con el modelo ARX del ensayo 1

El valor de la estimación es bajo por lo que no puede utilizarse el modelo obtenido, por este motivo es necesario buscar otros valores para el orden de los polinomios. Tomando un rango de 0 a 15 para el orden  $n_a$  y el rango 0 a 10 para el orden  $n_b$  y  $n_k$ , se realizaron varias pruebas y se encontró que la máxima estimación de datos se encontraba que para los siguientes órdenes.

$$n_a = 15, n_b = 10, n_k = 1 \quad (2.22)$$

En la figura 2.20 se muestra que la estimación de datos encontrada fue del 69.13%.



**Figura 2.20:** Máxima estimación de datos encontrada con el modelo ARX

Como se esperaba el modelo ARX no permite obtener una apropiada estimación de datos, por tal motivo este método de identificación no puede utilizarse.

### 2.3.2 Identificación con modelo ARMAX

Al igual que el para el modelo ARX lo que se va a identificar es el bloque "PROCESO" y, debido a que la válvula de diafragma está trabajando en un rango no lineal y la PV depende también de otras variables, se puede esperar que el modelo obtenido por ARMAX tenga una baja estimación de los datos.

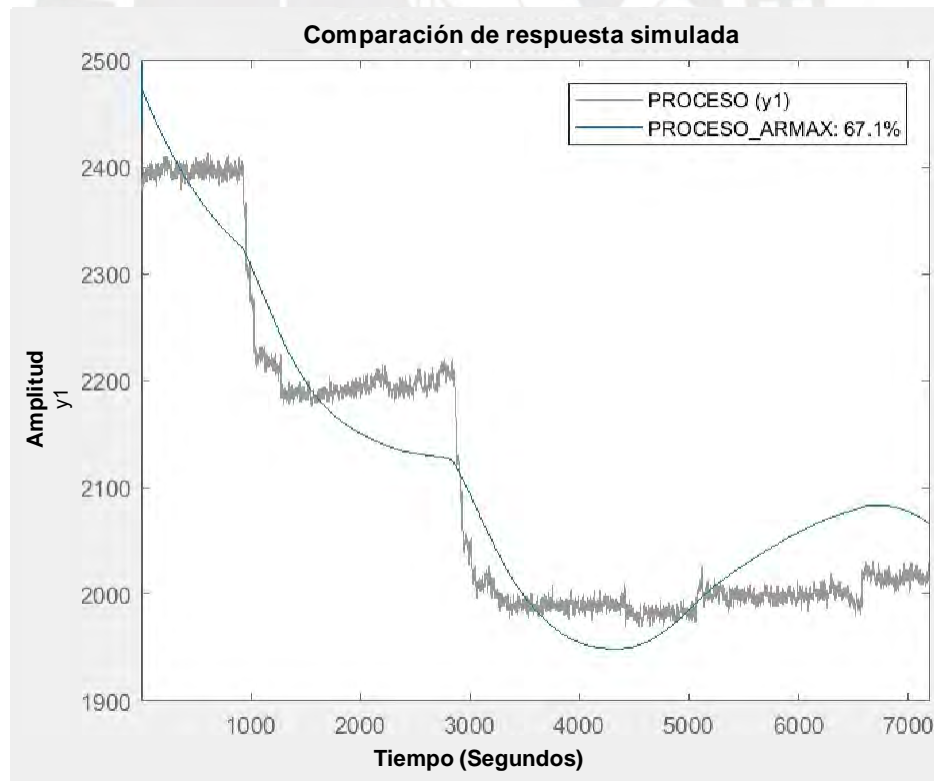
Para el primer ensayo del modelamiento con ARMAX se usará el mismo orden de los polinomios que se encontraron en la subsección 2.2.2.2, estos se muestran en la ecuación 2.23.

$$n_a = 2, n_b = 1, n_c = 1, n_k = 1 \quad (2.23)$$

Con el orden de cada polinomio y el muestreo de las señales MV y PV se obtiene el siguiente el modelo ARMAX.

$$y[k] = \frac{0.02891q^{-1}}{1 - 1.04q^{-1} + 0.041113q^{-2}}u[k] + \frac{1}{1 - 1.04q^{-1} + 0.041113q^{-2}}e[k] \quad (2.24)$$

Comparando el proceso muestreado (ver figura 2.17) con el modelo ARMAX encontrado en la ecuación 2.24 se obtiene una estimación de los datos de un 67.1% como se aprecia en la figura 2.21

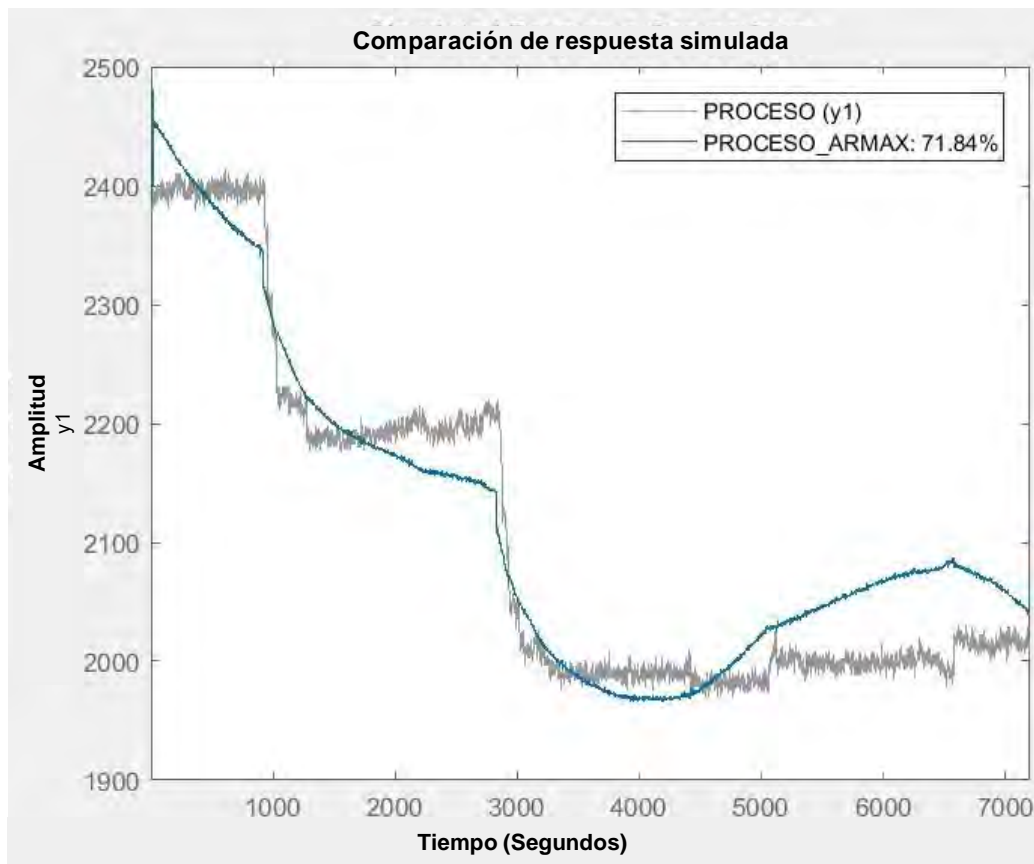


**Figura 2.21:** Comparación del proceso muestreado con el modelo ARMAX ensayo 1

El valor de la estimación es bajo por lo que no puede utilizarse el modelo obtenido, por este motivo se realiza más ensayos y se encuentra que la máxima estimación de datos se tiene para los siguientes órdenes.

$$n_a = 15, n_b = 4, n_c = 2, n_k = 2 \quad (2.25)$$

En la figura 2.20 se muestra que la estimación de datos encontrada fue del 71.84%.

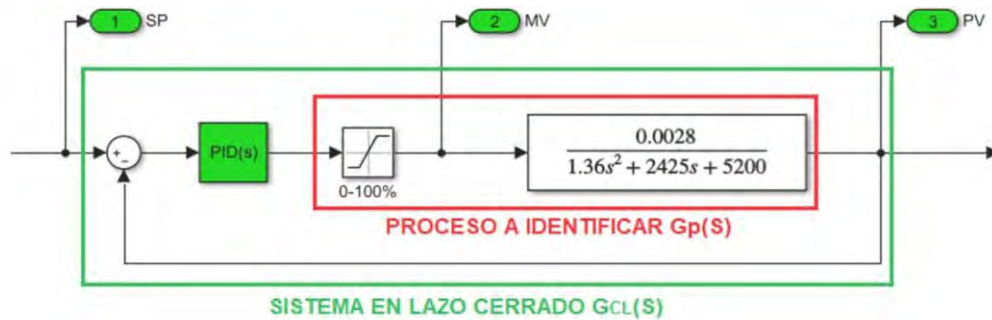


**Figura 2.22:** Máxima estimación de datos encontrada con el modelo ARMAX

Como se esperaba el modelo ARMAX no permite obtener una apropiada estimación de datos, aunque este haya llegado a superar el 70%, también se puede visualizar fácilmente en la figura 2.22 que la respuesta obtenida del modelo no es la adecuada, por tal motivo este método de identificación tampoco puede utilizarse.

### 2.3.3 Identificación de proceso con controlador conocido

Como se conoce los parámetros del controlador PID (ver ecuación 2.20) y se tiene el muestreo del SP y la PV (ver figura 2.18), en la figura 2.23 se encierra en un rectángulo rojo el proceso  $G_p(s)$  a identificar y, en un rectángulo verde el sistema en lazo cerrado  $G_{CL}(s)$  que tiene como entrada al SP y salida a la PV.



**Figura 2.23:** Proceso  $G_p(s)$  a identificar

Como se hizo en la subsección 2.2.2.3 se va a identificar  $G_{CL}(s)$  usando la estimación de la función de transferencia de tiempo continuo usando los datos en el dominio del tiempo discreto y aunque en esta subsección se usó 2 ceros y 3 polos, en este caso se usa 1 cero y 3 polos ya que con esto se encontró mejores resultados. En la figura 2.24 se muestra el código que se ejecuta en Matlab para encontrar  $G_{CL}(s)$

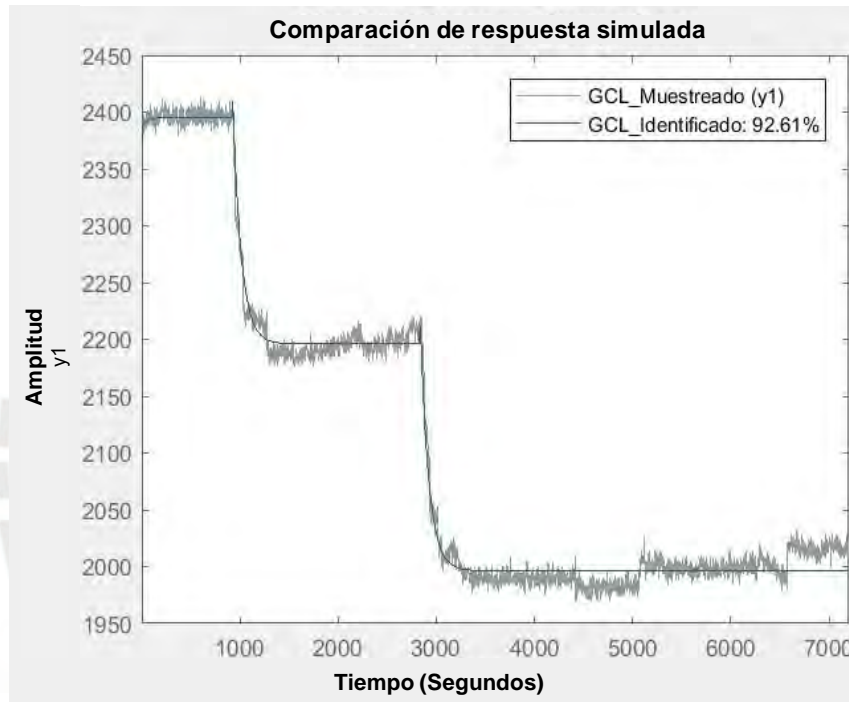
```
% Importar datos %
Datos=xlsread('210FIC0279.xls');
% Parametros
T=1; % tiempo de muestreo
SP=Datos(:,2);
PV=Datos(:,3);
Ceros=1;
Polos=3;
% Identificacion de sistema en lazo cerrado GCL(S)
GCL_Muestreado = iddata(PV,SP,T);
GCL_Identicado = tfest(GCL_Muestreado,Polos,Ceros);
% Comparacion de GCL muestreado y GCL identificado
figure (1)
compare(GCL_Muestreado,GCL_Identicado)
```

**Figura 2.24:** Código en Matlab para obtener  $G_{CL}(s)$

La función de transferencia que se obtiene se puede observar en la ecuación 2.26.

$$G_{CL}(s) = \frac{-0.3136s + 0.02516}{s^3 + 9.76s^2 + 2.272s + 0.0252} \quad (2.26)$$

Se obtiene una estimación de los datos del 92.61% para  $G_{CL}(s)$  como se aprecia en la figura 2.25.



**Figura 2.25:** Comparación de la respuesta  $G_{CL}(s)$  muestreado e identificado

Como ya se tiene las funciones de transferencia PID(s) en la ecuación 2.20 y  $G_{CL}(s)$  en la ecuación 2.26, estas se reemplazan en la ecuación 2.16 y se encuentra la función de transferencia de  $G_P(s)$ .

$$G_P(s) = \frac{a_5s^5 + a_4s^4 + a_3s^3 + a_2s^2 + a_1s}{s^8 + b_7s^7 + b_6s^6 + b_5s^5 + b_4s^4 + b_3s^3 + b_2s^2 + b_1s + b_0} \quad (2.27)$$

Donde los valores de los coeficientes del numerador y denominador se pueden ver en la figura 2.26.

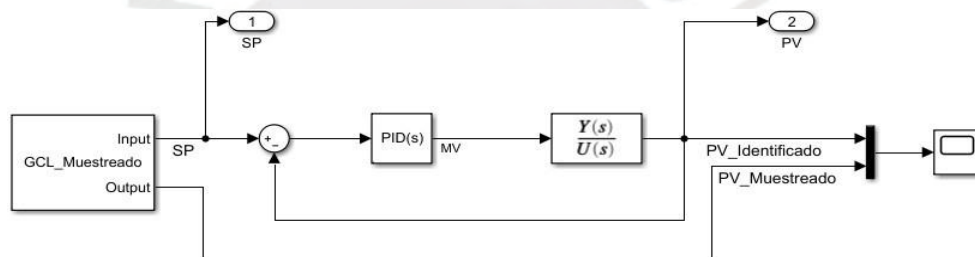
```

Gp =
From input to output "y1":
      -0.3136 s^5 - 3.036 s^4 - 0.4671 s^3 + 0.04926 s^2 + 0.0006341 s
-----
0.55 s^8 + 11.24 s^7 + 64.84 s^6 + 76.48 s^5 + 28.76 s^4 + 3.887 s^3 + 0.1347 s^2 + 0.001088 s + 1.907e-08
Continuous-time transfer function.

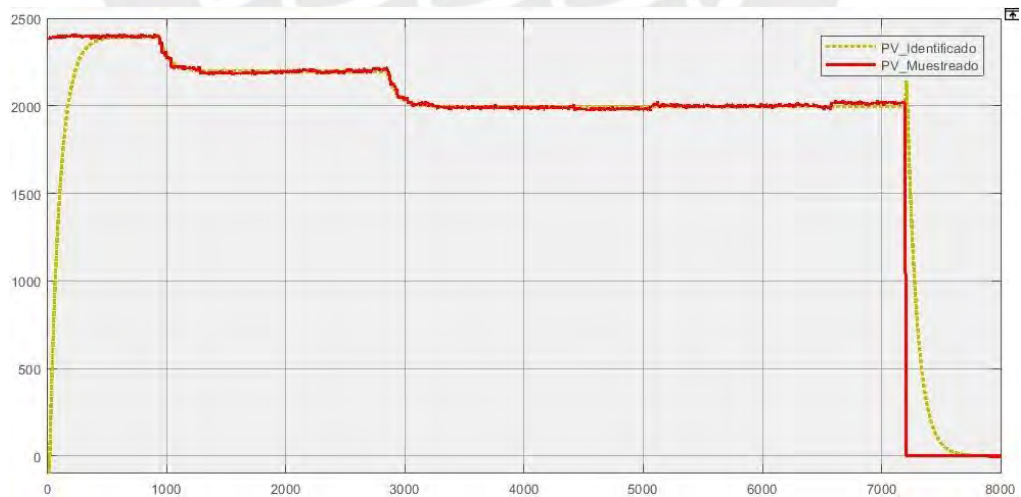
```

**Figura 2.26:** Función de transferencia  $G_p(s)$  calculado en Matlab

En la figura 2.27 se muestra un modelo en Simulink que permite poder comparar el muestreo del SP y la PV (ver figura 2.18) con el modelo obtenido (ver figura 2.26). Además, en la figura 2.28 se muestra la comparación entre la PV muestreada con la PV identificada.

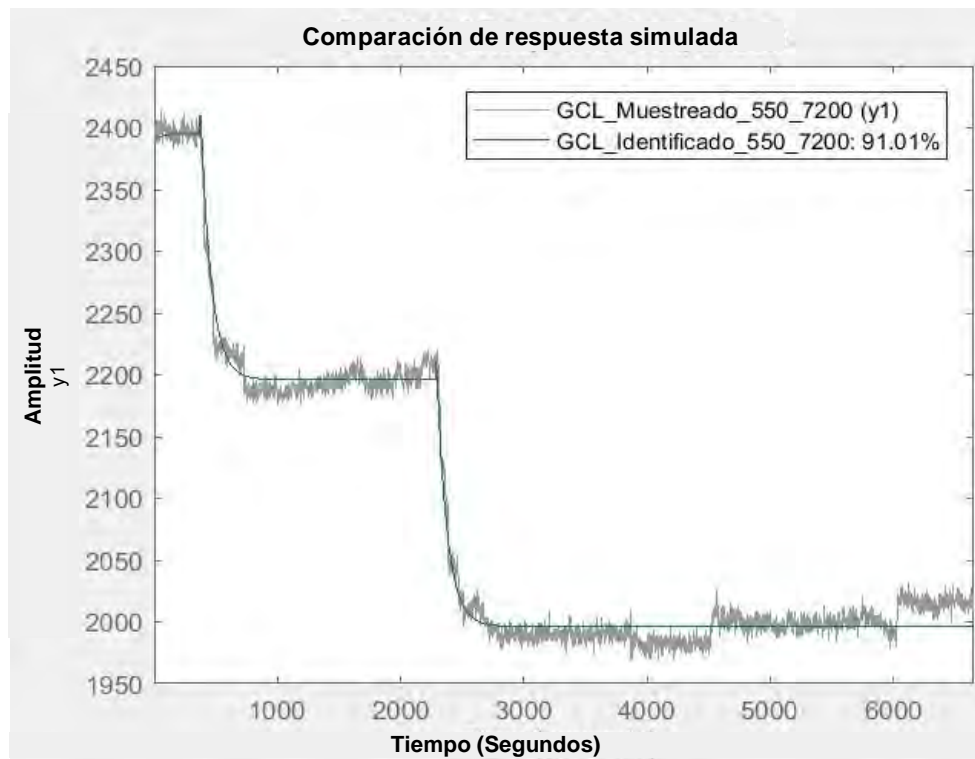


**Figura 2.27:** Modelo en Simulink para validar la identificación de  $G_p(s)$



**Figura 2.28:** Comparación en Simulink de los datos reales muestreados con la respuesta de la identificación de  $G_p(s)$

Como se puede observar en la figura 2.28 la respuesta del modelo identificado se aproxima bien a la respuesta muestreada, claro está que al no tener condiciones iniciales le toma 550 segundos en alcanzar el SP de 2500 y que cuando llega a 7200 segundos ya no se cuenta con valores muestreados. Tomando en cuenta lo anterior se obtiene una estimación de los datos del 91.01% al comparar los datos reales (PV muestreado) con la respuesta de la identificación de  $G_p(s)$  (PV identificado) como se aprecia en la figura 2.29.



**Figura 2.29:** Comparación de la PV muestreada con la PV identificada

Como se puede apreciar este último método de identificación alcanza una estimación de datos del 91.01% muy por encima a lo encontrado con la identificación usando ARMAX (71.84%) y usando ARX (69.13%). Esto básicamente se debe a que este último método trabaja con la entrada de SP que es de tipo escalón la cual es estable y además la PV responde de acuerdo con los cambios en el SP lo que hace que la identificación tenga una mejor estimación.

## 2.4 Conclusiones del capítulo

- La identificación usando los métodos de ARX, ARMAX e “identificación de proceso con controlador conocido” realizan una excelente estimación de los datos para el modelo teórico linealizado de un proceso de flujo con válvula de diafragma.
- Los datos reales muestreados del proceso de flujo que se analiza muestran que la válvula está trabajando en un rango no lineal y que la PV depende también de otras variables del proceso de molienda, por este motivo los modelos ARX y ARMAX no obtienen una estimación de datos que sea adecuada ya que la PV no cambia de acuerdo con las variaciones en la MV.
- El método de “identificación de proceso con controlador conocido” alcanzó un 99.85% de estimación de los datos sobre el proceso teórico y un 91.01% sobre el proceso real, por este motivo este método fue el elegido para utilizarlo en la sintonización con algoritmos genéticos.
- En una planta concentradora de cobre y en general en toda industria siempre se conoce el tipo de controlador PID y sus parámetros configurados. Y generalmente se cuenta con el histórico de datos del SP, la MV y la PV, de los cuales es preferible trabajar con el SP y la PV porque son variables más estables y las variaciones del SP y PV siempre guardan relación.

## CAPÍTULO 3: SINTONIZACION DE UN CONTROLADOR PID UTILIZANDO ALGORITMOS GENETICOS

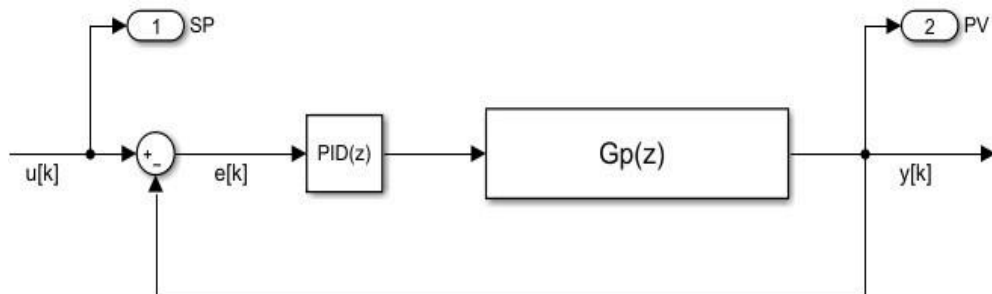
### 3.1 Introducción

Ahora que ya se escogió el método de identificación ya es posible realizar la sintonización usando algoritmos genéticos y es lo que se hará en el presente capítulo. Primero, se va a detallar cual es el programa que se ha realizado para la sintonía basada en algoritmos genéticos sobre un proceso teórico de control de flujo con válvula de diafragma. Segundo, se analizarán diferentes índices de desempeño y se escogerá la función de aptitud que permite encontrar la mejor respuesta del sistema en lazo cerrado y que debe utilizarse para las pruebas en un proceso real. Por último, se presenta las conclusiones del capítulo 3.

### 3.2 Método de algoritmos genéticos

#### 3.2.1 Modelamiento en tiempo discreto

Se va a realizar la sintonización basada en algoritmos genéticos del controlador PID del proceso teórico de flujo de segundo orden analizado en la subsección 2.2.2, para esto es necesario trabajar en tiempo discreto. En la figura 3.1 se muestra su diagrama de bloques.



**Figura 3.1:** Diagrama de bloques del lazo cerrado de control de flujo en tiempo discreto

Se va a trabajar con los mismos parámetros del controlador PID del proceso que se analizó (ver ecuación 3.1), estos serán los parámetros iniciales del controlador PID que se debe optimizar.

$$K_p = 2, K_I = 1, K_D = 1 \quad (3.1)$$

Tomando en cuenta lo anterior, se tiene que la función de transferencia en el dominio de la frecuencia  $G_p(s)$  es la mostrada en la ecuación 2.19, pero como se debe trabajar en tiempo discreto, es necesario transformar del dominio de “s” a “z” para lo cual se está considerando un tiempo de muestreo de 0.1 segundos, con esta información se obtiene la función de transferencia en tiempo discreto  $G_p(z)$  la cual se muestra en la ecuación 3.2.

$$G_p(z) = \frac{a_7 z^7 + a_6 z^6 + a_5 z^5 + a_4 z^4 + a_3 z^3 + a_2 z^2 + a_1 z}{z^8 + b_7 z^7 + b_6 z^6 + b_5 z^5 + b_4 z^4 + b_3 z^3 + b_2 z^2 + b_1 z + b_0} \quad (3.2)$$

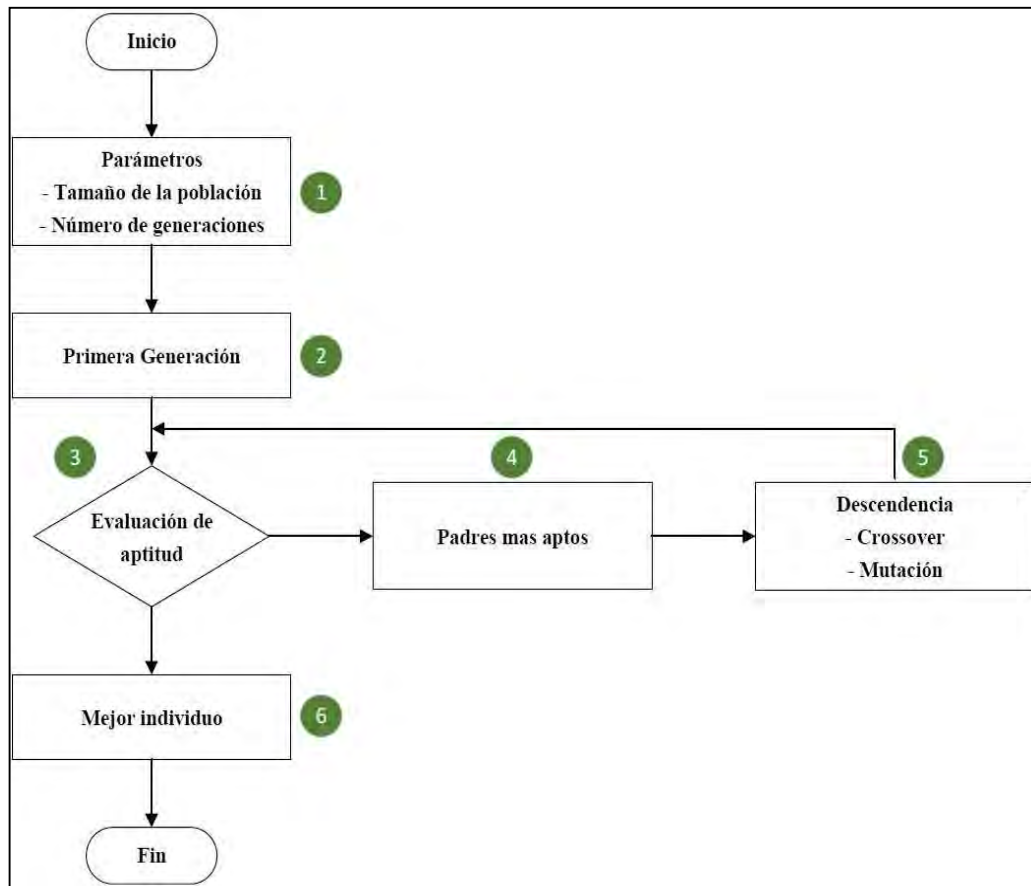
Donde los valores de los coeficientes del numerador y denominador se pueden ver en la figura 3.2.

```
Gp =
0.009963 z^7 - 0.03594 z^6 + 0.04637 z^5 - 0.02239 z^4 - 0.0008928 z^3 + 0.003417 z^2 - 0.0005248 z + 3.863e-07
-----
z^8 - 5.535 z^7 + 12.76 z^6 - 15.68 z^5 + 10.84 z^4 - 4 z^3 + 0.6173 z^2 - 0.001123 z + 4.945e-07
Sample time: 0.1 seconds
Discrete-time transfer function.
```

**Figura 3.2:** Función de transferencia  $G_p(z)$  calculado en Matlab

### 3.2.2 Programación del método de algoritmos genéticos

Para realizar la sintonización se va a utilizar la técnica de algoritmos genéticos simples cuyo diagrama de flujo se muestra en la figura 3.3 en donde se identifica con un número cada etapa del proceso.



**Figura 3.3:** Diagrama de flujo de la técnica de algoritmos genéticos

### 3.2.2.1 Etapa 1-Parámetros

En esta etapa se define las dimensiones de la matriz  $\mathbf{P}$  (población), la cual es el conjunto de posibles soluciones que pueden minimizar la función de aptitud, para poder realizar la sintonización del controlador PID, se necesita que la población este definida por los parámetros  $K_p$ ,  $K_I$  y  $K_D$ , esta matriz se muestra en la ecuación 3.3.

$$\mathbf{P} = \begin{bmatrix} K_p \\ K_I \\ K_D \end{bmatrix}_{3 \times n} = \begin{bmatrix} K_{p1} & K_{p2} & \cdot & \cdot & K_{pn} \\ K_{I1} & K_{I2} & \cdot & \cdot & K_{In} \\ K_{D1} & K_{D2} & \cdot & \cdot & K_{Dn} \end{bmatrix}_{3 \times n} \quad (3.3)$$

La matriz  $\mathbf{P}$  tiene  $n$  individuos y cada individuo es una posible solución y es una columna de la matriz  $\mathbf{P}$  lo que implica que el tamaño de la población es de  $n$ .

Cada matriz **P** también representa una generación, y cada una de estas es evaluada en la función de aptitud para poder encontrar al mejor individuo, siempre es necesario definir el número de generaciones que deberán ser evaluadas.

En la figura 3.4 se muestra el tamaño de la población y el número de generaciones que se encontraron apropiados para la sintonización del proceso teórico de flujo.

```
%% Parametros del Algoritmo Genetico

% Vector P=[Kp; Ki; Kd]
% Dimensión del vector P es 3 x n

n=42000; % Tamano de la poblacion
n_ite=10; % Numero de generaciones o iteraciones
```

**Figura 3.4:** Definición de los parámetros del algoritmo genético

### 3.2.2.2 Etapa 2-Primera generación

En esta etapa se define el primer conjunto de posibles soluciones que pueden minimizar la función de costo. Esta va a tener como primer individuo a los parámetros  $K_p$ ,  $K_I$  y  $K_D$  configurados inicialmente en el PID, los siguientes individuos serán una variación de estos parámetros.

En la figura 3.5 se muestra cual es la primera generación, esta se define de acuerdo a la respuesta que tiene la PV con el controlador PID actual.

```
%% Primera Generación

Kp_i=C(1); % Kp inicial configurado
Ki_i=C(1)/C(2); % Ki inicial configurado
Kd_i=C(1)*C(3); % Kd inicial configurado

P(1,1:n)=2*Kp_i*rand(1,n); % Vector de valores Kp
P(2,1:n)=2*Ki_i*rand(1,n); % Vector de valores Ki
P(3,1:n)=2*Kd_i*rand(1,n); % Vector de valores Kd

P(1,1)=Kp_i;
P(2,1)=Ki_i;
P(3,1)=Kd_i;
```

**Figura 3.5:** Definición de los parámetros del algoritmo genético

### 3.2.2.3 Etapa 3-Evaluación de aptitud

En esta etapa se calcula la función de aptitud  $FA$  para cada individuo de una generación. Como lo que se necesita minimizar es el error entre el SP y la PV, se escoge en primera instancia como función de aptitud al error cuadrático integral (ISE) el cual se puede ver en la ecuación 3.4.

$$FA = ISE = \int_{t_0}^{t_f} e(t)^2 dt \quad (3.4)$$

En la figura 3.6 se muestra cómo se realiza el cálculo de la función de aptitud, tomando en cuenta la función de transferencia en “z” del proceso teórico de flujo mostrado en la figura 3.2.

```
%% Cálculo en discreto
for t=length(numz):tm
    % Respuesta y error del modelo teórico en lazo cerrado
    y(t)=-denz(2)*y(t-1)-denz(3)*y(t-2)-denz(4)*y(t-3)-denz(5)*y(t-4)...
        -denz(6)*y(t-5)-denz(7)*y(t-6)-denz(8)*y(t-7)-denz(9)*y(t-8)...
        +numz(9)*u(t-8)+numz(8)*u(t-7)+numz(7)*u(t-6)+numz(6)*u(t-5)...
        +numz(5)*u(t-4)+numz(4)*u(t-3)+numz(3)*u(t-2)+numz(2)*u(t-1)...
        +numz(1)*u(t);
    e(t)=ref-y(t);

    % Salida de PID
    integral=integral+Ki*e(t)*ts;
    u(t)=Kp*e(t) + integral + Kd*(e(t)-e(t-1))*1/(ts) + p(t) ;

    % Cálculo del error integral
    if op==1
        e1(t)=e(t)*e(t);
        ISE=ISE+e1(t);
    end
end

FA=ISE;
```

**Figura 3.6:** Cálculo de la función de aptitud por cada individuo

### 3.2.2.4 Etapa 4-Padres más aptos

En esta etapa se encuentran los individuos que obtuvieron los mejores resultados por cada generación, esto se obtiene ordenando de menor a mayor el vector que contiene todos los valores de *FA* de la generación.

En la figura 3.7 se muestra cómo se realiza el ordenamiento para encontrar a los mejores individuos de cada generación.

```
%% Algoritmo Genético
% Evaluación por generación
for ite=1:n_ite %n_ite=5

    % Evaluación de toda la población
    for i=1:n
        aptitud_actual(i)=Funcion_Aptitud(P(:,i),numz_Gp,denz_Gp,MAX_SI,...
                                          MIN_SI,op,SP,T,Im,T_s);
    end

    % Ordenamiento ascendente de la función de aptitud
    [FA , ind]=sort(aptitud_actual);

    % Ordenamiento de vector Px, Py, Pz, donde
    % [Kp Ki Kd]=[Px(1) Py(1) Pz(1)] es el mejor individuo de la generación
    for i=1:n
        Px(i)=P(1,ind(i));
        Py(i)=P(2,ind(i));
        Pz(i)=P(3,ind(i));
    end
end
```

Figura 3.7: Obtención de los mejores individuos

### 3.2.2.5 Etapa 5-Descendencia

La descendencia son los individuos de la nueva generación los cuales se obtienen modificando la matriz **P** de la siguiente forma:

- Crossover, se ha establecido que es la mitad de la población de los cuales la octava parte de estos son los padres más aptos de la generación anterior y el resto son una combinación de estos padres los cuales se definen de acuerdo a la respuesta que tiene la PV con el actual controlador PID.

- Mutación, es la segunda mitad de la población la cual se obtiene de los valores aleatorios definidos en un rango específico, estos se definen de acuerdo a la respuesta que tiene la PV con el controlador PID actual.

En la figura 3.8 se muestra cómo se define la descendencia de la generación anterior.

```

% Descendencia (Crossover y Mutacion)

% Crossover
for i=1:n/8
    Hx1(i)=0.25*Px(i)*rand(1);
    Hy1(i)=0.25*Py(i)*rand(1);
    Hz1(i)=0.25*Pz(i)*rand(1);
    Hx2(i)=0.75*Px(i)*rand(1);
    Hy2(i)=0.75*Py(i)*rand(1);
    Hz2(i)=0.75*Pz(i)*rand(1);
    Hx3(i)=1.25*Px(i)*rand(1);
    Hy3(i)=1.25*Py(i)*rand(1);
    Hz3(i)=1.25*Pz(i)*rand(1);
end

P1=Px; P2=Py; P3=Pz;

P1(n/8+1:(n/4))=Hx1;
P2(n/8+1:(n/4))=Hy1;
P3(n/8+1:(n/4))=Hz1;
P1(n/4+1:(3*n/8))=Hx2;
P2(n/4+1:(3*n/8))=Hy2;
P3(n/4+1:(3*n/8))=Hz2;
P1((3*n/8)+1:n/2)=Hx3;
P2((3*n/8)+1:n/2)=Hy3;
P3((3*n/8)+1:n/2)=Hz3;

% Mutacion
P1((n/2)+1:3*n/4)=1.5*rand(1,n/4);
P2((n/2)+1:3*n/4)=1.5*rand(1,n/4);
P3((n/2)+1:3*n/4)=1.5*rand(1,n/4);
P1((3*n/4)+1:n)=2.5*rand(1,n/4);
P2((3*n/4)+1:n)=2.5*rand(1,n/4);
P3((3*n/4)+1:n)=2.5*rand(1,n/4);

% Siguiete generacion
P(1,1:n)=P1;
P(2,1:n)=P2;
P(3,1:n)=P3;

end

```

**Figura 3.8:** Descendencia de la generación anterior

### 3.2.2.6 Etapa 6-Mejor individuo

Una vez que se evalúan todas las generaciones se tiene que el mejor individuo se encuentra en la primera columna de la matriz **P** y estos valores son los parámetros sintonizados para el controlador PID los cuales son los que el algoritmo ha encontrado que minimizan mejor la función de aptitud, esto implica que con estos parámetros se obtendrá una mejor respuesta de la PV. En la figura 3.9 se muestra cómo se obtiene estos parámetros.

```
%% Parametros del PID sintonizado
Kp=P1(1);
Ki=P2(1); Ti=Kp/Ki;
Kd=P3(1); Td=Kd/Kp;
```

**Figura 3.9:** Parámetros sintonizados del controlador PID

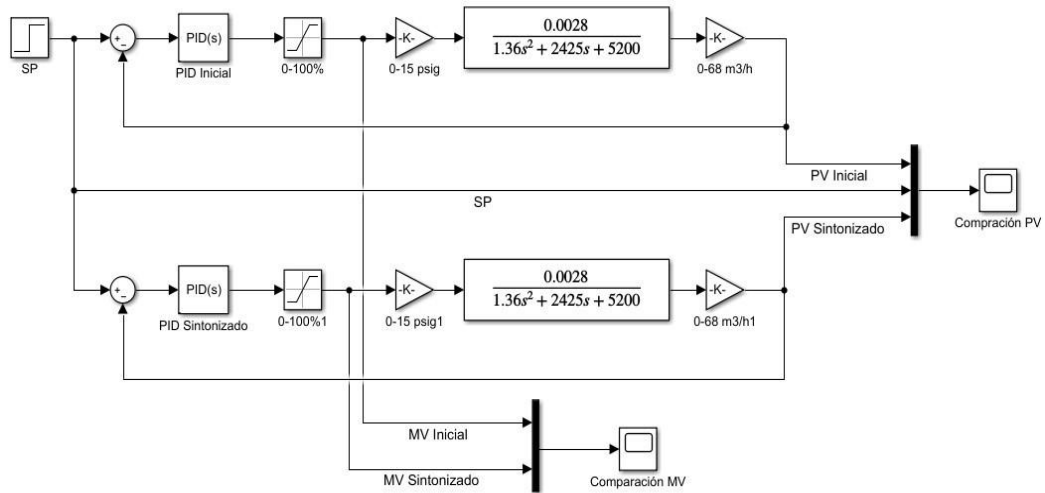
## 3.3 Sintonización del controlador PID

### 3.3.1 Índice de desempeño como función de aptitud

Ahora que ya está elaborada la programación de la sintonización de un controlador PID basado en el método de algoritmos genéticos, ya se puede encontrar los nuevos parámetros del controlador PID para el proceso teórico de flujo, para esta sintonización se está tomando como función de aptitud el error cuadrático integral que se indica en la ecuación 3.4 y además un SP que cambia de 0 a 50 m<sup>3</sup>/h. Al ejecutar el programa se encuentra los siguientes parámetros para el controlador PID.

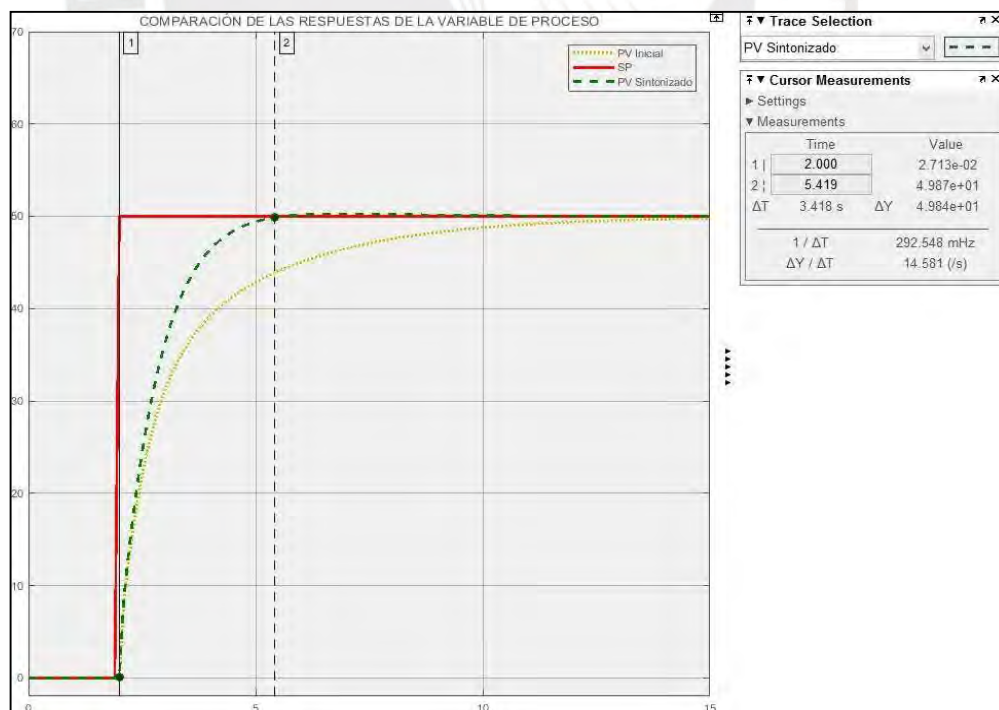
$$K_p = 4.4093, K_i = 2.0362, K_D = 2.9397 \quad (3.5)$$

En la figura 3.10 se muestra un modelo en Simulink que permitirá observar la respuesta de la PV y la MV que se obtiene con los parámetros sintonizados del controlador PID comparado con la respuesta de la PV y la MV que se obtiene con los parámetros iniciales.



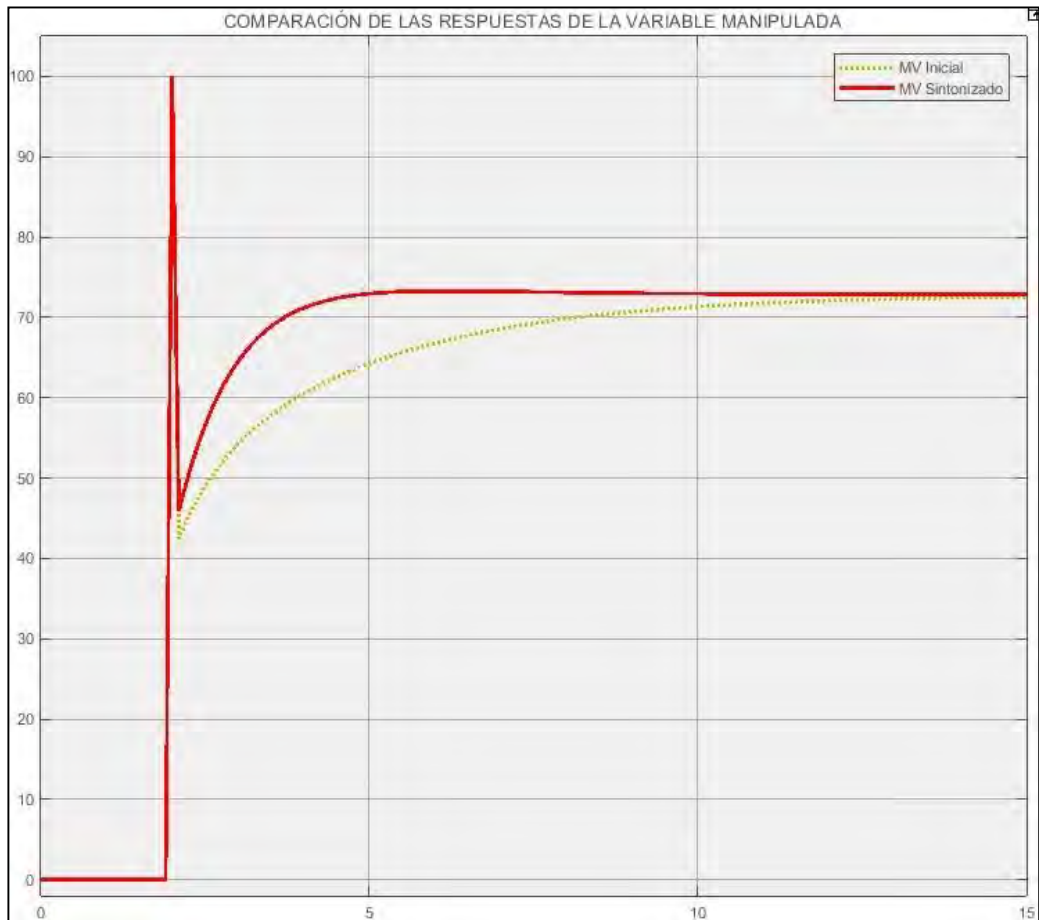
**Figura 3.10:** Modelo en Simulink para comparación de la PV y la MV

En la figura 3.11 se puede apreciar que con los parámetros sintonizados del controlador PID se ha obtenido un tiempo de subida y de establecimiento de 3.418 segundos, el cual es una respuesta más rápida que la respuesta inicial que era de 11 segundos, también se ha obtenido un máximo sobreimpulso del 0%.



**Figura 3.11:** Respuesta de la PV sintonizada y la PV inicial

En la figura 3.12 se puede observar que la respuesta de la MV que se obtiene con los parámetros sintonizados del controlador PID es más rápida a la obtenida con la respuesta inicial, es decir que la acción de control es mayor con estos nuevos parámetros.



**Figura 3.12:** Respuesta de la MV sintonizada y la MV inicial

Con los parámetros sintonizados se obtiene una respuesta más rápida pero está aún se puede mejorar ya que se ha obtenido un máximo sobreimpulso del 0% y la acción de control aunque es mayor que la inicial aún puede aumentarse sin afectar el actuador, esto implica que esta función de aptitud no es suficiente y se debe básicamente a que solo se ha considerado minimizar el error entre la PV y el SP, pero no se ha tomado en cuenta cuales son las especificaciones de la respuesta transitoria que deben de cumplirse.

### 3.3.2 Elección de la función de aptitud

Como se pudo ver en la subsección anterior, no basta solo con usar el índice de desempeño como función de aptitud para poder encontrar los parámetros sintonizados del controlador PID que permitan una respuesta transitoria adecuada, por este motivo es necesario agregar a la función de aptitud las especificaciones de la respuesta transitoria.

#### 3.3.2.1 Elección del índice de desempeño

Para escoger el índice de desempeño adecuado para la función de aptitud se va a realizar el análisis los siguientes índices mostrados en la subsección 1.4.

- *ISE* → Error cuadrático integral
- *IAE* → Error absoluto integral
- *ITSE* → Error de tiempo integrado
- *ITAE* → Error absoluto de tiempo integrado

Al programa realizado en la subsección 3.2.2 se agrega el cálculo de los índices de desempeño IAE, ITSE, ITAE, esto se puede ver en la figura 3.13.

```
% Cálculo del error integral
if op==1
    e1(t)=e(t)*e(t);
    ISE=ISE+e1(t);
end

if op==2
    e2(t)=abs(e(t));
    IAE=IAE+e2(t);
end

if op==3
    e3(t)=t*(e(t)*e(t));
    ITSE=ITSE+e3(t);
end

if op==4
    e4(t)=t*abs(e(t));
    ITAE=ITAE+e4(t);
end
```

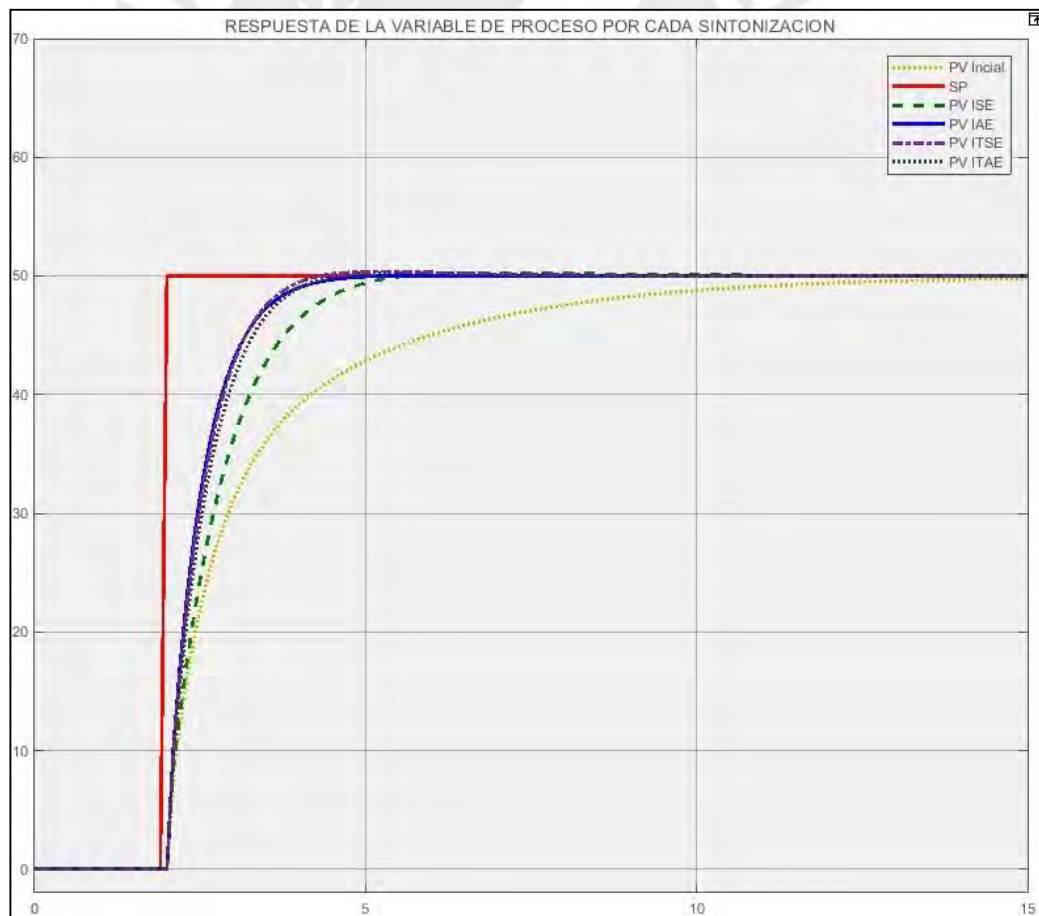
**Figura 3.13:** Calculo del error integral

En la tabla 3.1 se muestra los parámetros sintonizados que se han encontrado para cada índice de desempeño y también se muestra los parámetros iniciales.

Parámetros	ISE	IAE	ITSE	ITAE	Inicial
$K_p$	4.4093	4.3349	4.5962	3.4399	2
$K_I$	2.0362	2.9167	3.0220	2.7270	1
$K_D$	2.9397	1.5732	1.8739	1.4220	1

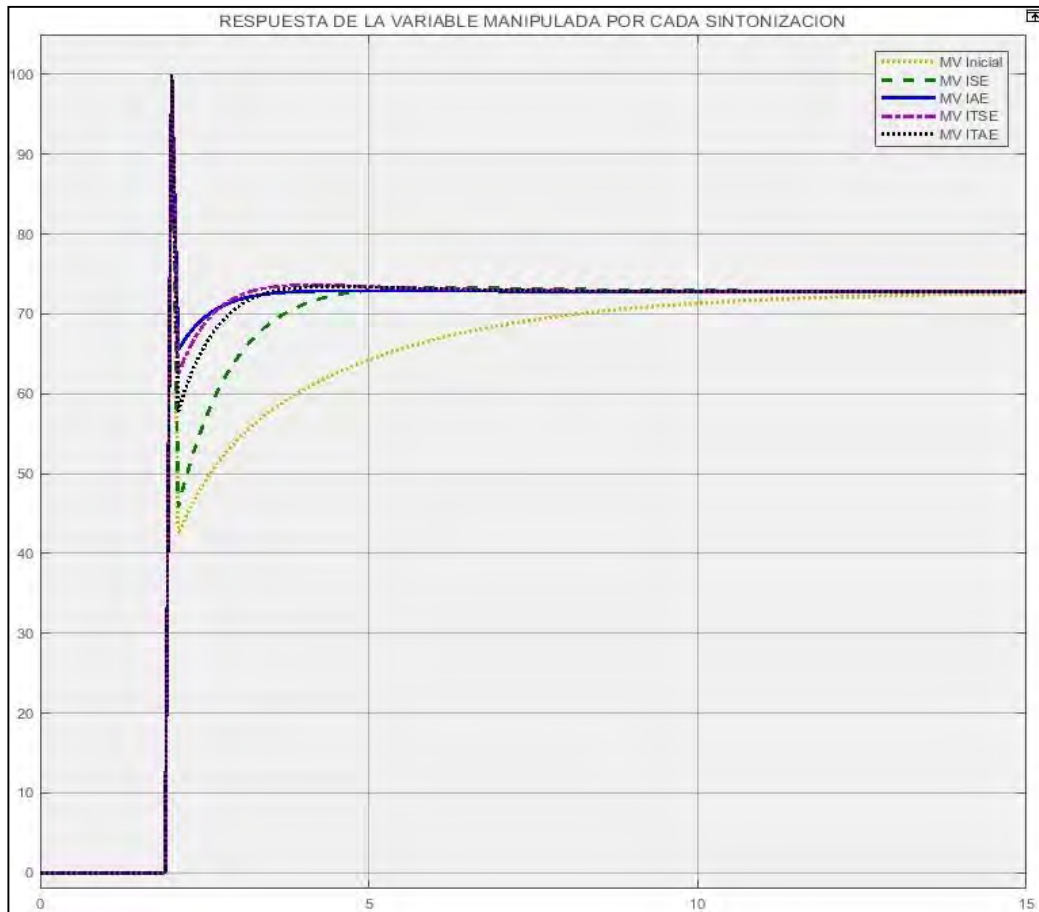
**Tabla 3.1:** Parámetros sintonizados del controlador PID

En la figura 3.14 se muestra la respuesta transitoria para cada uno de estos parámetros sintonizados encontrando que las respuestas obtenidas con el índice de desempeño IAE, ITSE e ITAE son las mejores.



**Figura 3.14:** Respuesta de la variable de proceso por cada índice de desempeño

También se encuentra que con el índice de desempeño IAE se obtiene una mejor respuesta de la variable manipulada y esto se puede observar en la figura 3.15.



**Figura 3.15:** Respuesta de la variable manipulada por cada índice de desempeño

Analizando las respuestas de la PV y la MV se escoge el IAE como índice de desempeño. La respuesta transitoria encontrada con este índice tiene las siguientes especificaciones:

- $t_r = 2.304 \text{ seg}$  → Tiempo de subida
- $M_p = 0 \%$  → Máximo sobreimpulso
- $t_s = 2.304 \text{ seg}$  → Tiempo de establecimiento
- $e_{es} \approx 0$  → Error en estado estacionario
- Respuesta → Sobre amortiguada

Para el caso del índice de desempeño normalizado de Harris visto en la subsección 1.4 se tiene que no puede ser usado en la función de aptitud ya que este índice solo indica que para cualquier valor cercano a cero se tiene una buena sintonización mas no diferencia cual valor da una mejor respuesta, es decir obtener un valor de 0.1 no implica una mejor respuesta que para un valor de 0.2.

Se debe tomar en cuenta que al ejecutar el programa basado en algoritmos genéticos se va a poder encontrar valores un poco diferentes de  $K_p$ ,  $K_I$  y  $K_D$  para el mismo índice de desempeño IAE, pero se encontró que siempre se obtienen especificaciones de la respuesta transitoria similares y su respuesta en general es ligeramente mejor tanto para la PV como para la MV. Por tal motivo se escoge el índice de desempeño IAE.

### 3.3.2.2 Elección de las especificaciones de la respuesta transitoria

Usando solo el índice de desempeño como función de aptitud se encuentra una mejor respuesta de la PV, pero solo se buscó minimizar el error mas no cumplir con las especificaciones de la respuesta transitoria, cumplir con estas especificaciones es importante para el proceso por lo que es necesario agregar las especificaciones de la respuesta transitoria a la función de aptitud, cada proceso tiene diferentes especificaciones, para este caso teórico he definido lo siguiente:

- $t_r < 2 \text{ seg}$  → Tiempo de subida
- $1\% < M_p < 4\%$  → Máximo sobreimpulso
- $t_s < 5 \text{ seg}$  → Tiempo de establecimiento
- $e_{es} \approx 0$  → Error en estado estacionario
- Respuesta → Sub amortiguada

Dentro de estas especificaciones las más importantes de cumplir casi estrictamente son el  $e_{es}$  y el  $M_p$ , para el caso del  $e_{es}$  es suficiente usar el índice de desempeño IAE ya que con este índice se busca minimizar el error y siempre se va a encontrar que el  $e_{es}$  se aproximara a cero, para el caso del  $M_p$  es necesario definir que este siempre debe

ser un valor superior a cero para evitar obtener una respuesta sobre amortiguada si es que esta no se desea y su valor máximo debe ser por lo menos el indicado en las especificaciones.

Las especificaciones  $t_r$  y  $t_s$  no pueden cumplirse estrictamente ya que estas pueden afectar considerablemente al  $M_p$ , por ejemplo, si consideramos un  $t_r$  muy bajo esto puede implicar encontrar un valor alto  $M_p$ , por tal motivo se busca cumplir estas especificaciones indirectamente estableciendo un rango de tiempo en el cual se debe encontrar el máximo sobreimpulso. En la figura 3.16 se muestra los valores considerados para la condición del máximo sobreimpulso en un determinado rango de tiempo.

```
%% Especificaciones del proceso
SP=50; % Cambio en el Set Point
T_s=60; % Rango de tiempo en el cual se encuentra el Máximo Sobreimpulso
        % Tiempo de muestreo T es de 0.1 segundos
        % Rango de tiempo escogido es de 50*0.1 = 5 segundos
MAX_SI=0.05*SP; %porcentaje de Sobre Impulso MAXIMO
MIN_SI=0.02*SP; %porcentaje de Sobre Impulso MINIMO

%% Máximo sobreimpulso (Mp)
Y_s=y(1:T_s); % FV en el rango de tiempo T_s
SI=max(Y_s)-ref; % ref es el SP
if MAX_SI>SI && SI>MIN_SI
    Mp=100*SI/ref;
else
    Mp=1e10; % Valor grande para desestimar
end
```

**Figura 3.16:** Condición de máximo sobreimpulso

Entonces se tendrá que la función de aptitud a minimizar tendrá el IAE y la condición de máximo sobre impulso, esto se puede observar en la figura 3.17

```
fc_2=IAE/10+10*Mp; % fc_2: Función de aptitud
                    % IAE: Error Absoluto Integral
                    % Mp: Máximo sobreimpulso
                    % 1/10 y 10 son los pesos
```

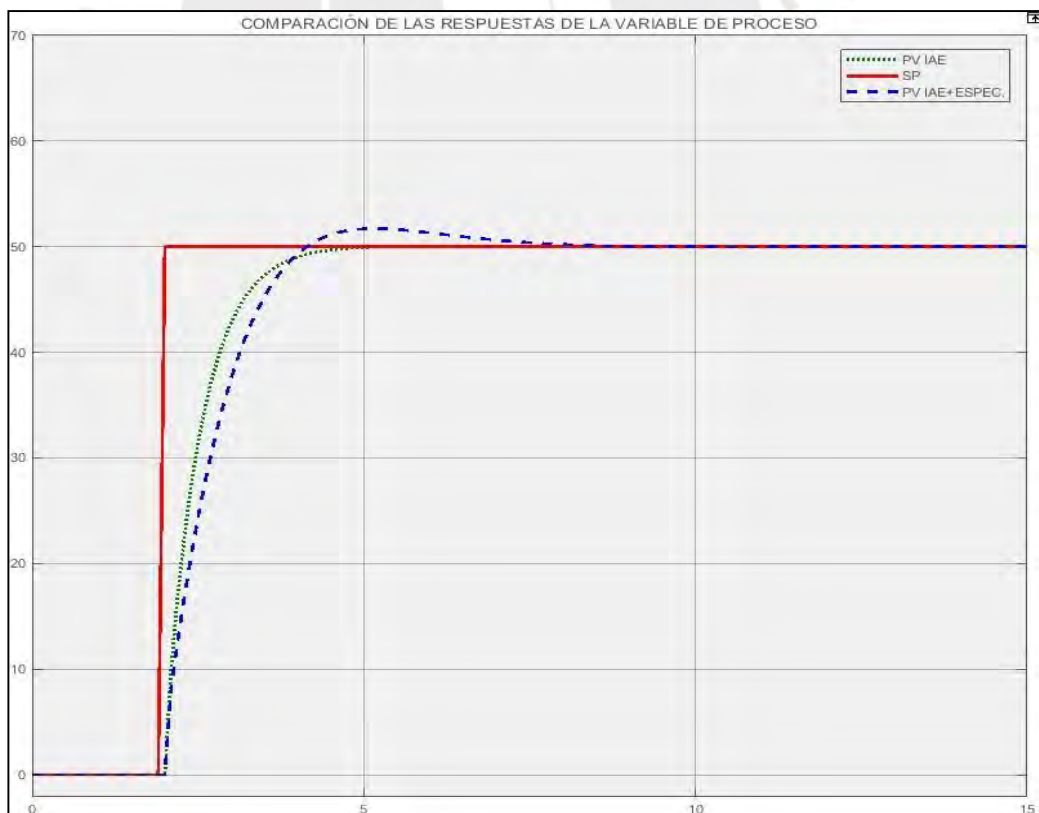
**Figura 3.17:** Función de aptitud

Ahora que se tiene la función de aptitud que considera tanto el índice de desempeño como las especificaciones de la respuesta transitoria, ya se puede encontrar los nuevos parámetros sintonizados. En la tabla 3.2 se compara estos parámetros con los que se encontraron en la tabla 3.1 para el IAE.

Función de aptitud	IAE	IAE + Especificaciones
$K_p$	4.3349	2.6596
$K_I$	2.9167	2.5653
$K_D$	1.5732	1.6590

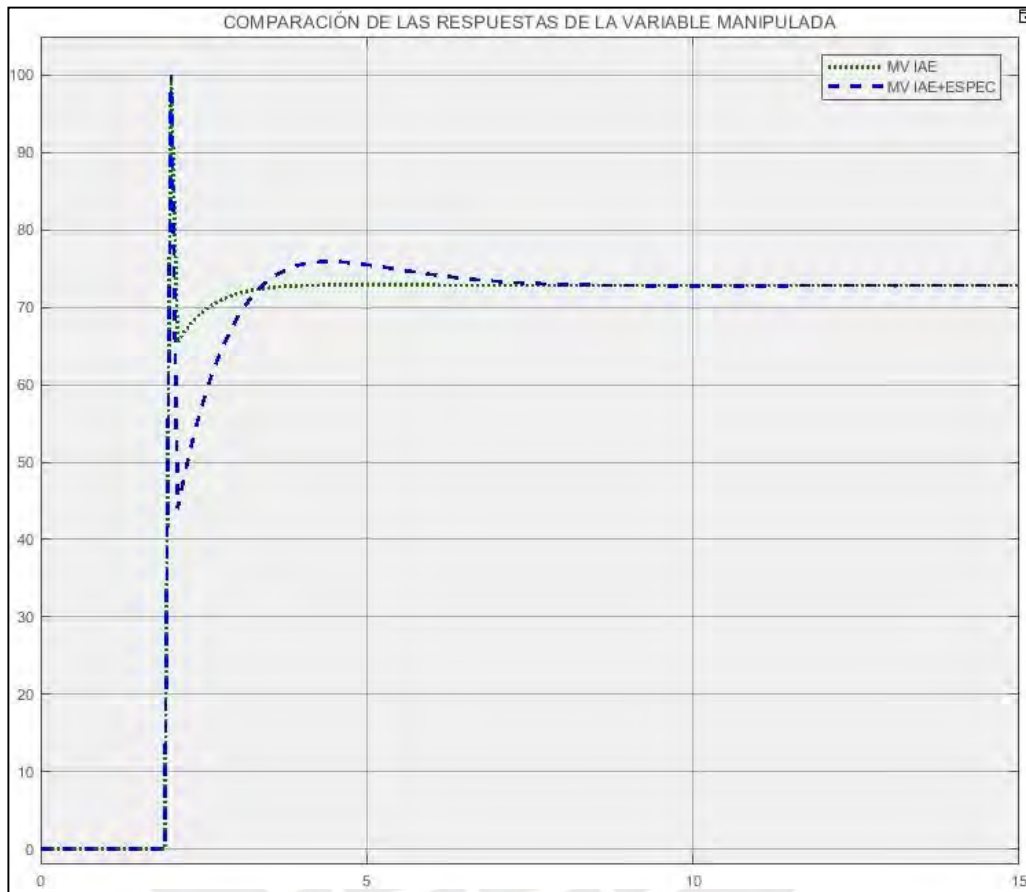
**Tabla 3.2:** Parámetros sintonizados del controlador PID

En la figura 3.18 se muestra la respuesta transitoria para cada una de las funciones de aptitud donde se puede observar que se obtiene una respuesta sub amortiguada cuando las especificaciones son agregadas.



**Figura 3.18:** Respuesta de la variable de proceso por cada función de aptitud

En la figura 3.19 se muestra la respuesta de la variable manipulada en la cual se puede observar que al agregar las especificaciones de la respuesta transitoria se obtiene una acción de control más rápida, lo que va a implicar obtener un menor tiempo de subida.



**Figura 3.19:** Respuesta de la variable manipulada por cada función de aptitud

La respuesta transitoria que se obtiene al usar la función de aptitud mostrada en la figura 3.17 tiene las siguientes especificaciones:

- $t_r = 2.145 \text{ seg}$  → Tiempo de subida
- $M_p = 3.34 \%$  → Máximo sobreimpulso
- $t_s = 5.229 \text{ seg}$  → Tiempo de establecimiento
- $e_{es} \approx 0$  → Error en estado estacionario
- Respuesta → Sub amortiguada

Se puede observar que las especificaciones cumplen estrictamente con el  $M_p$  y  $e_{es}$  que habían sido ya definidas, el  $t_r$  y  $t_s$  tienen un valor cercano al rango definido lo cual es suficiente, además se ha obtenido una respuesta sub amortiguada como se deseaba tener. Por estos motivos es necesario usar las especificaciones de la respuesta transitoria y el índice de desempeño para formar la función de aptitud la cual se muestra en la ecuación 3.6 y es la elegida para realizar la sintonización basada en algoritmos genéticos.

$$FA = a * IAE + b * M_p \quad (3.6)$$

Donde:

$$IAE = \int_{t_0}^{t_f} |e(t)| dt = \int_{t_0}^{t_f} |PV(t) - SP(t)| dt \quad (3.7)$$

$$M_p = \left( \frac{PV_{max} - SP}{SP} \right) \quad (3.8)$$

Además:

- $FA$  → Función de aptitud
- $IAE$  → Error absoluto integral
- $M_p$  → Máximo sobreimpulso
- $PV$  → Variable de proceso
- $PV_{max}$  → PV máximo en un rango de tiempo  $T_s$
- $T_s$  → Valor entre  $t_r$  y  $t_s$
- $SP$  → Set Point
- $a$  → Peso del IAE
- $b$  → Peso del MP

### 3.4 Conclusiones del capítulo

- Para poder escoger la primera generación, descendencia y mutación del algoritmo genético es importante conocer el proceso y evaluar su respuesta transitoria que se obtiene con los parámetros iniciales del controlador PID.
- Usando solo el índice de desempeño como función de aptitud se encuentra una mejor respuesta transitoria que la obtenida inicialmente, esto sucede para todos los índices ISE, IAE, ITSE e ITAE.
- La respuesta transitoria encontrada con el índice de desempeño IAE fue ligeramente mejor que la encontrada con los índices ISE, ITSE e ITAE. Pero aun así no cumple con las especificaciones que se necesitaban para la respuesta transitoria y esto se debe a que minimizar estos índices de desempeño se traduce solo en encontrar el menor error.
- Para cumplir con las especificaciones de la respuesta transitoria es necesario agregar estas a la función de aptitud.
- Se encontró que la respuesta transitoria que cumple con las especificaciones definidas se obtiene cuando la función de aptitud está formada por el índice de desempeño y la especificación del máximo sobreimpulso en un rango de tiempo.

## CAPÍTULO 4: RESULTADOS DE LA SINTONIZACION UTILIZANDO ALGORITMOS GENETICOS

### 4.1 Introducción

En el presente capítulo se aplica el programa realizado para la sintonización basada en algoritmos genéticos en dos plantas reales. Primero, se muestra los resultados que se han obtenido para la planta real de control de flujo analizada en la subsección 2.3 para el cual si fue posible modificar los parámetros del controlador PID en el DCS. Segundo, se muestra los resultados que se han obtenido para una planta de control de nivel que tiene la particularidad que su sensor de nivel tiene un alto nivel de ruido, para este no fue posible modificar los parámetros de controlador PID en el DCS así que solo se muestran simulaciones. Por último, se presenta las conclusiones del capítulo 4.

### 4.2 Resultado de la sintonización en un proceso real de control de flujo

#### 4.2.1 Respuesta transitoria de la PV con el PID inicial

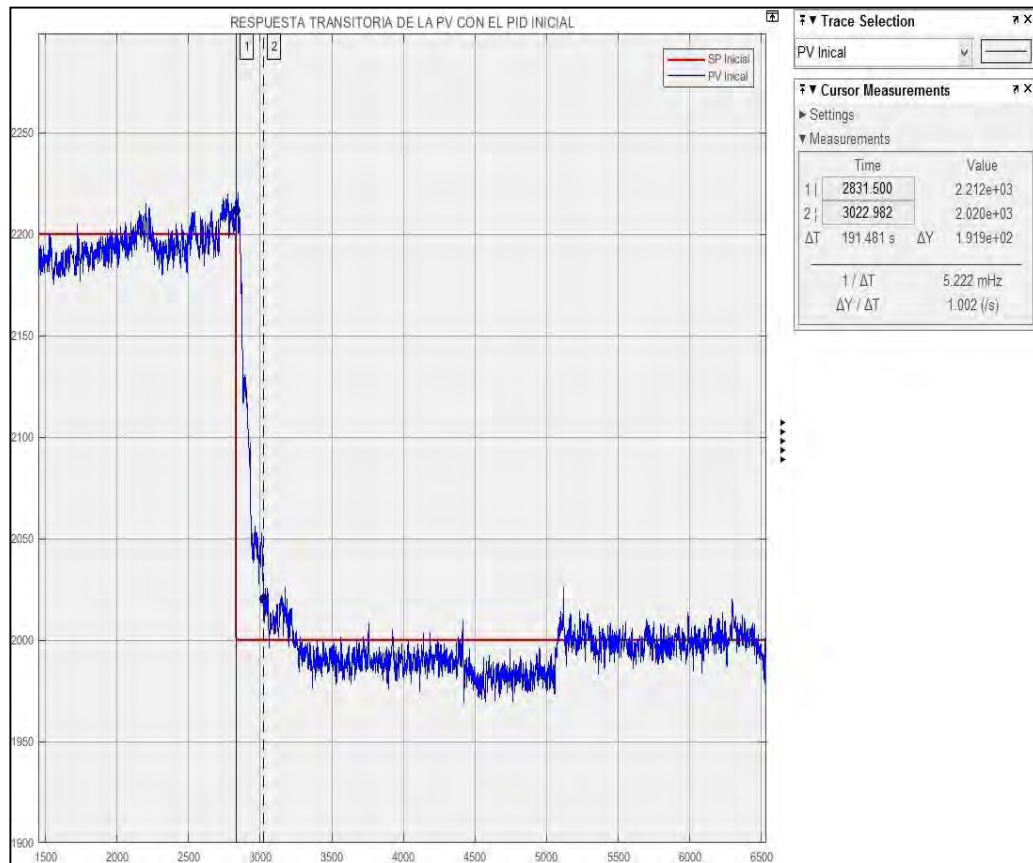
Antes de realizar la sintonización basada en algoritmos genéticos se debe analizar la respuesta transitoria que obtiene del proceso real de control de flujo analizado en la subsección 2.3 el cual tiene configurado en el DCS los siguientes parámetros iniciales del controlador PID.

$$K_p = 0.5, T_i = 30, T_d = 1.1 \quad (4.1)$$

Tomar en cuenta que los parámetros  $K_p$ ,  $T_i$  y  $T_d$  son los que se configuran en el DCS el cual trabaja con un PID tipo paralelo. En la ecuación 4.2 se muestra como expresa el DCS la función de transferencia del PID.

$$\text{PID}(s) = 0.5 \left( 1 + \frac{1}{30} * \frac{1}{s} + 1.1s \right) \quad (4.2)$$

Como ya se indicó en la subsección 2.3 se han obtenido de la planta los datos reales del SP, la MV y la PV a un tiempo de muestreo de 1 segundo. En la figura 4.1 se muestra la respuesta transitoria de la PV de 1500 a 6500 segundos.

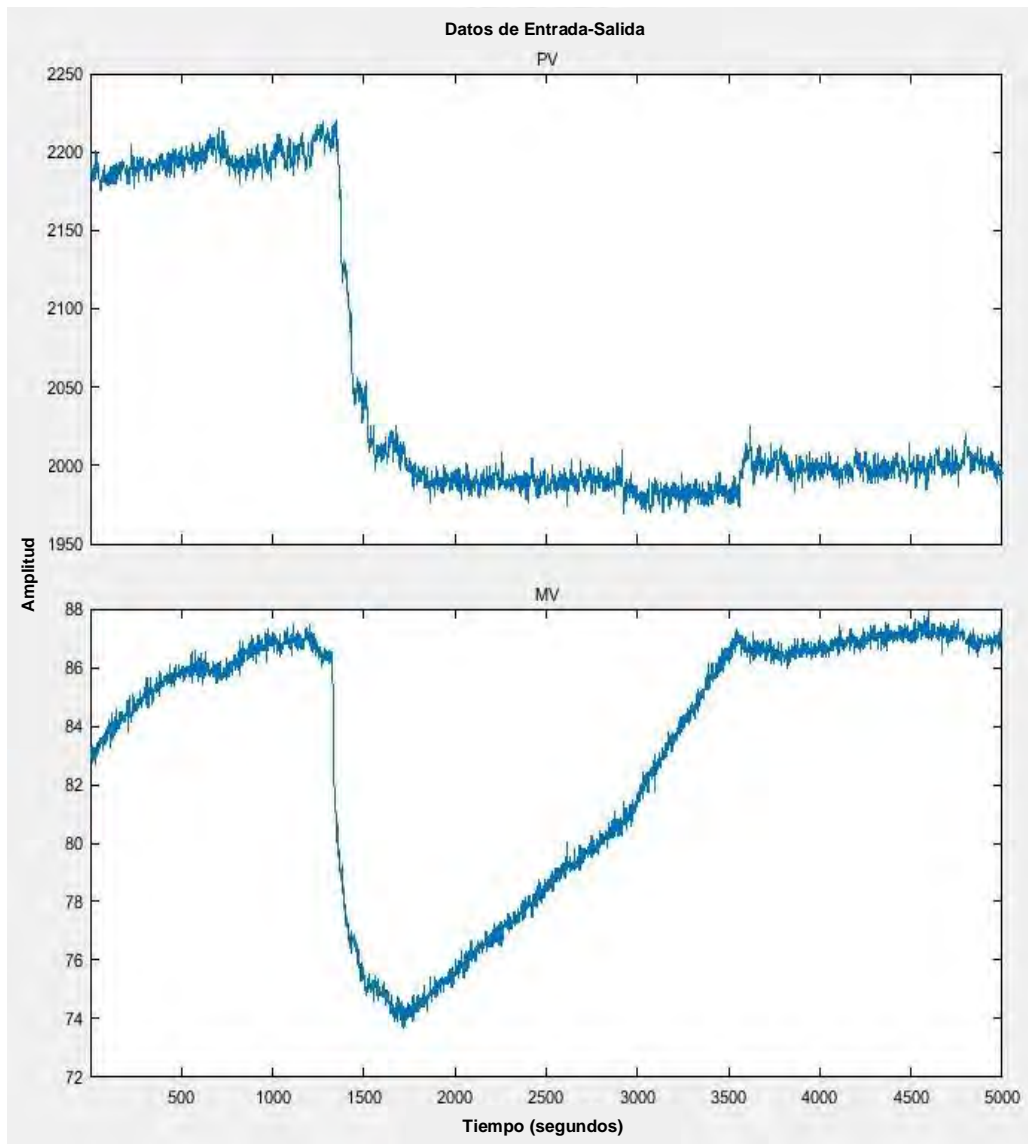


**Figura 4.1:** Respuesta de la PV con el PID inicial

Debido al comportamiento de la PV en este rango de tiempo, se considera que el tiempo de establecimiento se encuentra cuando se alcanza un error aproximado del 1%. Las especificaciones de la respuesta transitoria son las siguientes:

- $t_r = 191.481 \text{ seg}$  → Tiempo de subida
- $M_p = 0 \%$  → Máximo sobreimpulso
- $t_s = 191.481 \text{ seg}$  → Tiempo de establecimiento
- $e_{es} \approx 1\%$  → Error en estado estacionario
- Respuesta → Sobre amortiguada

En la figura 4.2 se muestra el comportamiento que tienen la PV con respecto a la MV donde se aprecia claramente que hay factores externos que afectan el comportamiento de la PV como es la bomba de alimentación de agua cruda o las decisiones que toma el operador y/o sistema experto en todo el proceso de molienda, también se puede ver que solo ante cambios bruscos de la MV se obtiene un cambio significativo en la PV y esto se debe al comportamiento no lineal de la válvula, así como también a la calibración de la retroalimentación de la posición de la válvula al DCS.

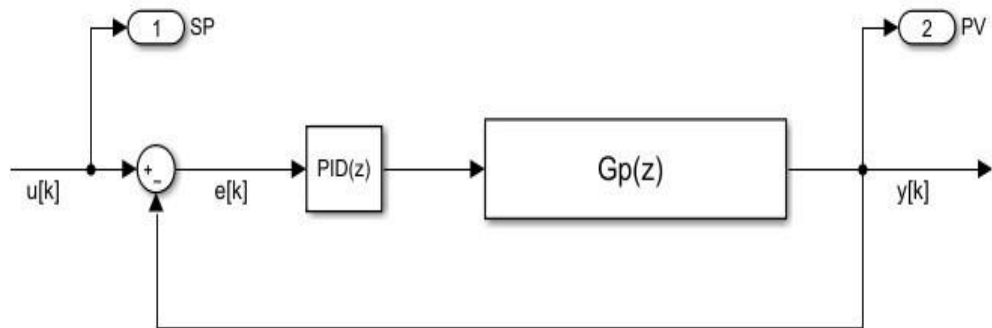


**Figura 4.2:** Comportamiento de la PV con respecto a la MV

## 4.2.2 Respuesta transitoria de la PV con el PID Sintonizado

### 4.2.2.1 Modelamiento en tiempo discreto

Para poder realizar la sintonización basada en algoritmos genéticos se debe tomar en cuenta que se necesita trabajar en tiempo discreto así que su diagrama de bloques en lazo cerrado será el mostrado en la figura 4.3.



**Figura 4.3:** Diagrama de bloques del proceso real de control de flujo en tiempo discreto

Para este diagrama de bloques se tiene que los parámetros iniciales configurados en el DCS son los mostrados en la ecuación 4.1 y su función de transferencia es la mostrada en la ecuación 4.2, de esta ecuación se tiene que las constantes proporcional, integral y derivativa son las siguientes.

$$K_p = 0.5, K_I = 0.01667, K_D = 0.55 \quad (4.3)$$

además, se tiene que la función de transferencia en el dominio de la frecuencia  $G_p(s)$  es la mostrada en la ecuación 2.27, pero como se debe trabajar en tiempo discreto, es necesario transformar del dominio de “s” a “z” para lo cual se está considerando cómo tiempo de muestreo 1 segundo ya que este fue el que se utilizó para obtener los datos de la planta real, con esta información se obtiene la función de transferencia en tiempo discreto  $G_p(z)$  la cual se muestra en la ecuación 4.4.

$$G_p(z) = \frac{a_7z^7 + a_6z^6 + a_5z^5 + a_4z^4 + a_3z^3 + a_2z^2 + a_1z + a_0}{z^8 + b_7z^7 + b_6z^6 + b_5z^5 + b_4z^4 + b_3z^3 + b_2z^2 + b_1z + b_0} \quad (4.4)$$

Donde los valores de los coeficientes del numerador y denominador se pueden ver en la figura 4.4.

```
Gp =
-0.01677 z^7 + 0.04875 z^6 - 0.03159 z^5 - 0.02935 z^4 + 0.04214 z^3 - 0.013 z^2 - 0.0001832 z + 1.348e-08
-----
z^8 - 4.93 z^7 + 10 z^6 - 10.67 z^5 + 6.293 z^4 - 1.938 z^3 + 0.242 z^2 - 3.603e-05 z + 1.341e-09
Sample time: 1 seconds
Discrete-time transfer function.
```

**Figura 4.4:** Función de transferencia  $G_p(z)$  calculado en Matlab

#### 4.2.2.2 Sintonización del controlador PID

Ahora que ya se tiene la función de transferencia en tiempo discreto ya se puede realizar la sintonización, pero antes es necesario realizar algunas modificaciones en el programa.

- Especificaciones del proceso, se debe tomar en cuenta que cada lazo de control en una planta concentradora tiene sus propias especificaciones y es necesario que estos sean modificados, para este proceso se necesita considerar las siguientes especificaciones.
  - $T_s = 120 \text{ seg}$  → Rango de tiempo para encontrar el MP
  - $M_p < 3 \%$  → Máximo sobreimpulso
  - $e_{es} \approx 0$  → Error en estado estacionario
  - Respuesta → Sub amortiguada o sobre amortiguada

En la figura 4.5 se muestra la modificación que se ha realizado en el programa de acuerdo a las especificaciones de la respuesta transitoria.

```

%% Especificaciones del proceso

SP=200; % Cambio en el Set Point
T_s=120; % Rango de tiempo en el cual se encuentra el Máximo Sobreimpulso
          % Tiempo de muestreo T es de 1 segundos
          % Rango de tiempo escogido es de 120*1 = 120 segundos
MAX_SI=0.03*SP; %porcentaje de Sobre Impulso MAXIMO
MIN_SI=0.01*SP; %porcentaje de Sobre Impulso MINIMO

```

**Figura 4.5:** Especificaciones de la respuesta transitoria

- Primera generación, se ha escogido de acuerdo a la respuesta transitoria que se mostró en la figura 4.1 pero considerando que no se debe encontrar un valor alto de  $K_p$  ya que podría provocar un valor alto del  $M_p$  para la planta real. En la figura 4.6 se muestra la primera generación.

```

%% Primera Generación

Kp_i=C(1); % Kp inicial configurado
Ki_i=C(1)/C(2); % Ki inicial configurado
Kd_i=C(1)*C(3); % Kd inicial configurado

P(1,1:n)=1.5*Kp_i*rand(1,n); % Vector de valores Kp
P(2,1:n)=1*Ki_i*rand(1,n); % Vector de valores Ki
P(3,1:n)=1.5*Kd_i*rand(1,n); % Vector de valores Kd

P(1,1)=Kp_i;
P(2,1)=Ki_i;
P(3,1)=Kd_i;

```

**Figura 4.6:** Primera generación

- Descendencia, se ha considerado lo siguiente:
  - Crossover, va a buscar aleatoriamente valores menores que los padres de la generación anterior para asegurar que no se esté dejando de considerar a un mejor individuo.
  - Mutación, al igual que para la primera generación se ha escogido de acuerdo a la respuesta transitoria que se mostró en la figura 4.1

En la figura 4.7 se muestra la modificación que se ha realizado en el programa para la descendencia.

```
% Descendencia (Crossover y Mutacion)

% Crossover
for i=1:n/8
    Hx1(i)=0.25*Px(i)*rand(1);
    Hy1(i)=0.25*Py(i)*rand(1);
    Hz1(i)=0.25*Pz(i)*rand(1);
    Hx2(i)=0.50*Px(i)*rand(1);
    Hy2(i)=0.50*Py(i)*rand(1);
    Hz2(i)=0.50*Pz(i)*rand(1);
    Hx3(i)=1.00*Px(i)*rand(1);
    Hy3(i)=1.00*Py(i)*rand(1);
    Hz3(i)=1.00*Pz(i)*rand(1);
end

P1=Px; P2=Py; P3=Pz;

P1(n/8+1:(n/4))=Hx1;
P2(n/8+1:(n/4))=Hy1;
P3(n/8+1:(n/4))=Hz1;
P1(n/4+1:(3*n/8))=Hx2;
P2(n/4+1:(3*n/8))=Hy2;
P3(n/4+1:(3*n/8))=Hz2;
P1((3*n/8)+1:n/2)=Hx3;
P2((3*n/8)+1:n/2)=Hy3;
P3((3*n/8)+1:n/2)=Hz3;

% Mutacion
P1((n/2)+1:3*n/4)=1*rand(1,n/4);
P2((n/2)+1:3*n/4)=0.5*rand(1,n/4);
P3((n/2)+1:3*n/4)=1*rand(1,n/4);
P1((3*n/4)+1:n)=1.25*rand(1,n/4);
P2((3*n/4)+1:n)=0.5*rand(1,n/4);
P3((3*n/4)+1:n)=1.25*rand(1,n/4);

% Siguiete generacion
P(1,1:n)=P1;
P(2,1:n)=P2;
P(3,1:n)=P3;

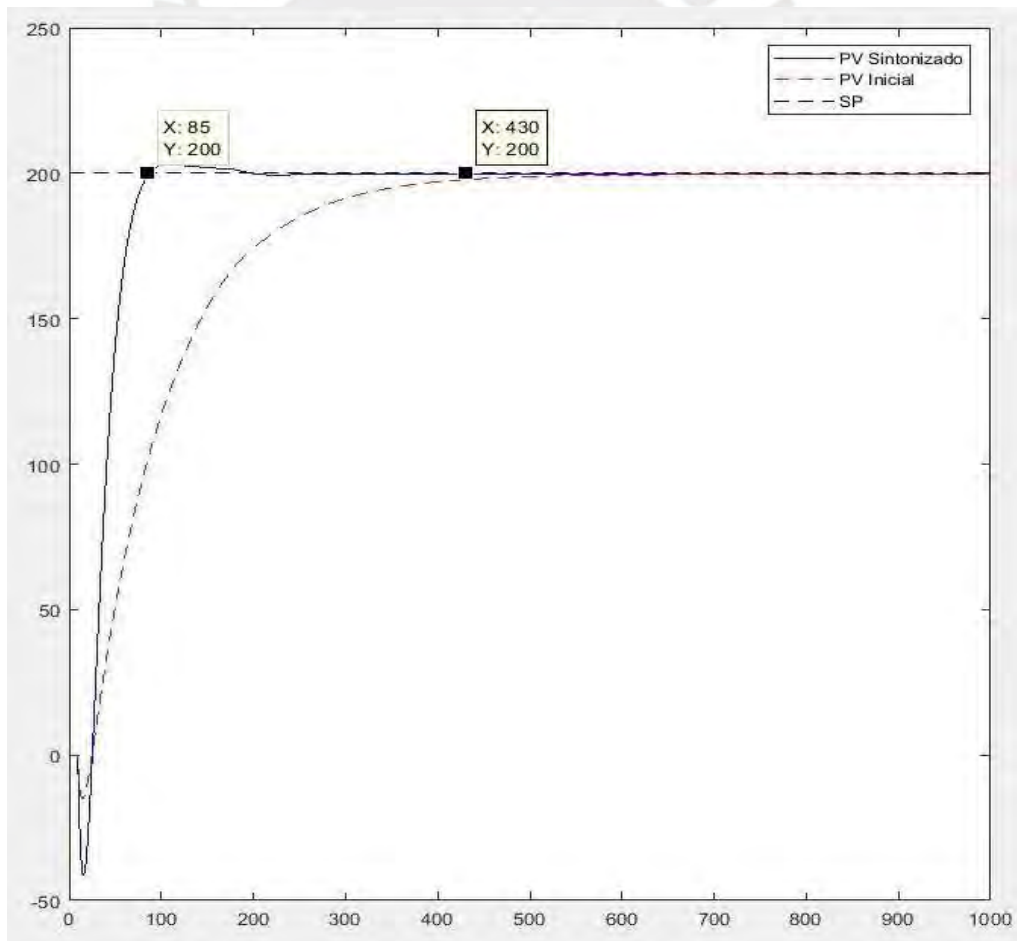
end
```

**Figura 4.7:** Descendencia (crossover y mutación)

Ya se tiene el programa terminado, este se ejecuta y se encuentra los siguientes parámetros para el controlador PID.

$$K_p = 1.2491, T_i = 26.7317, T_d = 0.8310 \quad (4.5)$$

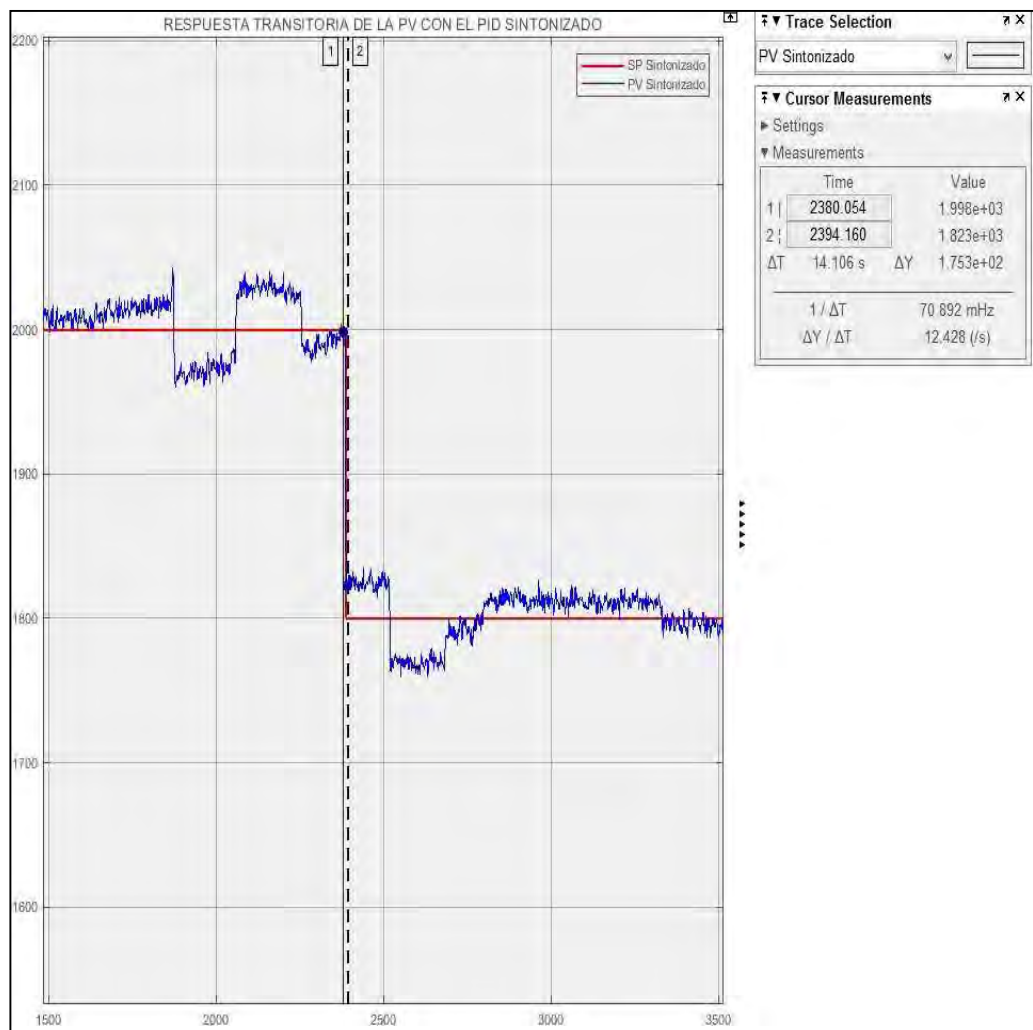
En la figura 4.8 se muestra la comparación de la PV sintonizada y la PV inicial, se aprecia claramente que la respuesta con los parámetros PID sintonizados es 345 segundos más rápida, pero se debe tomar en cuenta que esta respuesta es una simulación que se ha obtenido para un cambio de SP de 0 a 200 m<sup>3</sup>/h, para la planta real los valores de operación muestreados varían de 1600 a 2700 m<sup>3</sup>/h y en este rango se tienen tiempos de respuesta más rápidos.



**Figura 4.8:** Comparación de la PV Sintonizada y la PV inicial

### 4.2.2.3 Análisis de la repuesta transitoria

Los parámetros sintonizados del controlador PID que se muestran en la ecuación 4.5 son configurados en el DCS y se extraen 8100 datos para ser analizados con un tiempo de muestreo de 1 segundo. En la figura 4.9 se muestra la respuesta transitoria de la PV de 1500 a 3500 segundos.

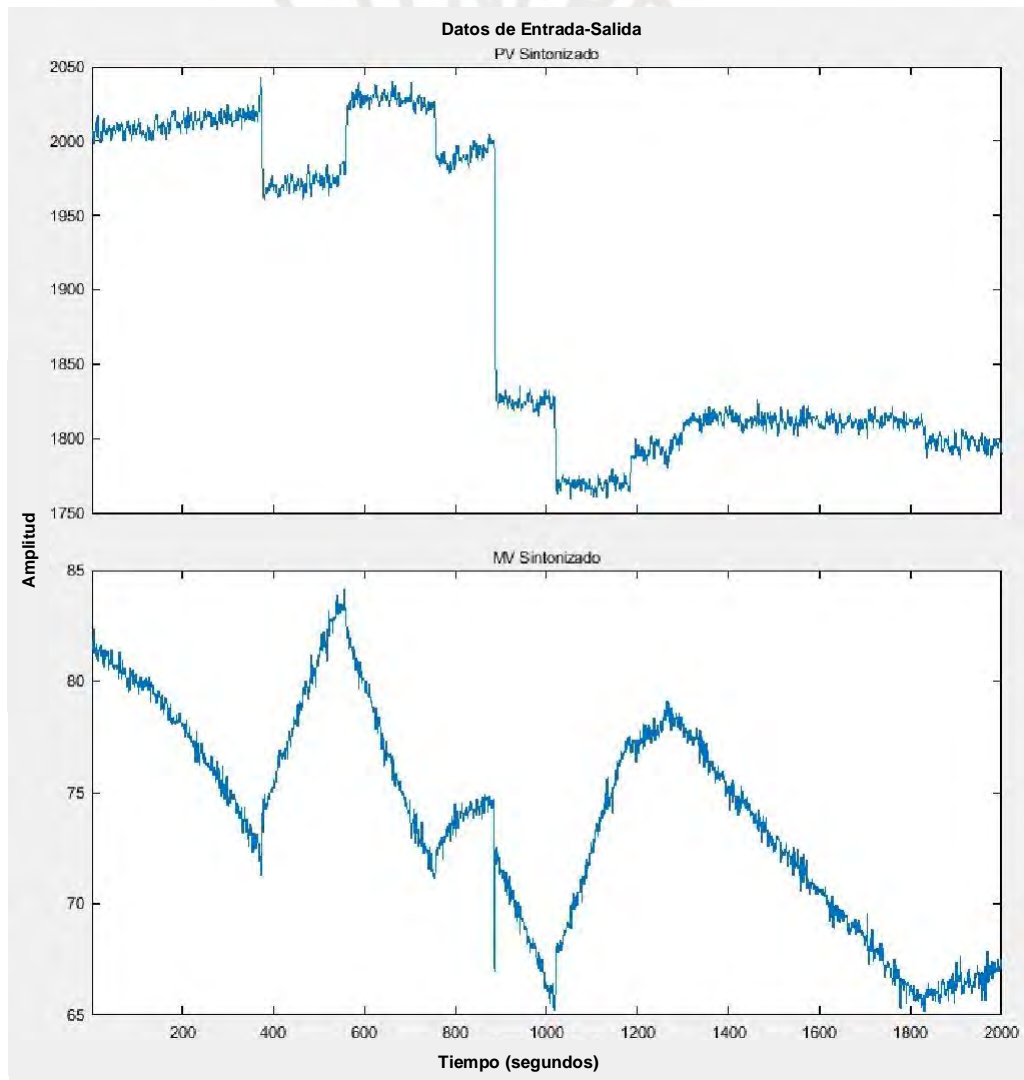


**Figura 4.9:** Respuesta de la PV con el PID sintonizado

Debido al comportamiento de la PV en este rango de tiempo, se considera que el tiempo de establecimiento se encuentra cuando se alcanza un error aproximado del 1.3%. Las especificaciones de la respuesta transitoria son las siguientes:

- $t_r = 14.106 \text{ seg}$  → Tiempo de subida
- $M_p = 0 \%$  → Máximo sobreimpulso
- $t_s = 14.106 \text{ seg}$  → Tiempo de establecimiento
- $e_{es} \approx 1.3\%$  → Error en estado estacionario
- Respuesta → Sobre amortiguada

En la figura 4.10 se muestra el comportamiento que tienen la PV con respecto a la MV donde también se aprecia claramente que los factores externos afectan el comportamiento de la PV y MV.



**Figura 4.10:** Comportamiento de la PV con respecto a la MV

En la tabla 4.1 se muestra la comparación entre los parámetros iniciales y los parámetros sintonizados del controlador PID.

Parámetros	Iniciales	Sintonizados
$K_p$	0.5	1.2491
$K_I$	0.0167	0.0467
$K_D$	0.55	1.0380

**Tabla 4.1:** Comparación entre los parámetros iniciales y los parámetros sintonizados del controlador PID

En la tabla 4.2 se compara las especificaciones de la respuesta transitoria que se han obtenido para los parámetros iniciales y los parámetros sintonizados del controlador PID.

Especificaciones	Iniciales	Sintonizados
$t_r(\text{seg})$	191.481	14.106
$M_p(\%)$	0	0
$t_s(\text{seg})$	191.481	14.106
$e_{ss}(\%)$	1	1.3
Respuesta	Sobre amortiguada	Sobre amortiguada
Cambio SP	2200 a 2000	2000 a 1800

**Tabla 4.2:** Comparación de las especificaciones de la respuesta transitoria

Se puede apreciar en la tabla 4.1 que se ha tenido un aumento en la acción proporcional, integrativa y derivativa lo que ha permitido obtener una respuesta 2.95 minutos más rápida y además esta respuesta aun es sobre amortiguada como se puede ver en la tabla 4.2. Este cambio es una mejora significativa en la respuesta transitoria de la PV y aunque el  $e_{ss}$  varia hasta un máximo de 1.3% esta respuesta es la que se requería para este proceso de control de flujo por lo que no es necesario realizar una nueva sintonización.

### 4.3 Resultado de la sintonización en un proceso real de control de nivel

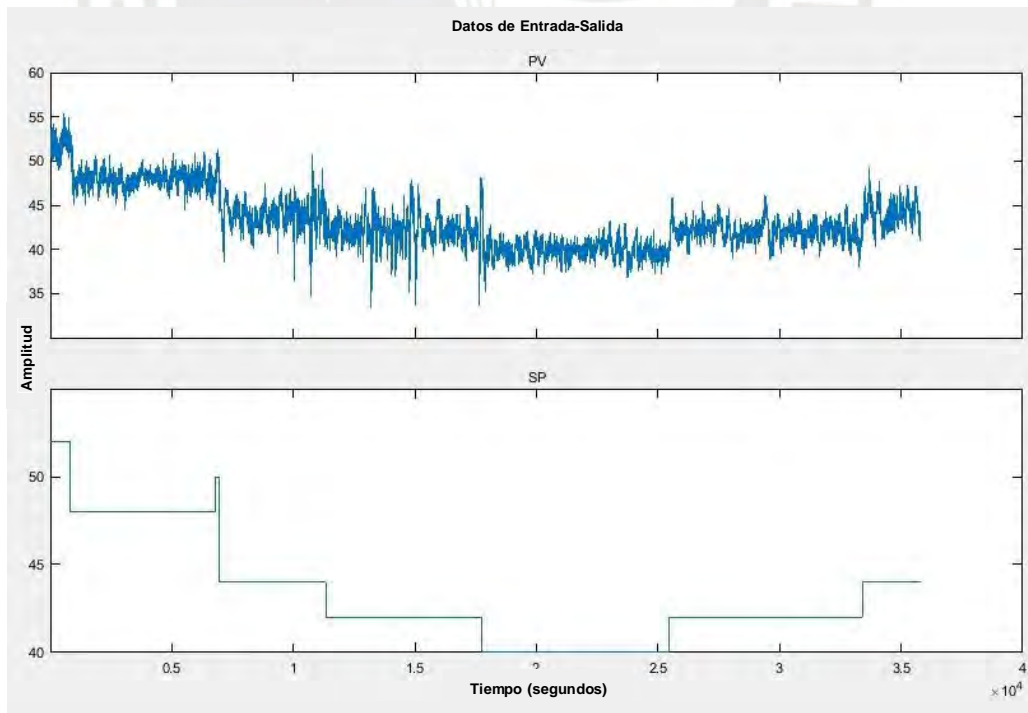
#### 4.3.1 Modelamiento del proceso real de control de nivel

Para poder realizar la sintonización es necesario primero encontrar el modelo del proceso de control de nivel de espuma de la celda de flotación Rougher en lazo cerrado, para lograrlo se usa el método que se realizó para el proceso real de control de flujo.

Para este proceso también se conoce el tipo de controlador PID y sus parámetros  $K_p$ ,  $T_i$  y  $T_d$ , su función de transferencia se muestra en la ecuación 4.6.

$$PID(s) = 0.5 \left( 1 + \frac{1}{55} * \frac{1}{s} + s \right) \quad (4.6)$$

De la base de datos se obtiene la variación de la PV con respecto al SP en un rango de 36000 segundos, este se muestra en la figura 4.11.



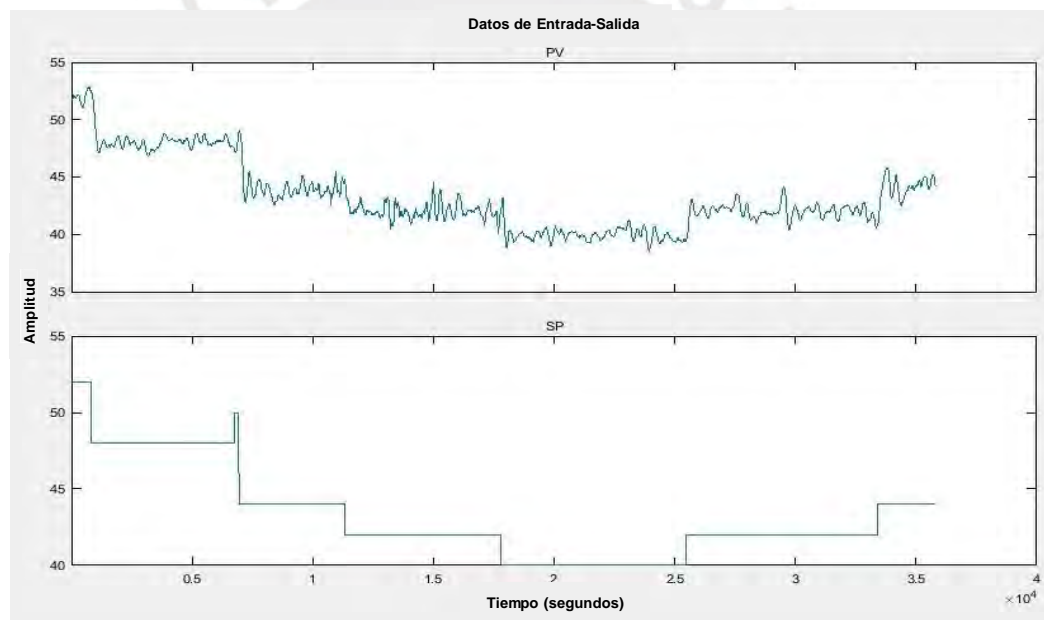
**Figura 4.11:** Variación de la PV con respecto al SP

Como se puede observar la PV tiene mucho ruido y esto se tiene en todas las celdas de flotación, por tal motivo es necesario trabajar el modelamiento y la sintonización con la PV filtrada. En la figura 4.12 se muestra cómo se realiza el filtro.

```
% Filtro de PV = Datos(:,3)
Tamano = 200;
b = (1/Tamano)*ones(1,Tamano);
a = 1;
PV_fil = filter(b,a,Datos(:,3));
```

**Figura 4.12:** Método para filtrar la PV

En la figura 4.13 se muestra la variación de la PV filtrada con respecto al SP, debido al filtro se tomará un rango de 35800 segundos.



**Figura 4.13:** Variación de la PV filtrada con respecto al SP

Debido al alto nivel de ruido en la PV se ha obtenido la información para un mayor rango de tiempo, de esta manera se tendrá una mejor estimación del modelo del proceso, ahora se calcula la función de transferencia de dicho modelo tal como se realizó con el proceso de control de flujo en la subsección 2.3.3. Para este proceso de control de nivel se ha obtenido una mejor estimación con 2 ceros y 3 polos, en la ecuación 4.7 se muestra la función de transferencia  $G_p(s)$  que se identifica.

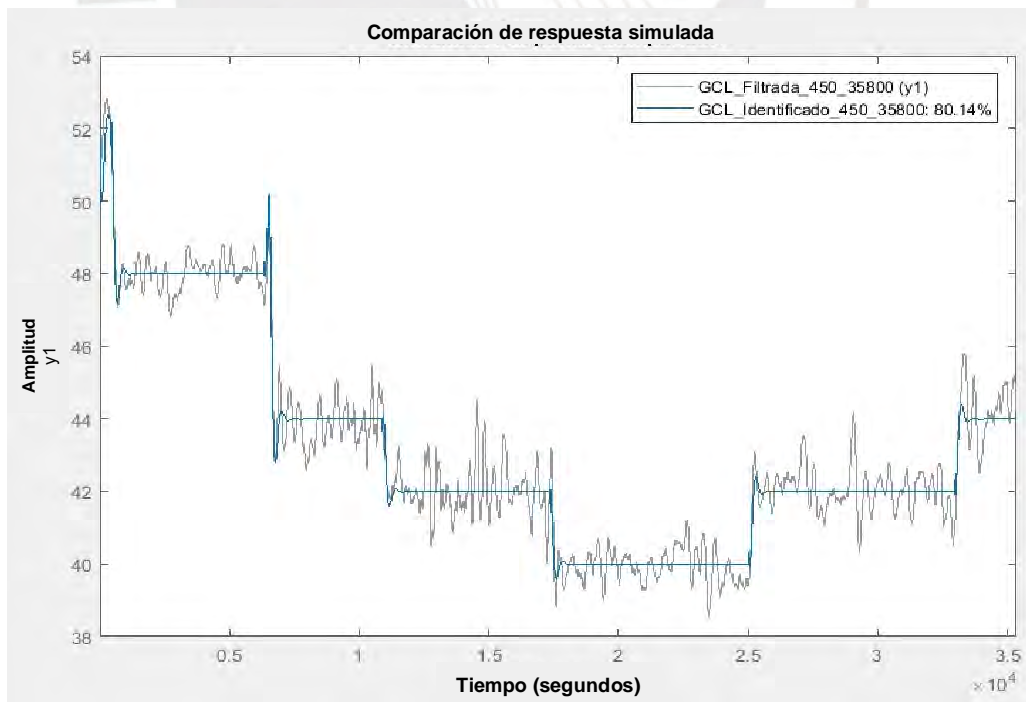
$$G_p(s) = \frac{a_6s^6 + a_5s^5 + a_4s^4 + a_3s^3 + a_2s^2 + a_1s}{s^8 + b_7s^7 + b_6s^6 + b_5s^5 + b_4s^4 + b_3s^3 + b_2s^2 + b_1s + b_0} \quad (4.7)$$

Donde los valores de los coeficientes del numerador y denominador se pueden ver en la figura 4.14.

```
Gp =
From input to output "y1":
    0.7134 s^6 + 2.817 s^5 - 0.1032 s^4 + 0.002113 s^3 + 1.66e-05 s^2 + 7.511e-07 s
-----
0.5 s^8 + 4.142 s^7 + 10.29 s^6 + 6.979 s^5 + 0.397 s^4 + 0.008913 s^3 + 0.0001107 s^2 + 7.159e-07 s + 1.356e-12
Continuous-time transfer function.
```

**Figura 4.14:** Función de transferencia  $G_p(s)$

En la figura 4.15 se muestra que se obtiene una estimación de los datos del 80.14% al comparar la PV filtrada (ver figura 4.13) con los datos de la PV identificada que se obtiene al simular la respuesta de la función de transferencia  $G_p(s)$ .



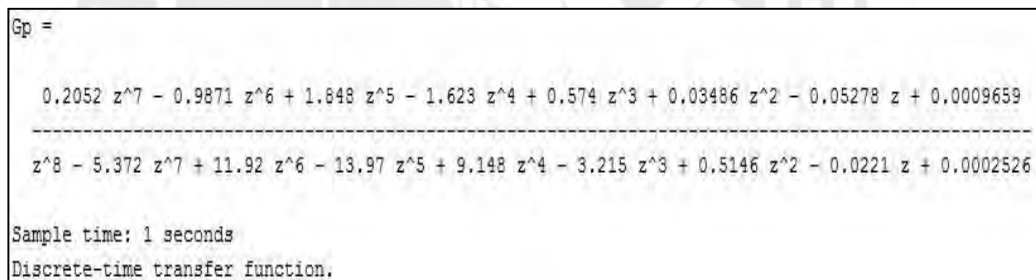
**Figura 4.15:** Comparación de la PV filtrada con la PV identificada

La estimación de los datos del 80.14% es un valor alto considerando el nivel de ruido de la PV por lo que este modelo se puede usar para la sintonización basada en algoritmos genéticos.

Como ya se indicó anteriormente se debe trabajar en tiempo discreto para realizar la sintonización, por lo que es necesario transformar del dominio de “s” a “z” para lo cual se está considerando cómo tiempo de muestreo 1 segundo ya que este fue el que utilizó para obtener los datos de la planta real, la función de transferencia en tiempo discreto  $G_p(z)$  se muestra en la ecuación 4.8.

$$G_p(z) = \frac{a_7z^7 + a_6z^6 + a_5z^5 + a_4z^4 + a_3z^3 + a_2z^2 + a_1z + a_0}{z^8 + b_7z^7 + b_6z^6 + b_5z^5 + b_4z^4 + b_3z^3 + b_2z^2 + b_1z + b_0} \quad (4.8)$$

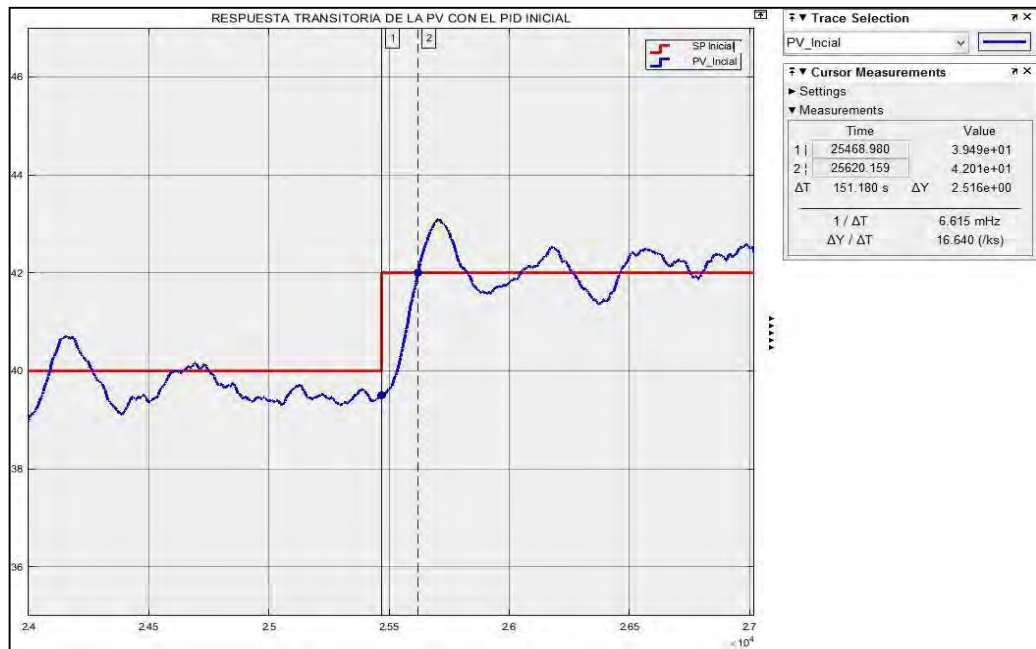
Donde los valores de los coeficientes del numerador y denominador se pueden ver en la figura 4.16.



**Figura 4.16:** Función de transferencia  $G_p(z)$

### 4.3.2 Respuesta transitoria de la PV con el PID inicial

Ahora que ya se cuenta con el modelo del proceso en tiempo discreto es posible realizar la sintonización del controlador PID, pero antes es necesario analizar la respuesta transitoria de la PV que se obtiene con el PID inicial la cual se mostró en la figura 4.14, debido a que se tiene varios cambios de SP se va a analizar solo uno de estos cambios, se ha elegido el cambio que se encuentra entre los 24000 y 27000 segundos. En la figura 4.17 se muestra la respuesta transitoria en ese rango de tiempo.



**Figura 4.17:** Respuesta de la PV filtrada que se obtiene con el PID inicial

Debido al comportamiento de la PV en este rango de tiempo, se considera que el tiempo de establecimiento se encuentra cuando se alcanza un error aproximado del 1.1%, también se observa que se trata de un proceso lento. Las especificaciones de la respuesta transitoria son las siguientes:

- $t_r = 151.180 \text{ seg}$  → Tiempo de subida
- $M_p = 2.619 \%$  → Máximo sobreimpulso
- $t_s = 436.542 \text{ seg}$  → Tiempo de establecimiento
- $e_{es} \approx 1.1\%$  → Error en estado estacionario
- Respuesta → Sub amortiguada

### 4.3.3 Sintonización del proceso real de control de nivel

Como ya se indicó en la subsección 4.2.2.2 es necesario establecer cuáles son las especificaciones de la respuesta transitoria que se necesita cumplir para poder realizar la sintonización, para este proceso se necesita considerar las siguientes especificaciones.

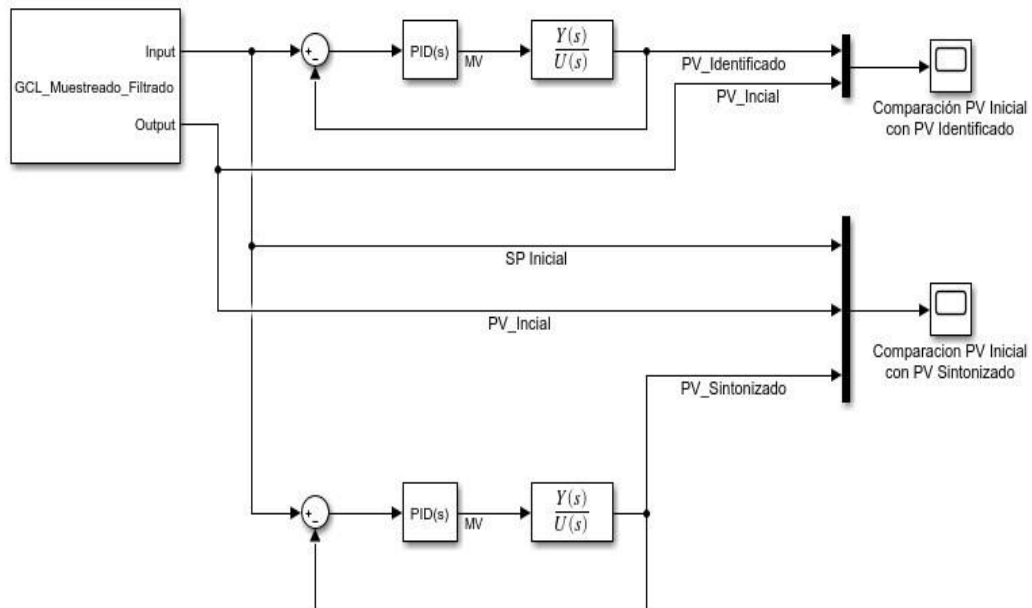
- $T_s = 160 \text{ seg}$  → Rango de tiempo para encontrar el MP
- $M_p < 2 \%$  → Máximo sobreimpulso
- $e_{es} \approx 1 \%$  → Error en estado estacionario
- Respuesta → Sub amortiguada

Para la primera generación y la descendencia se va a considerar lo mismo que lo mostrado en las figuras 4.6 y 4.7 respectivamente. Al ejecutar el programa se encuentra los siguientes parámetros sintonizados para el controlador PID.

$$K_p = 1.1199, T_i = 129.5448, T_d = 0.5250 \quad (4.9)$$

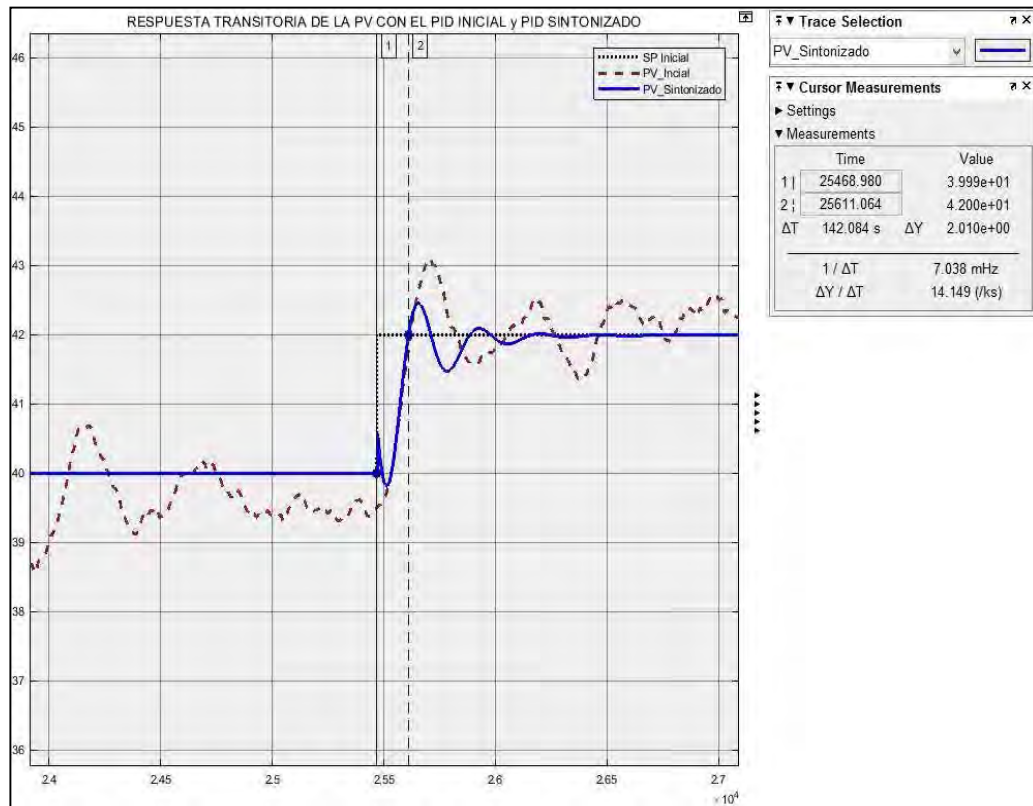
#### 4.3.4 Análisis de la repuesta transitoria

Debido a que para este proceso no fue posible modificar los parámetros del controlador PID en el DCS se ha tenido que elaborar un modelo en Simulink para poder analizar la respuesta transitoria de la PV obtenida con el PID sintonizado con respecto a la PV obtenida con el PID inicial, en la figura 4.18 se muestra el modelo elaborado.



**Figura 4.18:** Modelo en Simulink para comparación de la PV

En la figura 4.19 se muestra la respuesta transitoria de la PV obtenida con el PID sintonizado en un rango de tiempo de 24000 a 27000 segundos para poder ser comparada con la PV inicial.



**Figura 4.19:** Simulación de la respuesta de la PV obtenida con el PID sintonizado

Las especificaciones de la respuesta transitoria son las siguientes:

- $t_r = 142.113 \text{ seg}$  → Tiempo de subida
- $M_p = 1.071 \%$  → Máximo sobreimpulso
- $t_s = 350.098 \text{ seg}$  → Tiempo de establecimiento
- $e_{es} \approx 1\%$  → Error en estado estacionario
- Respuesta → Sub amortiguada

En la tabla 4.3 se muestra la comparación entre los parámetros iniciales y los parámetros sintonizados del controlador PID.

Parámetros	Iniciales	Sintonizados
$K_p$	0.5	1.1464
$K_I$	0.0091	0.0080
$K_D$	0.5	0.4358

**Tabla 4.3:** Comparación entre los parámetros iniciales y los parámetros sintonizados

En la tabla 4.4 se compara las especificaciones de la respuesta transitoria que se han obtenido para los parámetros iniciales y los parámetros sintonizados del PID.

Especificaciones	Iniciales	Sintonizados
$t_r(\text{seg})$	151.180	142.113
$M_p(\%)$	2.619	1.071
$t_s(\text{seg})$	436.542	350.098
$e_{ss}(\%)$	1.1	1
Respuesta	Sub amortiguada	Sub amortiguada
Cambio SP	40 a 42	40 a 42

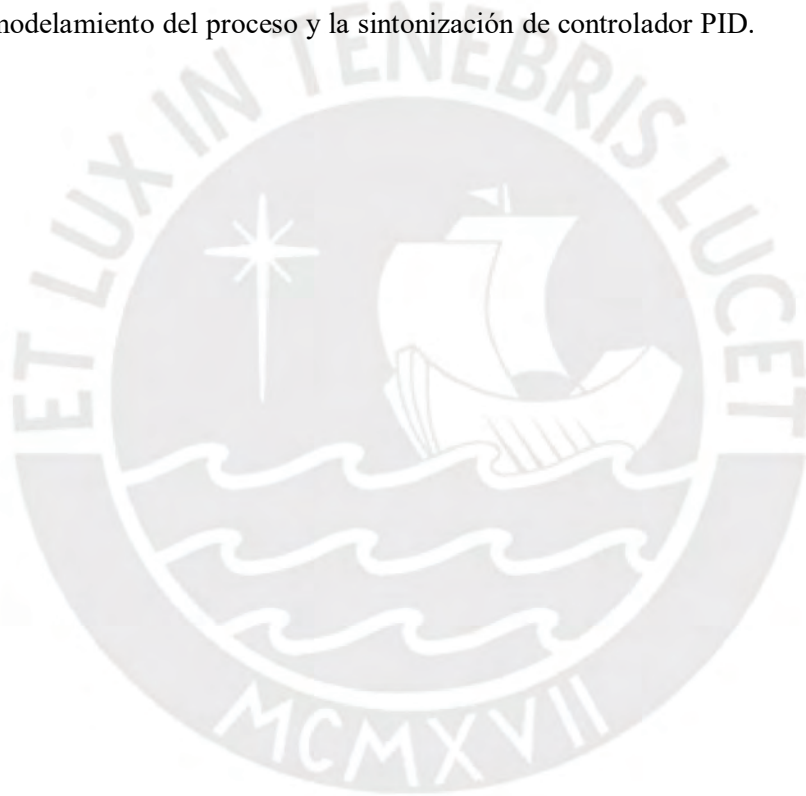
**Tabla 4.4:** Comparación de las especificaciones de la respuesta transitoria

Se puede apreciar en la tabla 4.3 que se ha tenido un aumento en la acción proporcional pero una disminución en la acción integrativa y derivativa, esto ha permitido que se obtenga una respuesta transitoria con un tiempo de subida menor por 9 segundos y un tiempo de establecimiento menor por 86.4 segundos, además también se ha obtenido un menor sobreimpulso. Este cambio en la sintonización mejora la respuesta de la PV.

#### 4.4 Conclusiones del capítulo

- La sintonización basada en algoritmos genéticos permitió mejorar la respuesta de la PV para el lazo de control de flujo del proceso de inyección de agua cruda al cajón de alimentación de las bombas de ciclones, esto fue probado tanto en simulación como en la planta real y ha permitido garantizar el porcentaje de sólidos y la adecuada clasificación en los hidrociclones.

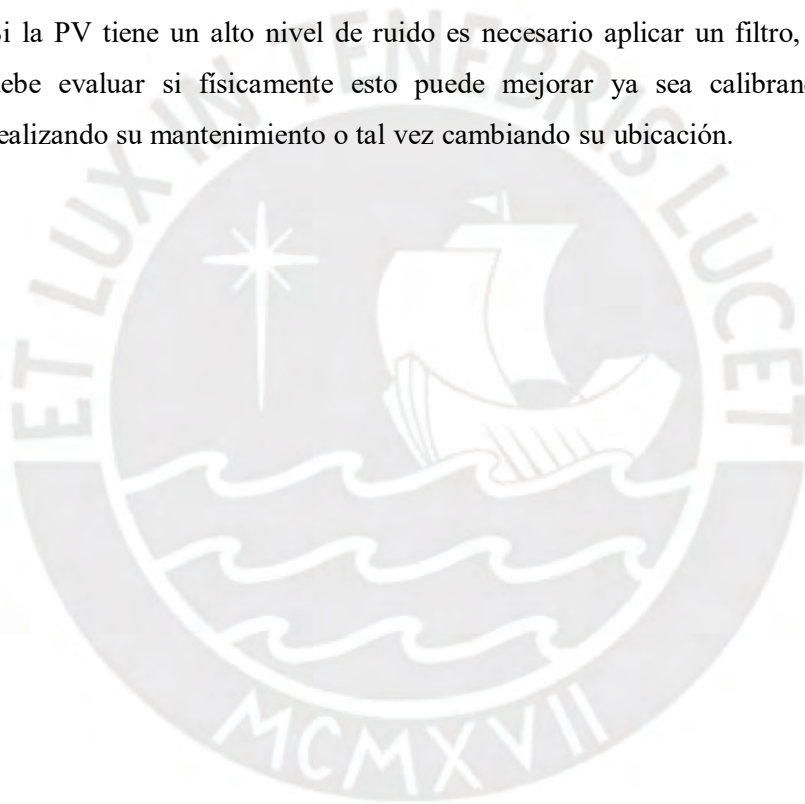
- La sintonización basada en algoritmos genéticos permitió mejorar la respuesta de la PV para el lazo de control de nivel de espuma de una de las celdas de flotación Rougher, esto fue probado solo en simulación.
- Para el proceso de control de nivel no se ha contado con la información de la MV, pero esta información no ha sido necesaria para realizar el modelamiento del proceso y la sintonización del controlador PID.
- Si la señal de la PV tiene mucho ruido es necesario filtrarla para realizar el modelamiento del proceso y la sintonización de controlador PID.



## CONCLUSIONES GENERALES

- El método de “identificación de proceso con controlador conocido” permite una buena estimación de los datos, esto ha sido comprobado con el proceso teórico de control de flujo, el proceso real de control de flujo y el proceso real de control de nivel.
- Con la sintonización basada en algoritmos genéticos se puede encontrar distintos valores para los parámetros del controlador PID, todos los resultados obtenidos permiten encontrar una respuesta transitoria que cumple con las especificaciones del proceso, estos parámetros deben ser evaluados por el ingeniero de control antes de ser configurados en el DCS o PLC.
- La sintonización basada en algoritmos genéticos puede ser un método bien aceptado por los ingenieros de control ya que va a requerir de su conocimiento y experiencia en el proceso para escoger las adecuadas especificaciones de la respuesta transitoria, así como también los rangos de solución para escoger una adecuada descendencia. De esta manera se encontrarán parámetros de sintonización coherentes que mejoren el proceso.
- Para el lazo que controla la adición agua del “cajón de alimentación de las bombas de ciclones” se ha logrado reducir considerablemente el tiempo de establecimiento de 191 segundos a 14 segundos lo que permite tener una adición correcta de agua para mantener un porcentaje de solidos entre 25% y 30% en el Overflow de los hidrociclones.
- Para el lazo que controla el nivel de espuma de la “primera celda de flotación Rougher” se ha logrado reducir el sobre pico de 2.619% a 1.071% y el tiempo de establecimiento de 436.542 segundos a 350.098 segundos lo que permite mantener más tiempo el nivel en el rango que el ingeniero metalúrgico a calculado para evitar un tiempo de permanencia bajo o alto en la celda de flotación, de esta manera se obtiene una mayor recuperación de cobre y un concentrado limpio.

- Se ha comprobado que la sintonización basada en algoritmos genéticos permite mejorar la respuesta transitoria para el proceso teórico de control de flujo, el proceso real de control de flujo y el proceso real de control de nivel.
- No se necesita la información de la MV para realizar el método de “identificación del proceso con controlador conocido” o para la “sintonización basada en algoritmos genéticos”, pero esta información es muy importante para poder evaluar si la acción de control del PID genera cambios no deseados en el actuador.
- Si la PV tiene un alto nivel de ruido es necesario aplicar un filtro, pero antes se debe evaluar si físicamente esto puede mejorar ya sea calibrando el sensor, realizando su mantenimiento o tal vez cambiando su ubicación.



## BIBLIOGRAFIA

1. Ahmed Alkamachi. Modelling and Genetic Algorithm Based-PID Control of H-Shaped Tacing Quadcopter. King Fahhd University of Petroleum & Minerals, 2017.
2. Aidan O'Dowyer, HandBook of PI and PID Controller Tuning Rules 3rd Edition. Imperial College Press, 2009.
3. Arun K. Tangirala. Principles of System Identification Theory and Practice. CRC Press Taylor & Francis Group, 2015.
4. Biao Huang and Sirish L. Shah. Performance Assessment of Control Loops theory and Applications. Springer- Verlag London Ltd, 1999.
5. Cheng-Ching Yu. Autotuning of PID Controllers. Springer- Verlag London, 2006.
6. Gabriel Luque and Enrique Alba. Parallel Genetic Algorithms Theory and Real World Applications. Springer- Verlag Berlin, 2011.
7. Glover F and G. A. Kochenberger. Hanbook of Metaheuristics. Kluwer Academic Publishers, 2003.
8. Grzegorz Mzyk. Combinedn Parametric-Nonparametric Identification of Block-Oriented Systems. Springer International Publishing Switzerland, 2014.
9. Fulu Cao. PID Controller Optimized by Genetic Algorithm for Direct-Drive Servo System. The Natural Computing Application Forum, 2018.
10. Igor Boiko. Non-parametric Tunning of PID Controllers. Springer – Verlag London, 2013.

11. Jacob Fraden. Handbook of Modern Sensors. Springer International Publishing Switzerland, 2016.
12. Jan Erik Mont Gomery Pinto. Aplicación Práctica de Método de Sintonía de Controladores PID Utilizando Método Relé con Histeriesisl. Programa Post Grado Universidad Federal de Rio Grande Del Norte, 2014.
13. Jhon Norton. Identification of Parametric Models from Experimental Data. Springer Masson, 1997.
14. Jun Yi. Optimization of PID Control System Based on Improved Genetic Algorithm. Academic Journal of Manufacturing Engineering, 2018.
15. Karl J. Astrom and Tore Hagglund. Advanced PID Control. ISA- Instrumentation, Systems, and Automation Society, 2006.
16. Katsuhiko Ogata. Ingeniería de Control Moderna Quinta Edición. Pearson Educación S.A., 2010.
17. Konstantinos G. Papadopoulos. PID Controller Tunning Using the Magnitude Optimum Criterion. Springer International Publishing Switzerland, 2015.
18. Lane Desborough and Thomas Harris. Performance Assessment Measures for Univariate Feedback Control. Department of Chemical Engineering, Queen's University, Kingston, Ontario K7L 3N6, 1992.
19. Lei Jinli. Adaptative Fuzzy PID Control For Boiler Deaerator. International Conference on Industrial Control and Electronics Engineering, 2012.
20. Liuping Wang and Hugues Garnier. System Identification, Environmental Modelling, and Control System Design. Springer-Verlag London, 2012.

21. Michael A. Johnson and Mohammad H. Moradi. PID Control New Identification and Design Methods. Springer-Verlag London, 2005.
22. Mingjian Zhou, Yuqin Wang and Haibing Wu. Control Design of the Wave Compensation System based on the Genetic PID Algorithm. Dimitrios E. Manolakos, 2019.
23. Mohiedine Jelali. Control Performance Management in Industrial Automation. Springer-Verlag London, 2013.
24. Muhammet Unal, Ayca Ak, Vedat Topuz and Hasan Erdal. Optimization of PID Controllers Using Ant Colony and Genetic Algorithms. Springer-Verlag Berlin Heidelberg, 2013.
25. Rodrigo Juliani Correa de Godoy. Sintonía Óptima de Controladores. Escuela Politécnica de Universidad de San Paulo, 2012.
26. Salah Eddine, Said Sadki, Bahloul Bensassi. System Identification of a DC Servo Motor Using ARX and ARMAX Model. IEEE, 2019.
27. Salvador Ortiz Santos. Tesis Sintonización de un Controlador PID basado en un algoritmo heurístico para el control de Ball and Beam. Universidad Autónoma de Querétaro, 2014.
28. Shaofei Wu. Study on an Improved Algorithm for Optimization of PID Parameters. Wuhan Institute of Technology, 2016.
29. Tan Kok Kiong, Wang Qing-Guo, Hang Chang Chied. Advances in PID Control. Springer-Verlag London Berlin Heidelberg, 1999.
30. V. Bobál, J. Bohm, J. Fressl, J. Machacek. Digital Self-Tuning Controllers. Springer-Verlag London, 2005.

## ANEXOS

### Anexo A: Programas

#### A.1. Modelamiento y Sintonización

```
%% Modelo Matemático del Proceso teórico de flujo

clear; clc; close all;

% Variables

load('Datos_Modelo_Teorico_Flujo.mat');
T = 0.1; % Tiempo de muestreo
Tm = 200; % Rango de tiempo muestreado
SP = Datos_Modelo_Teorico_Flujo(:,1); % Set point
PV = Datos_Modelo_Teorico_Flujo(:,3); % Variable de Proceso
Ceros = 2;
Polos = 3;

% Identificación del sistema en lazo cerrado

GCL_Muestreado = iddata(PV,SP,T);
GCL_Identificado = tfest(GCL_Muestreado,Polos,Ceros);

% PID configurado en controlador C=[Kp Ti Td]%

C = [2 2/1 1/2];
PID = tf([C(1)*C(3) C(1) C(1)/C(2)], [1 0]);

% Identificación del proceso

G_PID_P = GCL_Identificado/(1-GCL_Identificado);
Gp = G_PID_P/PID; % Proceso identificado

% Numerador y denominador de la Función de transferencia de Gp(s)
y Gp(z)

% Sistema continuo
[nums_Gp,dens_Gp] = tfdata(Gp,'v');

% Sistema discreto
[numz_Gp,denz_Gp] = c2dm(nums_Gp,dens_Gp,T,'zoh');

% numz_Gp =
%      0      0.0100   -0.0359    0.0464   -0.0224   -0.0009
0.0034   -0.0005    0.0000
% denz_Gp =
%      1.0000   -5.5352   12.7604  -15.6835   10.8417   -3.9995
0.6173   -0.0011    0.0000
```

**%% Especificaciones del proceso**

```
SP=50; % Cambio en el Set Point
T_s=60; % Rango de tiempo en el cual se encuentra el Máximo
        Sobreimpulso
        % Tiempo de muestreo T es de 0.1 segundos
        % Rango de tiempo escogido es de 50*0.1 = 5 segundos
MAX_SI=0.05*SP; %porcentaje de Sobre Impulso MAXIMO
MIN_SI=0.02*SP; %porcentaje de Sobre Impulso MINIMO
```

**%% Menu para selección del error integral para la función de costo**

```
w=menu('Seleccione error integral','ISE','IAE','ITSE','ITAE');
```

```
if w==1
    op=1;
end
if w==2
    op=2;
end
if w==3
    op=3;
end
if w==4
    op=4;
end
```

**%% Parametros del Algoritmo Genetico**

```
% Vector P=[Kp; Ki; Kd]
% Dimensión del vector P es 3 x n
```

```
n=42000; % Tamano de la 2oblación
n_ite=10; % Numero de generaciones o iteracciones
```

**%% Primera Generación**

```
Kp_i=C(1); % Kp inicial configurado
Ki_i=C(1)/C(2); % Ki inicial configurado
Kd_i=C(1)*C(3); % Kd inicial configurado
```

```
P(1,1:n)=2*Kp_i*rand(1,n); % Vector de valores Kp
P(2,1:n)=2*Ki_i*rand(1,n); % Vector de valores Ki
P(3,1:n)=2*Kd_i*rand(1,n); % Vector de valores Kd
```

```
P(1,1)=Kp_i;
P(2,1)=Ki_i;
P(3,1)=Kd_i;
```

```

%% Algoritmo Genético

% Evaluación por generación

for ite=1:n_ite %n_ite=5

    % Evaluación de toda la población

    for i=1:n

        aptitud_actual(i)=Funcion_Aptitud(P(:,i),numz_Gp,denz_Gp...
            ,MAX_SI,MIN_SI,op,SP,T,Tm,T_s);

    end

    % Ordenamiento ascendente de la función de aptitud

    [FA , ind]=sort(aptitud_actual);

    % Ordenamiento de vector Px, Py, Pz, donde
    % [Kp Ki Kd]=[Px(1) Py(1) Pz(1)] es el mejor individuo de la
    Generación

    for i=1:n
        Px(i)=P(1,ind(i));
        Py(i)=P(2,ind(i));
        Pz(i)=P(3,ind(i));
    end

    % Descendencia (Crossover y Mutacion)

    % Crossover

    for i=1:n/8
        Hx1(i)=0.25*Px(i)*rand(1);
        Hy1(i)=0.25*Py(i)*rand(1);
        Hz1(i)=0.25*Pz(i)*rand(1);
        Hx2(i)=0.75*Px(i)*rand(1);
        Hy2(i)=0.75*Py(i)*rand(1);
        Hz2(i)=0.75*Pz(i)*rand(1);
        Hx3(i)=1.25*Px(i)*rand(1);
        Hy3(i)=1.25*Py(i)*rand(1);
        Hz3(i)=1.25*Pz(i)*rand(1);
    end

    P1=Px; P2=Py; P3=Pz;

    P1(n/8+1:(n/4))=Hx1;
    P2(n/8+1:(n/4))=Hy1;
    P3(n/8+1:(n/4))=Hz1;
    P1(n/4+1:(3*n/8))=Hx2;
    P2(n/4+1:(3*n/8))=Hy2;
    P3(n/4+1:(3*n/8))=Hz2;
    P1((3*n/8)+1:n/2)=Hx3;
    P2((3*n/8)+1:n/2)=Hy3;
    P3((3*n/8)+1:n/2)=Hz3;

```

```
% Mutacion
```

```
P1((n/2)+1:3*n/4)=1.5*rand(1,n/4);  
P2((n/2)+1:3*n/4)=1.5*rand(1,n/4);  
P3((n/2)+1:3*n/4)=1.5*rand(1,n/4);  
P1((3*n/4)+1:n)=2.5*rand(1,n/4);  
P2((3*n/4)+1:n)=2.5*rand(1,n/4);  
P3((3*n/4)+1:n)=2.5*rand(1,n/4);
```

```
% Siguiete generaci3n
```

```
P(1,1:n)=P1;  
P(2,1:n)=P2;  
P(3,1:n)=P3;
```

```
end
```

```
%% Parametros del PID sintonizado
```

```
Kp=P1(1);  
Ki=P2(1); Ti=Kp/Ki;  
Kd=P3(1); Td=Kd/Kp;
```

```
%% Resultados
```

```
Kp  
Ki  
Kd
```

## A.2. Funci3n de aptitud

```
%% funci3n
```

```
FA=Funcion_Aptitud(k,numz,denz,MAX_SI,MIN_SI,op,SP,T,Tm,T_s)
```

```
%% Par3metros Iniciales
```

```
tm=Tm; % Cantidad de muestras  
y=zeros(tm,1); % y-> salida proceso  
u=zeros(tm,1); % u-> salida PID  
e=zeros(tm,1); % Error SP - y  
ISE=0;  
IAE=0;  
ITAE=0;  
ITSE=0;  
ts=T; % Tiempo de muestreo ingresado en funci3n principal  
ref=SP; % Set Point  
fc=zeros(1,tm); %Salida de la funci3n  
Kp=k(1); % Vector de Kp  
Ki=k(2); % Vector de Ki  
Kd=k(3); % Vector de Kd  
integral=0;
```

```

%% Cálculo en discreto

for t=length(numz):tm

    % Respuesta y error del modelo teórico en lazo cerrado

    y(t)=-denz(2)*y(t-1)-denz(3)*y(t-2)-denz(4)*y(t-3)-
denz(5)*y(t-4)...
        -denz(6)*y(t-5)-denz(7)*y(t-6)-denz(8)*y(t-7)-
denz(9)*y(t-8)...
        +numz(9)*u(t-8)+numz(8)*u(t-7)+numz(7)*u(t-
6)+numz(6)*u(t-5)...
        +numz(5)*u(t-4)+numz(4)*u(t-3)+numz(3)*u(t-
2)+numz(2)*u(t-1)...
        +numz(1)*u(t);
    e(t)=ref-y(t);

    % Salida de PID

    integral=integral+Ki*e(t)*ts;
    u(t)=Kp*e(t) + integral + Kd*(e(t)-e(t-1))*1/(ts) + p(t) ;

%PID

    % Cálculo del error integral

    if op==1
        e1(t)=e(t)*e(t);
        ISE=ISE+e1(t);
    end

    if op==2
        e2(t)=abs(e(t));
        IAE=IAE+e2(t);
    end

    if op==3
        e3(t)=t*(e(t)*e(t));
        ITSE=ITSE+e3(t);
    end

    if op==4
        e4(t)=t*abs(e(t));
        ITAE=ITAE+e4(t);
    end

end
end

```

**%% Máximo sobreimpulso (Mp)**

```
Y_s=y(1:T_s); % PV en el rango de tempo T_s
SI=max(Y_s)-ref; % ref es el SP
```

```
if MAX_SI>SI && SI>MIN_SI
```

```
    Mp=100*SI/ref;
```

```
else
```

```
    Mp=1e10;
```

```
    % Valor grande para desestimar
```

```
end
```

**%% Resultado de Función de aptitud**

```
fc_1=ISE/100; %+10*Mp
```

```
fc_2=IAE/10+10*Mp; % fc_2: Función de aptitud para IAE
```

```
    % IAE: Error Absoluto Integral
```

```
    % Mp: Máximo sobreimpulso
```

```
    % 1/10 y 10 son los pesos
```

```
fc_3=ITSE/10000; %+10*Mp
```

```
fc_4=ITAE/1000; %+10*Mp
```

```
if op==1
```

```
    FA=fc_1;
```

```
end
```

```
if op==2
```

```
    FA=fc_2;
```

```
end
```

```
if op==3
```

```
    FA=fc_3;
```

```
end
```

```
if op==4
```

```
    FA=fc_4;
```

```
end
```

