

**UNIVERSIDAD NACIONAL DE INGENIERÍA**

**FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA**



**DISEÑO DEL SISTEMA DE CONTROL PID PARA UN  
EVAPORADOR DE LECHE**

**INFORME DE SUFICIENCIA**

**PARA OPTAR EL TÍTULO PROFESIONAL DE:**

**INGENIERO ELECTRONICO**

**PRESENTADO POR:**

**SIMÓN ANGEL CACCIRE MELGAREJO**

**PROMOCIÓN**

**2003 - ii**

**LIMA – PERÚ**

**2010**

**DISEÑO DEL SISTEMA DE CONTROL PID PARA UN  
EVAPORADOR DE LECHE**

### **Dedicatoria**

En primer lugar a Dios, por ser mi guía y consejero; en segundo lugar, a mi madre y padre por haberme brindado su protección y educación; y en tercer lugar a mis hermanos, amigos y en especial a MM, por todo el apoyo brindado y por alentarme a seguir adelante siempre.

## SUMARIO

En el presente informe se trata sobre el diseño de un sistema de control para una planta de evaporación de leche de efectos múltiples, empleando controladores PID independientes para las variables involucradas en el proceso. Mediante el proceso de evaporación, se busca aumentar la concentración de sólidos de una solución mediante la eliminación del solvente en forma de vapor. Este procedimiento es ampliamente usado en la industria alimentaria.

Las variables involucradas son: flujo de leche en ingreso al primer efecto, temperatura de producto en ingreso a primer efecto y presión de vapor en el termocompresor. Para cada una de éstas variables se halló un modelo matemático del proceso relacionado a dicha variable y a partir del modelo encontrado se halló el controlador PID que cumple con las especificaciones de diseño para el proceso, como el porcentaje de sobre-nivel, tiempo de estabilización y error en estado estacionario.

Para hallar la función de transferencia de los procesos se usó el método gráfico de la curva de reacción, empleando los datos obtenidos a partir de un programa de control y simulación de comportamiento de procesos, SIMATIC PCS7 de SIEMENS, que posee bloques en Laplace, como retardos de primer orden, tiempo muerto, integradores y ruido aleatorio, que se pueden interconectar entre ellos para formar un modelo de proceso bastante real.

Usando como herramienta el programa MATLAB, se exportaron los datos obtenidos de las curvas de reacción y se realizó el análisis de los mismos para calcular los parámetros de los controladores. Intensos estudios de simulación y una posterior implementación han demostrado que los diseños desarrollados en este trabajo, han cumplido enteramente los objetivos de control pre-establecidos.

## INDICE

<b>PRÓLOGO</b> .....	1
<b>CAPÍTULO I</b>	
<b>EVAPORADOR DE EFECTOS MÚLTIPLES</b> .....	2
1.1 Descripción del proceso.....	2
1.2 Variables a controlar.....	4
<b>CAPÍTULO II</b>	
<b>MODELAMIENTO MATEMÁTICO DE PROCESOS</b> .....	8
2.1 Determinación de los parámetros del proceso.....	8
2.2 Obtención de las curvas de reacción.....	10
<b>CAPÍTULO III</b>	
<b>MÉTODOS DE DISEÑO Y SINTONIZACIÓN DE CONTROLADORES PID</b> .....	11
3.1 Controladores PID.....	11
3.2 Métodos de ajuste de controladores PID.....	13
<b>CAPÍTULO IV</b>	
<b>DISEÑO DE LOS CONTROLADORES PID</b> .....	17
4.1 Flujo de producto en ingreso al primer efecto.....	17
4.2 Temperatura de producto en ingreso al primer efecto.....	25
4.3 Presión de vapor en termocompresor.....	31
<b>CONCLUSIONES Y RECOMENDACIONES</b> .....	37
<b>ANEXO</b>	
<b>LISTADO DE PROGRAMAS EN MATLAB Y SIMULINK</b> .....	41
<b>BIBLIOGRAFIA</b> .....	68

## PRÓLOGO

En el presente informe se trata sobre el diseño de un sistema de control PID para una planta de evaporación de leche de efectos múltiples. Mediante el proceso de evaporación se busca aumentar la concentración de sólidos de una solución mediante la eliminación del solvente en forma de vapor. Este procedimiento es ampliamente usado en la industria alimentaria. El presente trabajo se desarrollo en cinco capítulos.

En el capítulo I, se explica el proceso de evaporación, principalmente el funcionamiento de un efecto, el componente principal en dicho proceso, y se establecen las variables más importantes a controlar para garantizar un funcionamiento eficiente y un producto de buena calidad.

En el capítulo II, se explica algunos de los métodos existentes para la determinación de los parámetros y funciones de transferencia de los procesos, que se aplican para la obtención del modelo matemático y la posterior prueba y simulación de los controladores diseñados.

En el capítulo III, se detalla sobre controladores PID y algunos métodos de diseño experimentales. Usando éstos métodos se hallan los parámetros de sintonización para los controladores en estudio.

En el capítulo IV, se halla la función de transferencia de los procesos estudiados y se realiza el diseño del controlador PID correspondiente, usando los métodos descritos en los capítulos anteriores y teniendo como herramienta para los cálculos el programa MATLAB. Se determina el controlador más adecuado que cumple con las especificaciones de diseño para el proceso, como el porcentaje de sobre-nivel, tiempo de subida, tiempo pico y tiempo de estabilización.

Seguidamente, se presentan las conclusiones derivadas del presente estudio y las recomendaciones finales.

En los anexos en la parte final del informe se muestran los programas de MATLAB y modelos de la herramienta SIMULINK usados para el desarrollo de la simulación y las pruebas.

## **CAPÍTULO I EVAPORADOR DE EFECTOS MÚLTIPLES**

### **1.1 Descripción del proceso**

#### **1.1.1 Evaporación**

La evaporación es el procedimiento mediante el cual se retira solvente en forma de vapor de una solución. En nuestro caso, dicho solvente es el agua y la solución es la leche pasteurizada. Existen dos sub-procesos principales en la evaporación: la transferencia de calor, necesaria para llevar la solución a su punto de ebullición, y la separación de vapor – líquido, en donde se obtiene el producto concentrado y el vapor resultante de la ebullición.

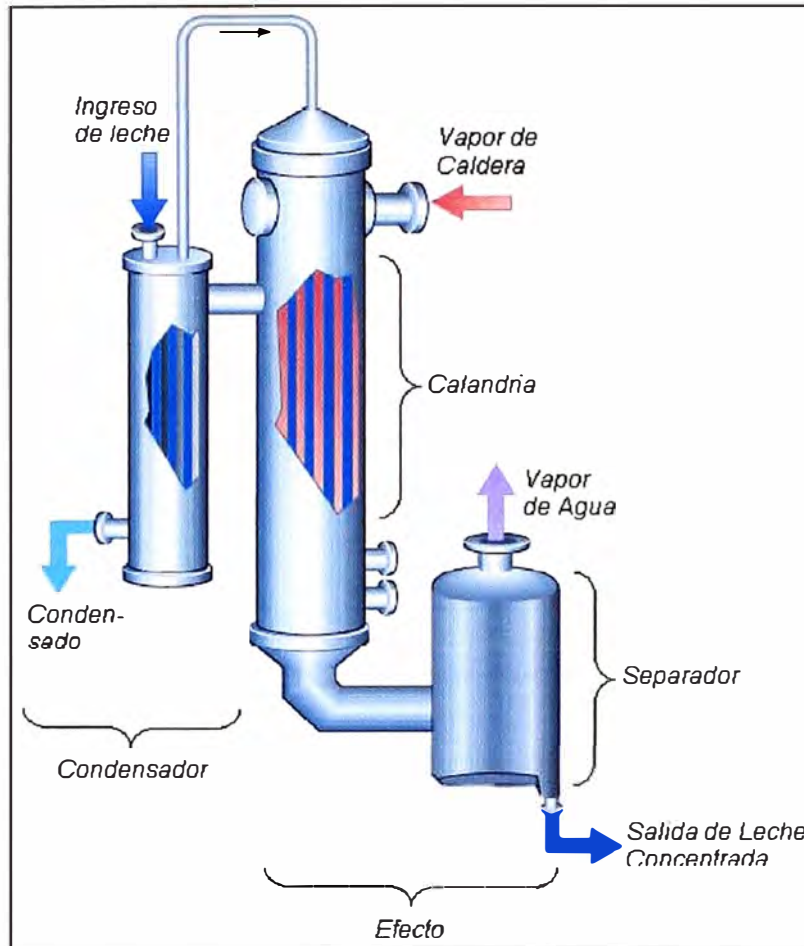
Las unidades en las cuales se lleva a cabo la transferencia de calor se denominan calandrias y aquellas en las que se lleva a cabo la separación de vapor – líquido se conocen como separadores o cámaras de vacío. Un efecto es el término usado para describir el conjunto de una calandria y un separador de vapor. La calandria es un arreglo de tubos verticales de acero inoxidable de diámetro pequeño distribuidos dentro de un cuerpo cilíndrico que los alberga.

El producto, que debe estar a una temperatura cercana a la de ebullición, ingresa uniformemente por la parte superior y se desliza por la parte interna de los tubos formando una película o capa delgada adyacente a la pared interna de las tuberías. Por la parte externa de los tubos circulará vapor de caldera de tal forma que al pasar la leche por la parte interna se producirá la ebullición del agua. La circulación continúa por el separador, en donde el vapor formado se separa del producto concentrado (Figura 1.1).

#### **1.1.2 Evaporadores de Efectos Múltiples**

Para la operación del evaporador se requiere una gran cantidad de energía calorífica para separar agua del producto. Esta energía es suministrada en forma de vapor de caldera. Para reducir la cantidad de vapor de caldera necesario, se usan normalmente los evaporadores de varios efectos, en los cuales, dos o más unidades son conectadas en serie operando progresivamente a menores presiones (más vacío) y en consecuencia a menores puntos de ebullición. Mediante este arreglo, el vapor de agua producido en un efecto puede ser usado como medio de calentamiento en el siguiente. En nuestro caso, la planta de evaporación consta de cinco.

El resultado es que la cantidad de vapor de caldera necesario es aproximadamente igual a la cantidad total de agua evaporada dividida entre el número de efectos. Evaporadores de hasta siete efectos son usados en las industrias lácteas con el fin de disminuir el consumo calorífico. Como ejemplo, en la Figura 1.2 se muestra un evaporador de tres efectos.



**Figura 1.1: Evaporador de película descendente de un efecto**

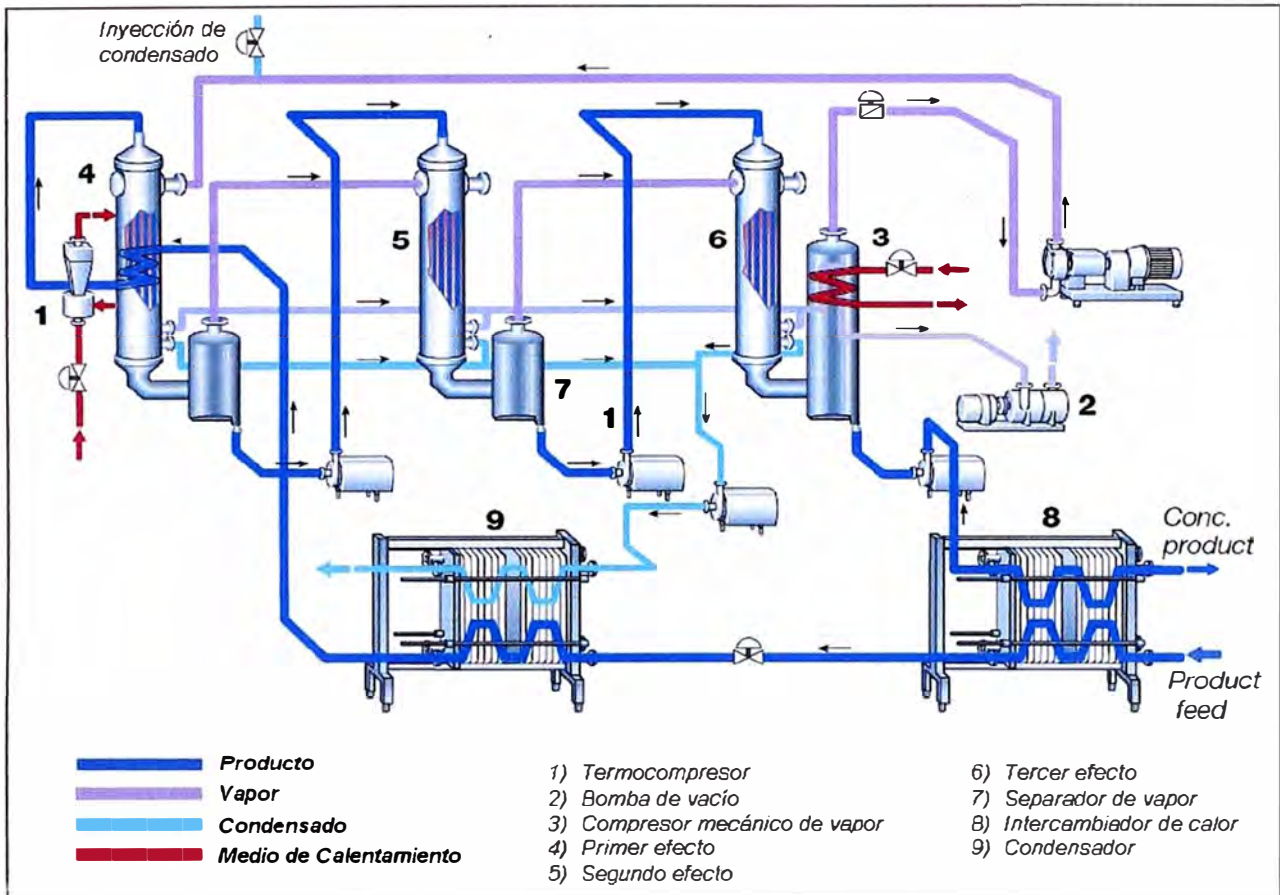
### 1.1.3 Termocompresión del Vapor

Una forma de mejorar la eficiencia térmica de un evaporador es usando la termocompresión. Un termocompresor es un equipo en el cual el vapor de agua obtenido del producto es comprimido y usado como medio de calentamiento.

Parte del vapor de agua proveniente de los separadores es alimentado al termocompresor, en el cual se tiene también una conexión con vapor de caldera a alta presión (entre 6 – 10 bar). El compresor usa la alta presión para incrementar la energía cinética y la mezcla de vapor es expulsada a una alta velocidad a través de la tubería de salida del termocompresor.

Un evaporador de un efecto con termocompresor puede ser tan económico como uno de dos efectos sin termocompresión de vapor. Combinando la termocompresión con los arreglos de múltiples efectos se puede mejorar la eficiencia térmica [1].





**Figura 1.2: Evaporador de tres efectos**

## 1.2 Variables a Controlar

De lo estudiado en la descripción del proceso, se deberá tener control sobre las siguientes variables principales para garantizar un proceso óptimo de evaporación:

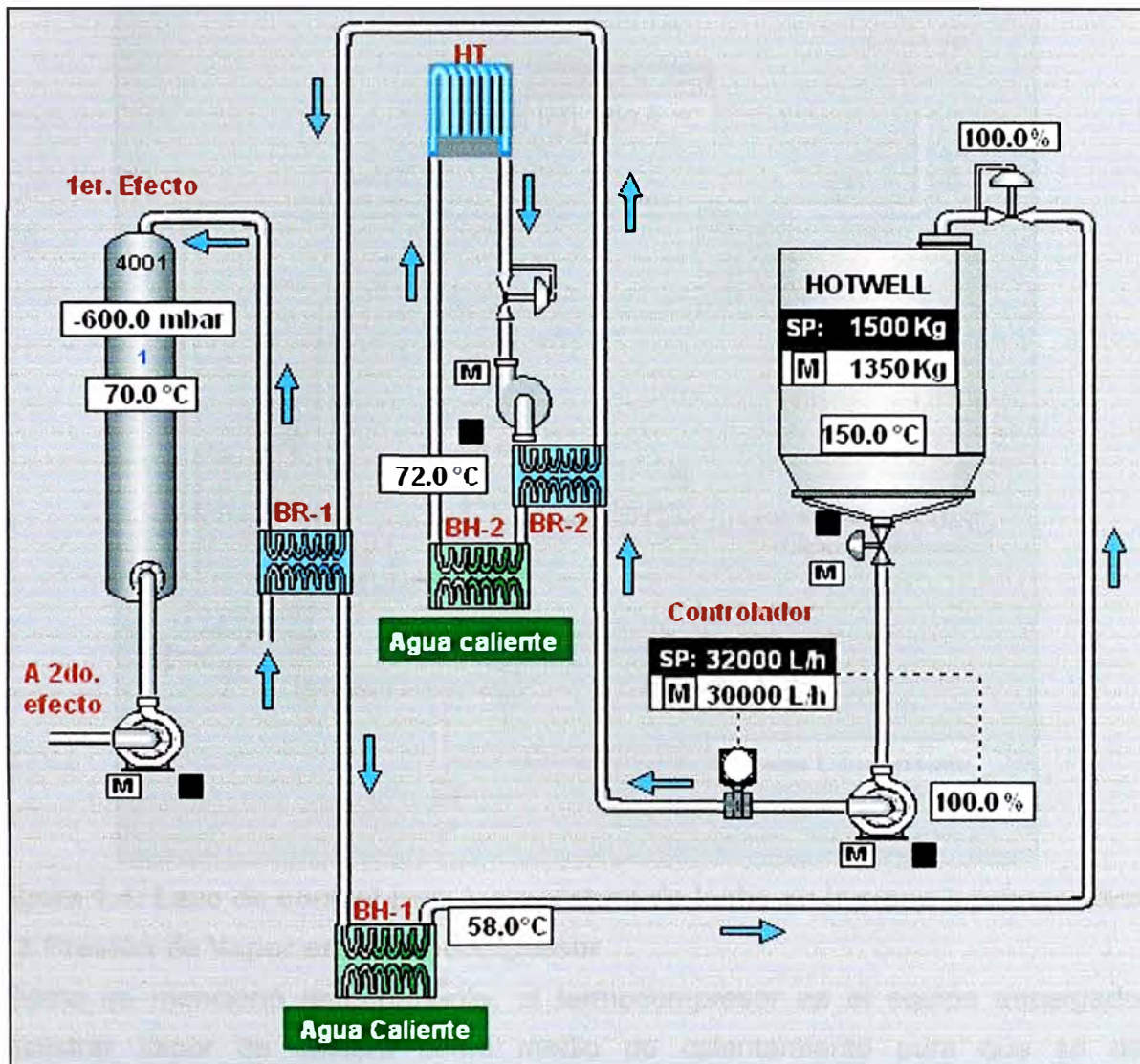
- Flujo de producto en ingreso al primer efecto, que debe ser constante e uniforme.
- Temperatura de producto en ingreso al primer efecto, que debe ser cercana al punto de ebullición para la presión de vacío correspondiente.
- Presión de vapor en el termocompresor, necesario para que se dé la evaporación.

La presión de vacío a través de los efectos, también es una variable de interés, pero que generalmente se toma como parámetro para iniciar el proceso. El valor deseado se alcanza con el encendido de bombas de succión, las cuales se mantienen encendidas mientras dura el proceso de evaporación. No se realiza mayor control sobre éstas.

### 1.2.1 Flujo de Producto en Ingreso al Primer Efecto

La alimentación de producto al primer efecto se hace desde un tanque de almacenamiento temporal, también denominado "tanque pulmón", que debe mantener siempre una cantidad suficiente de producto como para garantizar un flujo uniforme y continuo durante todo el proceso de evaporación. Este es un requisito fundamental para obtener una evaporación uniforme a lo largo de todo el proceso y garantizar la calidad del producto final.

Para mantener el flujo de salida uniforme, se usa una bomba de descarga con variador de velocidad en la salida del tanque de mantenimiento (denominado "hotwell"). El controlador recibe la lectura del caudal de salida mediante un transmisor de flujo de tipo electromagnético, compara la lectura con el valor deseado ("set point") y de acuerdo al error obtenido modificará la referencia de velocidad del variador para tratar de mantener estable el caudal en el valor buscado (Figura 1.3). El diseño de este controlador se hará en el capítulo 4. El valor deseado deberá estar entre 30,000 y 34,000 lt/h.

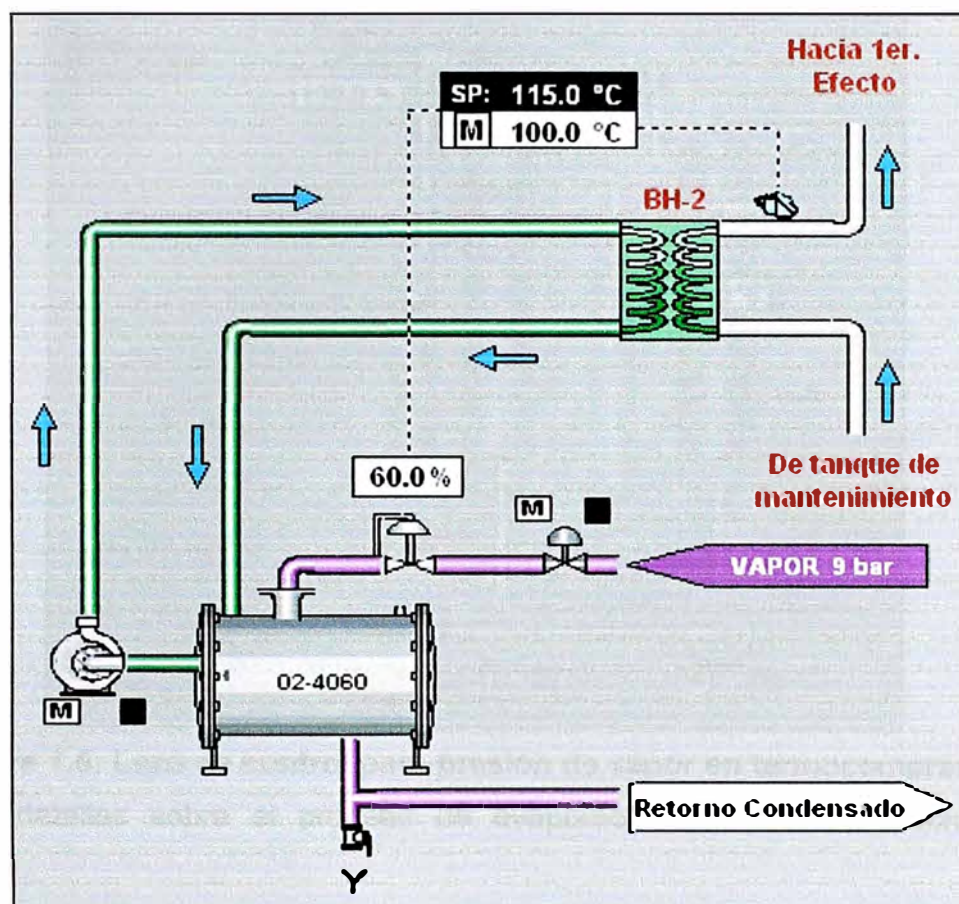


**Figura 1.3: Lazo de control para flujo de producto en ingreso a primer efecto**

### 1.2.2 Temperatura de Producto en Ingreso al Primer Efecto

Para poder separar el solvente en forma de vapor, el producto debe ingresar al primer efecto a una temperatura cercana al punto de ebullición para la presión de vacío correspondiente. La temperatura se controla haciendo circular la leche por un intercambiador de calor de tipo tubular. El medio de calentamiento en dicho intercambiador es agua caliente proveniente de una caldera que posee una válvula proporcional en el ingreso de vapor (Figura 1.4).

En éste caso, el agua caliente debe ingresar al intercambiador a una temperatura de 115°C para poder elevar la temperatura de la leche hasta un valor aproximado de 72°C. Para establecer la temperatura del producto en el valor deseado un controlador recibirá la señal de temperatura de salida de la leche del intercambiador y regulará la cantidad de vapor que ingresa a la caldera manipulando la posición de apertura de la válvula proporcional en la tubería de alimentación de vapor, hasta que se obtenga la temperatura deseada. El diseño del controlador se hará en el capítulo 4.



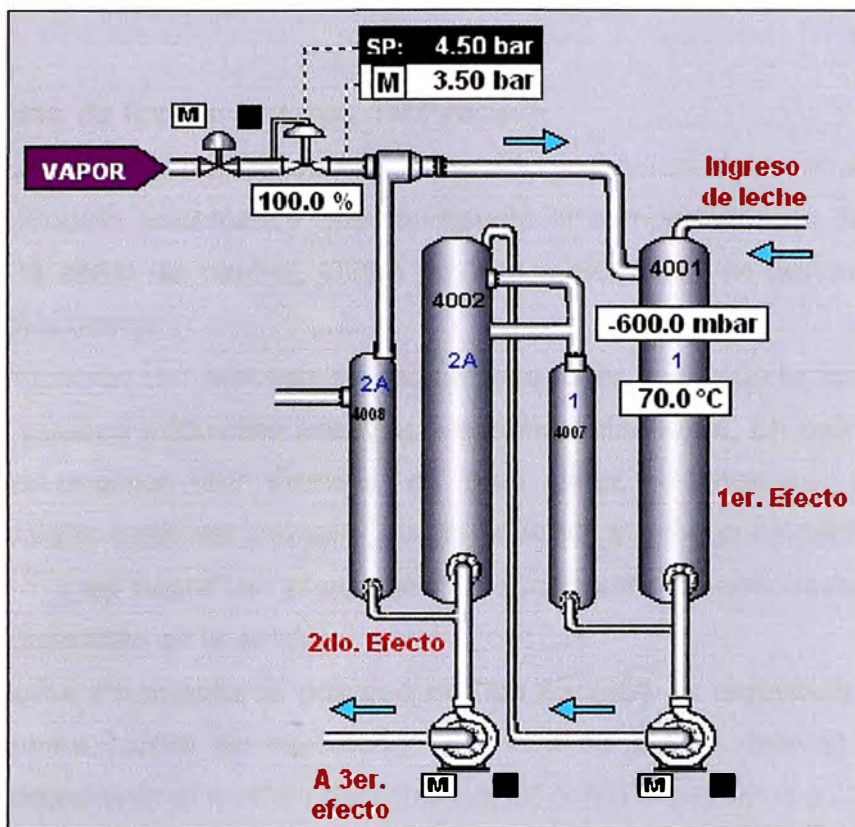
**Figura 1.4: Lazo de control para temperatura de leche en ingreso a primer efecto**

### 1.2.3 Presión de Vapor en Termocompresor

Como se mencionó anteriormente, el termocompresor es el equipo encargado de suministrar vapor de caldera como medio de calentamiento para que se dé la evaporación en los efectos. A su vez, el vapor obtenido en los efectos es recuperado y combinado con el vapor de caldera en el termocompresor, ayudando así al ahorro energético. Para un correcto funcionamiento, es necesario mantener constante y controlada la presión de vapor en el termocompresor.

El control de presión en el termocompresor se realiza mediante la variación del porcentaje de apertura de una válvula proporcional en el ingreso de vapor de caldera hacia el termocompresor, ya que en la caldera siempre existe una presión mayor que en el termocompresor (6 – 8 bar aproximadamente).

Se usa un transmisor de presión para medir ésta variable en la entrada al termocompresor y enviar la señal al controlador. Entonces, el controlador comparará la lectura de presión con el valor deseado (normalmente entre 4 y 5 bar) y de acuerdo al error hallado modificará el porcentaje de apertura de una válvula en el ingreso de vapor entre 0% y 100% para corregir el valor de presión (Figura 1.5).



**Figura 1.5: Lazo de control para presión de vapor en termocompresor**

Mayores detalles sobre el proceso de evaporación se pueden encontrar en la referencia [1].

## CAPÍTULO II MODELAMIENTO MATEMÁTICO DE PROCESOS

### 2.1 Determinación de los Parámetros del Proceso

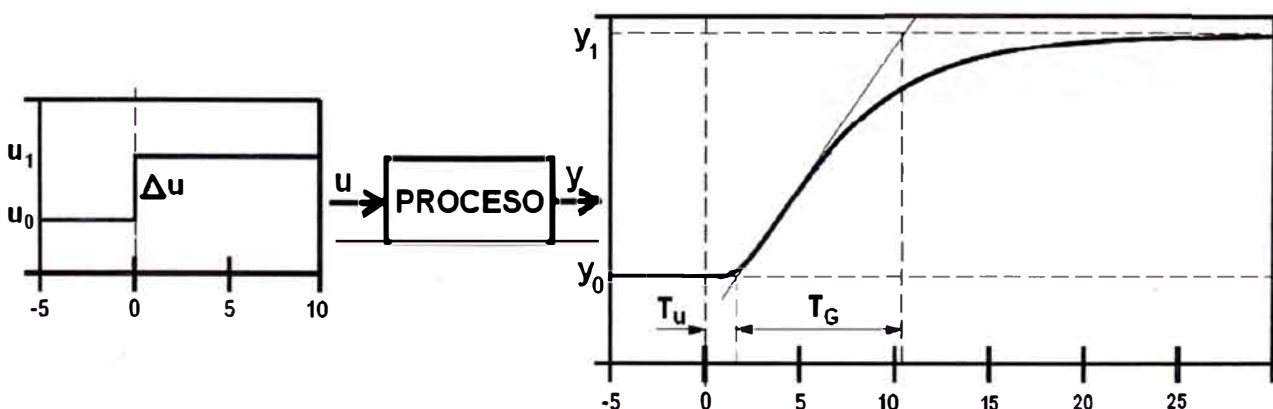
Para poder analizar la acción de un controlador sobre un determinado proceso, se debe hallar un modelo matemático que represente el comportamiento del proceso ante variaciones de la señal de control. Dicho modelo matemático se denomina función de transferencia del proceso.

Para la optimización del proceso se requiere conocer sus características estáticas y dinámicas, que pueden adquirirse analítica o experimentalmente. En muchas ocasiones, el conocimiento analítico del proceso es muy difícil y laborioso. En cambio, el conocimiento experimental del proceso puede llevarse a cabo considerándolo como un recinto cerrado o “caja negra” en el que se introducen señales adecuadas y se analizan las respuestas obtenidas en la salida.

Cuando la señal introducida al proceso es tipo escalón, la respuesta obtenida en la salida se denomina “curva de reacción”. Entonces se puede usar el “método de la tangente” para determinar el modelo del proceso, tal como explicamos a continuación [2].

#### 2.1.1 Método de la Tangente

La respuesta inicial ante una entrada escalón es la más conveniente para el análisis de un proceso y se obtiene pasando el controlador a manual y cambiando bruscamente la posición de la válvula de control. Esto se ilustra en la Figura 2.1.



**Figura 2.1: Respuesta inicial de un proceso ante un cambio brusco en la entrada**

Usando éste método, la mayor parte de los procesos pueden aproximarse a los siguientes modelos de función de transferencia:

### 2.1.1a) Modelo usando un bloque tipo $PT_1T_t$

Combina un elemento de primer orden y un retardo:

$$G_p(s) = \frac{K_p}{1 + T_G \cdot s} \cdot e^{-T_t \cdot s} \quad (2.1)$$

Donde:

$$K_p = \frac{\Delta y}{\Delta u} \quad (2.2)$$

### 2.1.1b) Modelo usando un bloque tipo $PT_n$ (Samal)

Usa la siguiente función de transferencia para modelar el proceso:

$$G_p(s) = \frac{K_p}{(1 + T \cdot s)^n} \quad (2.3)$$

Bloque P:

$$K_p = \frac{\Delta y}{\Delta u}$$

Bloque  $T_n$ :

$$\frac{1}{(1 + T \cdot s)^n} \quad (2.4)$$

Donde "n": entero (orden del sistema)

El orden del sistema "n" y el valor de "T" se hallan de la siguiente tabla:

**Tabla 2.1: Valores de  $T_U/T_G$  y de  $T/T_G$  de acuerdo al orden del sistema**

n	2	3	4	5	6	7	8	9	10
$T_U/T_G$	0.104	0.218	0.320	0.410	0.493	0.591	0.641	0.709	0.775
$T/T_G$	0.368	0.270	0.224	0.195	0.175	0.151	0.148	0.140	0.132

#### Caso especial:

Si al hallar los parámetros de un proceso, se cumple:

$$0 < \frac{T_t}{T_G} < 0.104 \quad (2.5)$$

Entonces, la función de transferencia del proceso se puede representar por:

$$G_p(s) = \frac{K_p}{(1 + T_1 \cdot s) \cdot (1 + T_2 \cdot s)} \quad (2.6)$$

Con:

$$T_2 = K \cdot T_1, \quad K > 1 \quad (2.7)$$

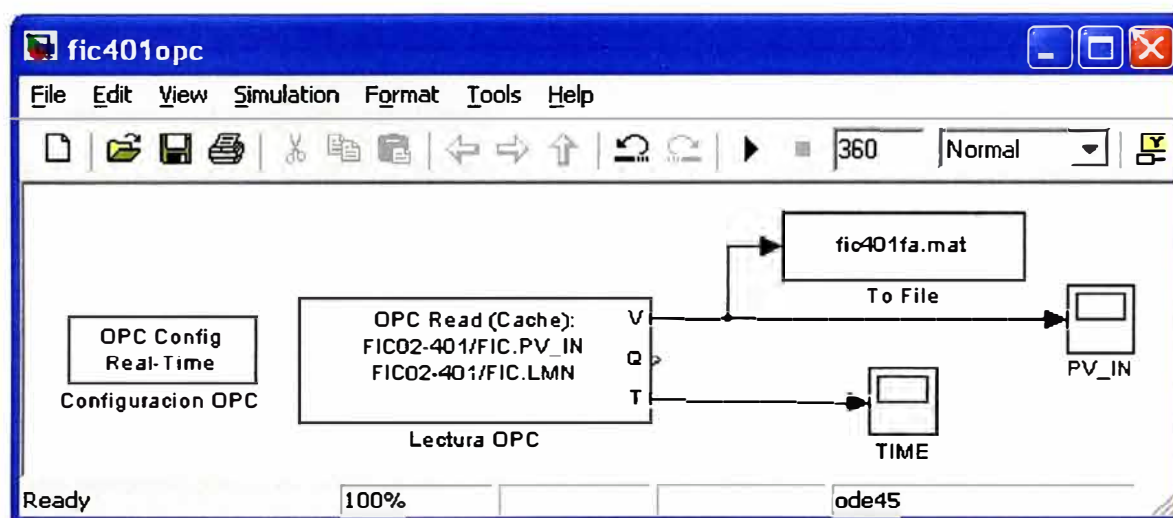
Y donde el valor de " $T_1$ " se halla de la Tabla 2.2.

**Tabla 2.2: Valores de “K” y de “ $T_1/T_G$ ” de acuerdo al valor de “ $T_U/T_G$ ”**

K	1	2	3	4	5	6	7	8	9
$T_U/T_G$	0.094	0.090	0.085	0.080	0.074	0.069	0.064	0.058	0.053
$T_1/T_G$	0.238	0.175	0.140	0.120	0.107	0.097	0.088	0.081	0.074

## 2.2 Obtención de las Curvas de Reacción

Para la obtención de las curvas de reacción, aprovecharemos las posibilidades de conexión de Matlab con un servidor OPC en tiempo real. La planta de evaporación en análisis cuenta con un sistema de automatización con PLC y SCADA basado en servidor OPC. El modelo usado para la conexión en tiempo real se muestra en la Figura 2.2. Con los datos en Matlab podremos realizar el tratamiento y análisis de las curvas de reacción del proceso y calcular los parámetros del mismo.



**Figura 2.2: Modelo en Matlab (Simulink) para comunicación con un Servidor OPC**

Por otro lado, el método para obtener la curva de reacción requiere la acción física sobre el proceso. Como el presente estudio sólo incluye la parte de diseño y simulación se usarán unas funciones de simulación para modelamiento de procesos, que son parte del programa usado para el desarrollo de la automatización. Dichas funciones se aproximan bastante al comportamiento real, ya que ofrece distintos bloques en el dominio de Laplace, como retardos de primer orden, tiempo muerto, integradores y generadores de ruido. Mayor detalle se puede encontrar en la sección 15.2.1 de la referencia [3].

## CAPÍTULO III MÉTODOS DE DISEÑO Y SINTONIZACIÓN DE CONTROLADORES PID

### 3.1 Controladores PID

Como sabemos, un controlador PID está formado por la combinación de tres acciones: la acción proporcional, la acción integral y la acción derivativa. Estas tres acciones combinadas del controlador PID actúan sobre el elemento final de control (por ejemplo una válvula proporcional). Las características esenciales de cada una de estas partes pueden resumirse así [4]:

- a) La acción proporcional varía su salida proporcionalmente a la desviación de la variable con respecto al punto de consigna.
- b) La acción integral varía su salida a una velocidad proporcional a la desviación con respecto al punto de consigna.
- c) La acción derivada varía su valor proporcionalmente a la velocidad de cambio de la variable controlada.

En la Tabla 3.1 se indica una guía general para seleccionar un sistema de control, señalando que sólo sirve de orientación general y debe consultarse únicamente como guía de aproximación al control idóneo.

**Tabla 3.1: Guía de selección de un sistema de control**

Control	Proceso		Cambios de Carga	Aplicaciones
	Capacitancia	Resistencia		
Proporcional	Pequeña a media	Pequeña	Moderados	Presión, temperatura y nivel donde puede haber "offset"
Proporcional + Integral	Cualquiera	Pequeña	Cualquiera	La mayor parte de aplicaciones, incluye caudal
Proporcional + Derivativo	Media	Pequeña	Cualquiera	Cuando es necesaria una gran estabilidad con un "offset" mínimo
Proporcional + Integral + Derivativo	Cualquiera	Grande	Rápido	Procesos con cambios rápidos y retardos apreciables



### 3.1.1 Criterios de Estabilidad en el Control

La estabilidad del control es la característica del sistema que hace que la variable vuelva al punto de consigna después de una perturbación. Los criterios deseables para la estabilidad se representan en la Figura 3.1 y se describen a continuación.

#### 3.1.1.a) Criterio de razón de amortiguamiento.

La amortiguación de la respuesta es tal que la relación de amplitudes entre las crestas de los dos primeros ciclos sucesivos es 0,25, es decir, que cada onda equivale a una cuarta parte de la anterior.

Este criterio es un compromiso entre la estabilidad de la respuesta del controlador y la rapidez del retomo de la variable a un valor estable: una relación mayor que 1/4 dará mayor estabilidad pero prolongará el tiempo de normalización de la variable, mientras que una relación menor que 1/4 devolverá la variable más rápidamente al punto de consigna o a un valor estable, pero perjudicará la estabilidad del sistema.

Este criterio es el más importante y se aplica especialmente en los procesos donde la duración de la desviación es tan importante como el valor de la misma.

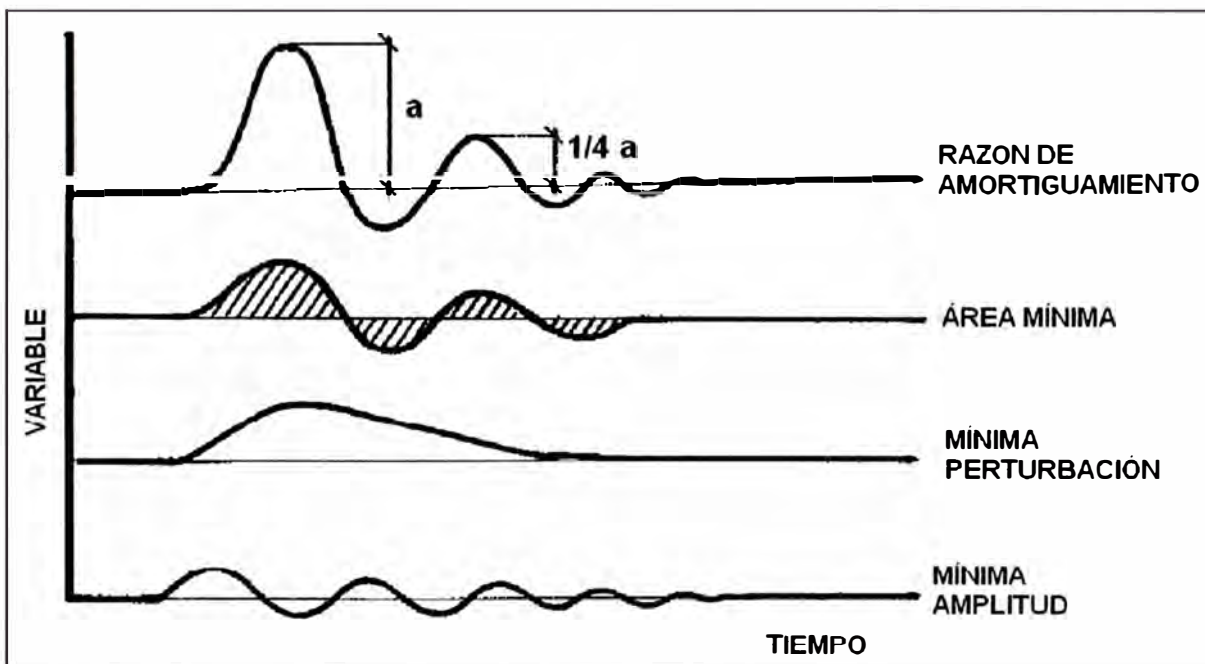


Figura 3.1: Criterios de estabilidad

#### 3.1.1.b) Criterio de área mínima.

Este criterio indica que el área de la curva de recuperación debe ser mínima, para lograr que la desviación sea mínima en el tiempo más corto. Es decir, la expresión:

$$A = \int_0^t |e| dt \quad (3.1)$$

En la que  $|e|$  es el valor absoluto de la desviación, debe tener el valor mínimo. A éste método se le denomina criterio de la "integral mínima del valor absoluto del error" (IAE).

### **3.1.1.c) Criterio de mínima perturbación.**

Este requiere una curva de recuperaciones no cíclicas, y se aplica cuando por ejemplo, las correcciones rápidas o cíclicas de una válvula de control de vapor, pueden perturbar seriamente las presiones de vapor de alimentación e influir en otros procesos alimentados por la misma fuente. Otro caso puede ser el control en cascada en que la señal de salida de un controlador varíe cíclicamente y se aplique como punto de consigna en un segundo controlador creándole serias variaciones de carga.

### **3.1.1.d) Criterio de amplitud mínima.**

De acuerdo con este criterio, la amplitud de la desviación debe ser mínima, lo cual se aplica especialmente a procesos en que el producto o el equipo puede ser dañado por desviaciones momentáneas excesivas, y en este caso la magnitud de la desviación es más importante que su duración. Por ejemplo, en el caso de fusión de algunas aleaciones metálicas, el sobrepasar temporalmente una determinada temperatura puede destruir el metal. Se aplica también este criterio en el caso de procesos exotérmicos con el punto de consigna próximo a la temperatura de disparo de la reacción incontrolable.

## **3.2 Métodos de Ajuste de Controladores PID**

Existen varias formas para ajustar los controladores al proceso, es decir, para que la banda proporcional (ganancia), el tiempo de acción integral (minutos/repetición) y el tiempo de acción derivada (minutos de anticipo) del controlador, caso de que posea las tres acciones, se acoplen adecuadamente con el resto de los elementos del bucle de control (proceso + transmisor + válvula de control). Este acoplamiento debe ser tal que, ante una perturbación, se obtenga una curva de recuperación que satisfaga cualquiera de los criterios mencionados para que el control sea estable, en particular, el de área mínima con una relación de amortiguación de 0,25 entre crestas sucesivas de la onda.

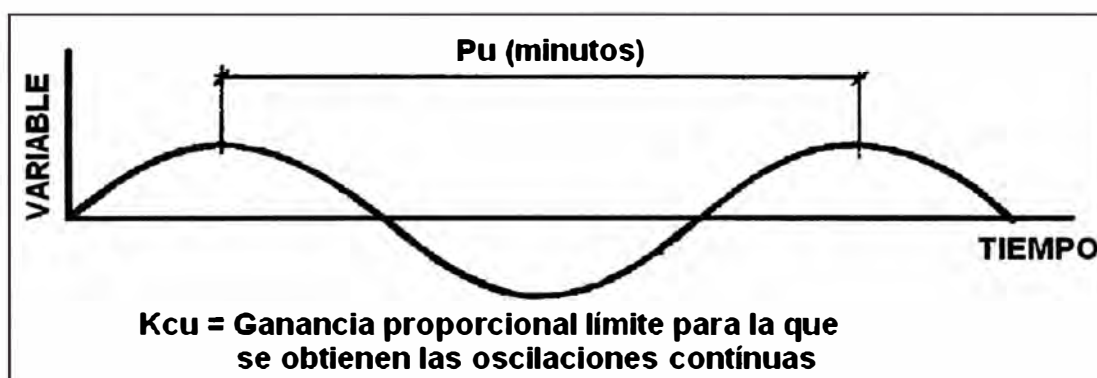
Para que este acoplamiento entre el controlador y el proceso sea posible es necesario un conocimiento inicial de las características estáticas y dinámicas del sistema controlado. Existen dos métodos fundamentales para determinar estas características, el método analítico y el método experimental. El método analítico se basa en determinar la ecuación relativa a la dinámica del proceso, es decir, su evolución en función del tiempo, lo cual generalmente es difícil de hallar debido a la complejidad de los procesos industriales. En el método experimental las características estáticas y dinámicas del proceso se obtienen a partir de una serie de medidas realizadas sobre el proceso real. Estas respuestas del proceso pueden obtenerse de tres formas principales:

- a) Método de tanteo (lazo cerrado).
- b) Método de ganancia límite (lazo cerrado).
- c) Método de curva de reacción (lazo abierto).

### 3.2.1 Método de ganancia límite.

Este método de lazo cerrado fue desarrollado por Ziegler y Nichols, en 1941 y permite calcular los tres términos de ajuste del controlador a partir de los datos obtenidos en una prueba rápida de características del bucle cerrado de control.

El método se basa en aumentar gradualmente la ganancia o equivalentemente reducir la banda proporcional con los ajustes de integral y derivada en su valor de menor influencia, mientras se crean pequeños cambios en el punto de consigna o "set point", hasta que el proceso empieza a oscilar de modo continuo y con amplitud constante (Figura 3.2).



**Figura 3.2: Oscilación mantenida (sensibilidad límite)**

El valor de la ganancia así obtenida se denomina "ganancia límite" ( $K_{CU}$ ). Equivalentemente se habla de una "banda proporcional límite" ( $P_{BU}$ ). Se anota entonces este valor y el periodo del ciclo de las oscilaciones  $P_U$  en minutos, y la última ganancia proporcional  $K_{CU}$ .

Los ajustes de control que producirán aproximadamente una respuesta con una relación de amplitudes 0,25 se calculan como sigue:

Controlador Proporcional (P):

$$a) \text{ Ganancia proporcional} = 0.5 \cdot K_{CU} \quad (3.2)$$

Controlador Proporcional + Integral (PI):

$$a) \text{ Ganancia proporcional} = 0.45 \cdot K_{CU} \quad (3.3)$$

$$b) \text{ Tiempo integral (min/rep)} = \frac{P_U}{1.2} \quad (3.4)$$

Controlador Proporcional + Integral + Derivativo (PID):

$$a) \text{ Ganancia proporcional} = 0.6 \cdot K_{CU} \quad (3.5)$$

$$b) \text{ Tiempo integral (min/rep)} = \frac{P_U}{2} \quad (3.6)$$

$$c) \text{ Tiempo derivativo (min)} = \frac{P_U}{8} \quad (3.7)$$

### 3.2.2 Método de Mínimo Rebasamiento del Punto de Consigna.

Una variante del método de ganancia límite es el *método de mínimo rebasamiento del punto de consigna*. Una vez obtenida la oscilación auto mantenida de periodo  $P_U$  en minutos para una ganancia límite ( $K_{CU}$ ), las acciones de control son:

$$a) \text{ Ganancia proporcional (\%)} = 0.8 \cdot K_{CU} \quad (3.8)$$

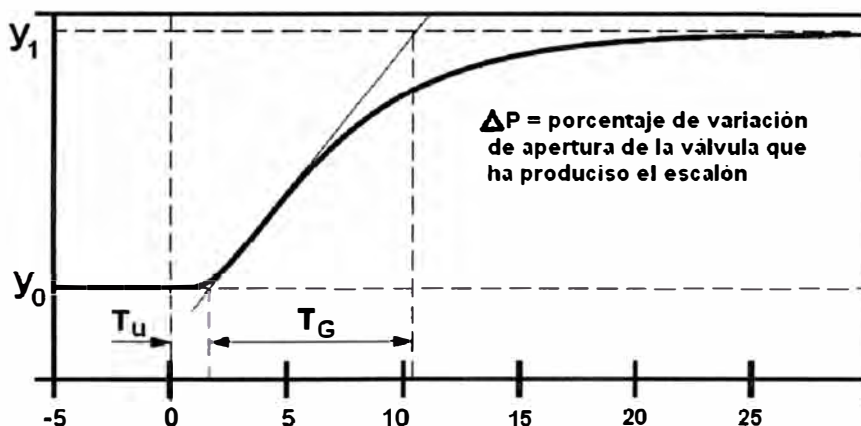
$$b) \text{ Tiempo integral (min/rep)} = 0.6 \cdot P_U \quad (3.9)$$

$$c) \text{ Tiempo derivativo (min)} = 0.19 \cdot P_U \quad (3.10)$$

### 3.2.3 Método de la Curva de Reacción.

En este método de lazo abierto, el procedimiento general consiste en abrir el bucle cerrado de regulación antes de la válvula o elemento final de control, es decir, operar directamente la válvula con el controlador en manual y crear un pequeño y rápido cambio en escalón en la señal de entrada. La respuesta obtenida se introduce en un registrador de gráfico de banda de precisión o usando una curva de tendencia si el sistema tiene software de supervisión (SCADA), con el mayor tamaño posible del gráfico para obtener la mayor exactitud.

En el punto de inflexión de la curva obtenida se traza una tangente lo más aproximada posible y se miden los valores  $T_U$  y  $T_G$  (Figura 3.3).



**Figura 3.3: Curva de reacción**

Si se aproxima el proceso a uno de primer orden su función de transferencia es:

$$G_p(s) = \frac{K_p}{1 + T_G \cdot s} \cdot e^{-T_U \cdot s} \quad (3.11)$$

Donde:  $K_p$  = ganancia del proceso,  $T_U$  = tiempo de retardo del proceso y  $T_G$  = constante de tiempo del proceso. El retardo  $T_U$  es el tiempo en minutos que transcurre entre el instante del cambio en escalón y el punto en que la tangente anterior cruza el valor inicial de la variable controlada.  $\Delta P$  es el porcentaje de variación de posición de la válvula de control que introduce el escalón en el proceso.

Las formulas a aplicar son las siguientes:

Controlador Proporcional (P):

a) Ganancia proporcional:  $K_C = \frac{T_G}{K_P \cdot T_U}$  (3.12)

Controlador Proporcional + Integral (PI):

a) Ganancia proporcional:  $K_C = 0.9 \cdot \frac{T_G}{K_P \cdot T_U}$  (3.13)

b) Tiempo integral:  $T_I = 3.33 \cdot T_U$  (3.14)

Controlador Proporcional + Integral + Derivativo (PID):

a) Ganancia proporcional:  $K_C = 1.2 \cdot \frac{T_G}{K_P \cdot T_U}$  (3.15)

b) Tiempo integral:  $T_I = 2 \cdot T_U$  (3.16)

c) Tiempo derivativo:  $T_D = 0.5 \cdot T_U$  (3.17)

## **CAPÍTULO IV DISEÑO DE LOS CONTROLADORES PID**

Se realizó el análisis de la respuesta de los sistemas más importantes involucrados en el proceso de evaporación que se mencionaron en el capítulo 1, que son:

- a) Flujo de leche en ingreso al primer efecto.
- b) Temperatura de leche en ingreso al primer efecto.
- c) Presión de vapor en el termocompresor.

Para esto se usaron los métodos descritos en los capítulos 2 y 3 para la obtención de la función de transferencia del proceso y los parámetros de sintonización del controlador, respectivamente.

### **4.1 Flujo de Producto en Ingreso al Primer Efecto**

#### **4.1.1 Objetivos de Control y Especificaciones de Diseño**

El objetivo es mantener constante el flujo de alimentación al evaporador en el primer efecto. El producto es alimentado al primer efecto desde un tanque de almacenamiento temporal denominado “tanque de mantenimiento”. Por lo tanto, en éste caso la variable de proceso a controlar es el flujo de salida del tanque de mantenimiento. La variable manipulada es la referencia de velocidad del variador de la bomba de descarga del tanque.

El controlador diseñado debe ser capaz de llevar rápidamente el flujo de salida al valor deseado y sin que exista demasiado sobre-impulso ni oscilaciones permanentes. El controlador a usar será de tipo PID, que nos da mayores posibilidades de controlar las variaciones repentinas del flujo.

Para cumplir con el objetivo de control, planteamos las siguientes condiciones de diseño para el controlador:

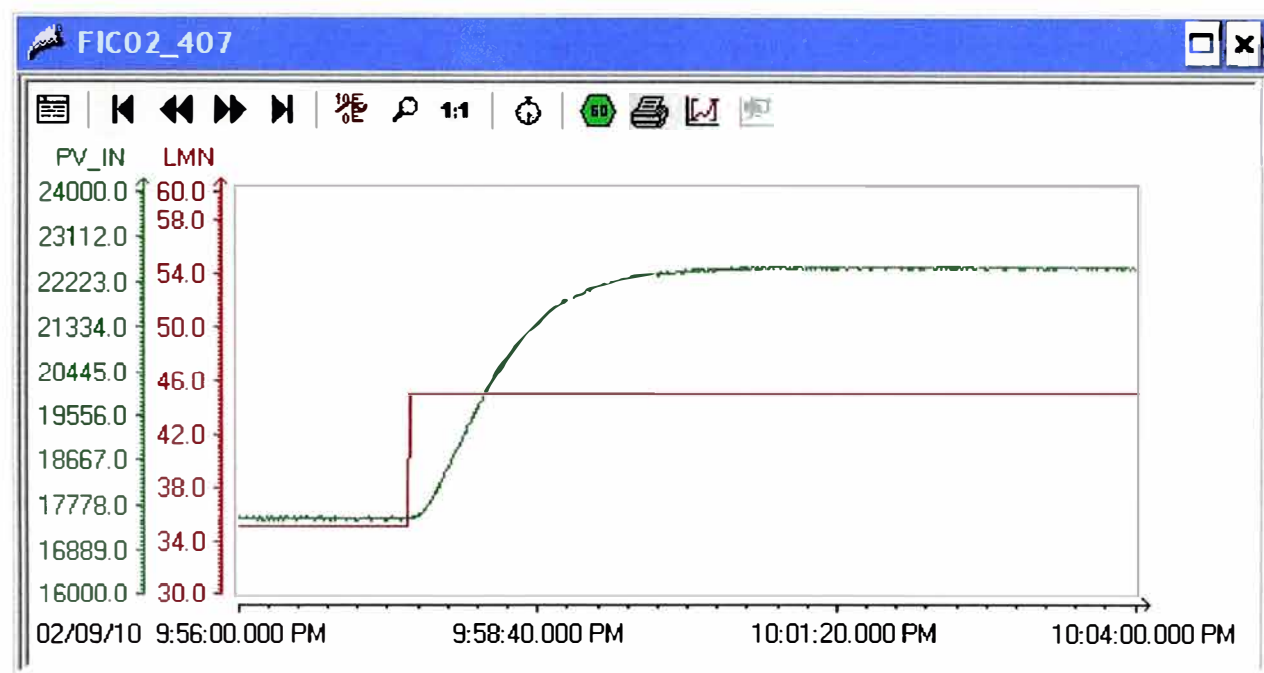
- a) Porcentaje de sobre-nivel (“overshoot”): 10%.
- b) Tiempo de subida (“rise time”): 15 segundos.
- c) Tiempo pico (“peak time”): 20 segundos.
- d) Tiempo de estabilización (“settling time”): 60 segundos.
- e) Error en estado estacionario (“offset”): +/- 2% del valor deseado.

Para poder satisfacer dichas condiciones usamos el método de mínimo rebasamiento del punto de consigna (sección 3.2.2), que es el que produce la mejor respuesta.

## 4.1.2 Obtención de la Función de Transferencia del Proceso

### Curva de reacción

Para obtener la respuesta del proceso, colocamos el controlador en modo manual y estando la referencia de velocidad del variador en 35%, se aplicará una variación brusca, tipo escalón, hasta 45%. Esto se hace en el software de supervisión SCADA de la planta y los datos se obtienen en una curva de tendencia (Figura 4.1).



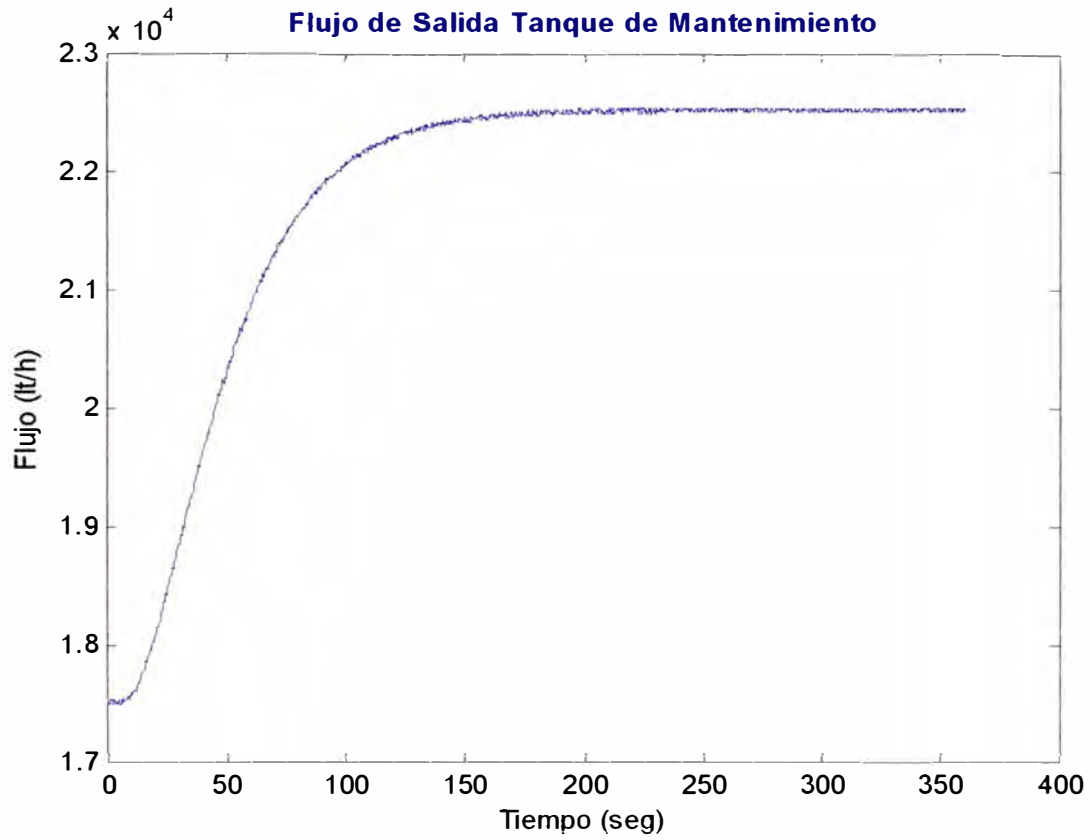
**Figura 4.1: Curva de reacción para flujo de salida en software SCADA**

Los datos obtenidos se exportan hacia el software MATLAB mediante un modelo en SIMULINK para comunicación con un servidor OPC. El modelo usado se puede encontrar en la sección A.1.1 del anexo (“fic407opc.mdl”). Los datos obtenidos se grafican contra el tiempo en segundos y se pueden apreciar en la Figura 4.2.

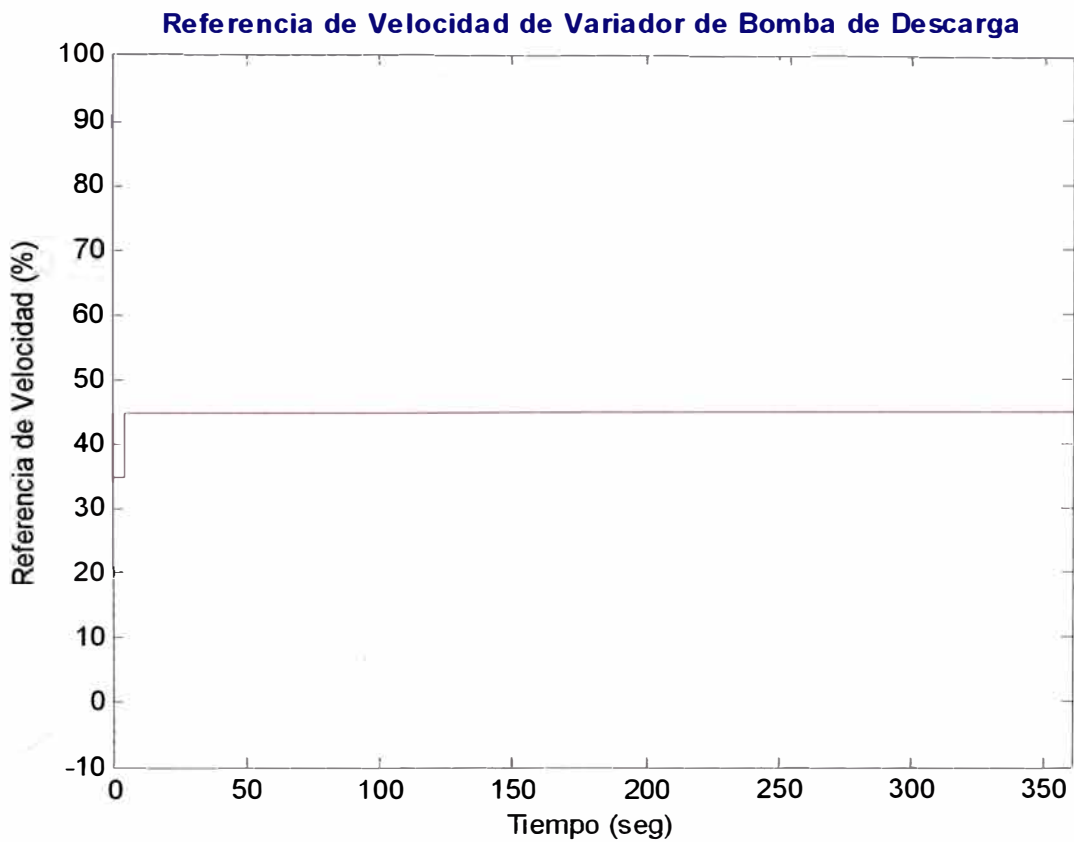
### Cálculo de parámetros para el modelo del proceso

Teniendo los datos en MATLAB, realizamos el análisis de la señal para hallar los parámetros del proceso. Para esto, hallamos los valores de  $T_U$  y  $T_G$  a partir de la curva de reacción obtenida. De acuerdo al procedimiento descrito anteriormente, debemos encontrar el punto de inflexión en la curva de reacción y luego trazar una recta tangente en dicho punto. El punto de inflexión se hallará como el punto de máxima pendiente, o sea, el punto en el cual la primera derivada alcanza su máximo valor.

Para poder realizar un mejor tratamiento de la señal, se le aplicará primero una función de alisamiento que nos servirá para “suavizar” la curva y eliminar el ruido presente en ella. Esta es una función ampliamente utilizada en el tratamiento de señales. Su implementación se hace en MATLAB (sección A.4 del anexo) con la función “fastsmooth2.m”. El resultado se muestra en la Figura 4.3.



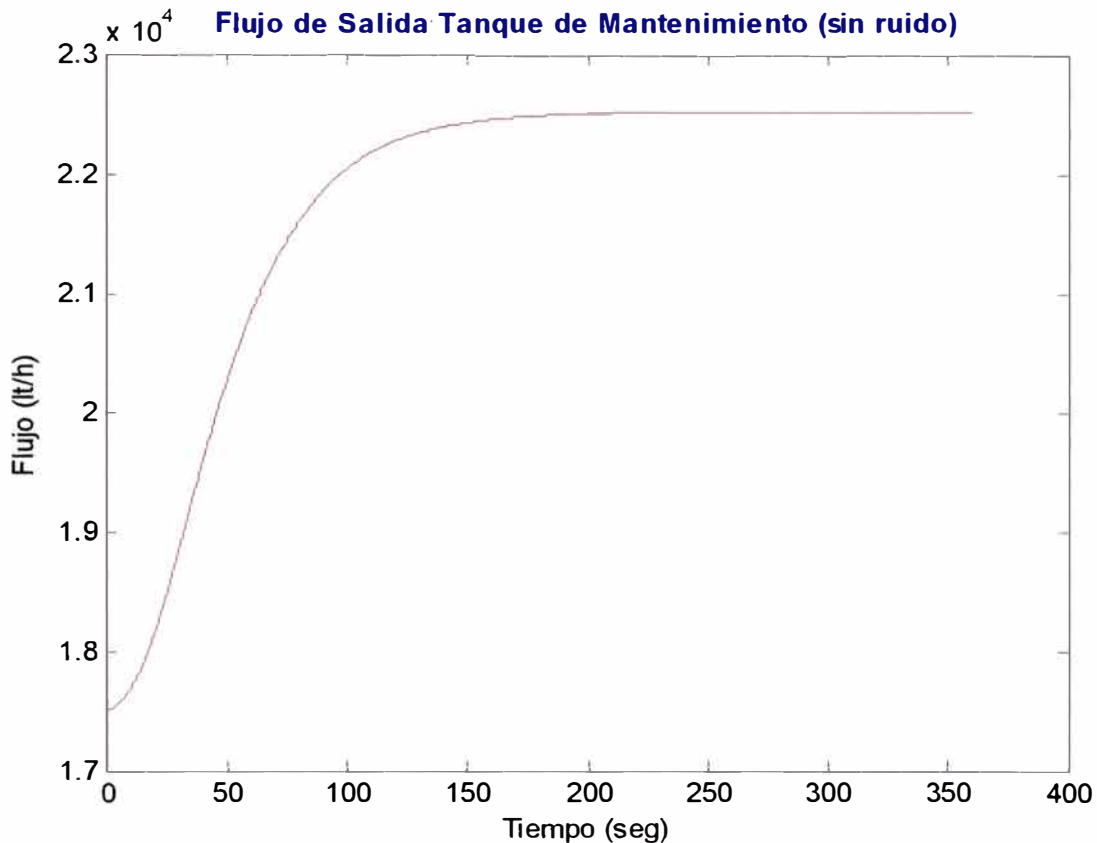
**a) Flujo de salida de tanque de mantenimiento**



**b) Referencia de velocidad de variador de bomba de descarga**

**Figura 4.2: Curva de reacción obtenida en MATLAB para controlador de flujo**





**Figura 4.3: Flujo de salida del tanque de mantenimiento con eliminación de ruido**

Los cálculos para encontrar el valor máximo de la primera derivada se realizan de manera discreta, mediante diferencias sucesivas de los elementos del vector de datos. El procedimiento completo se puede encontrar en la sección A.1.2 del anexo (programa "fic407anl.m"). El resultado se muestra en la Figura 4.4.

Una vez hallado el punto de inflexión, trazamos una recta tangente en dicho punto a la curva de reacción y calculamos gráficamente los valores de " $T_U$ " y " $T_G$ " (Figura 4.5). Los valores obtenidos son los siguientes:

$$T_U = 8.25 \text{ seg}$$

$$T_G = 66.15 \text{ seg}$$

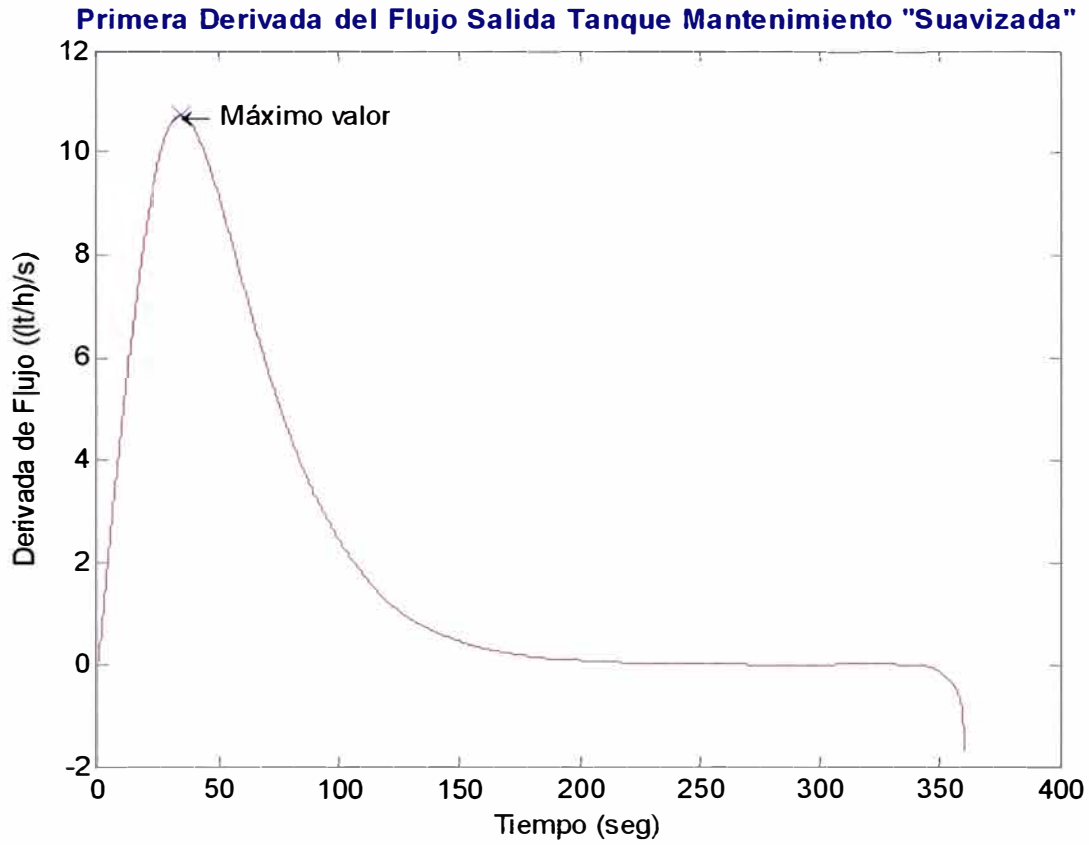
Y la ganancia del proceso es:

$$K_p = \frac{\Delta Y}{\Delta u} = 500.56$$

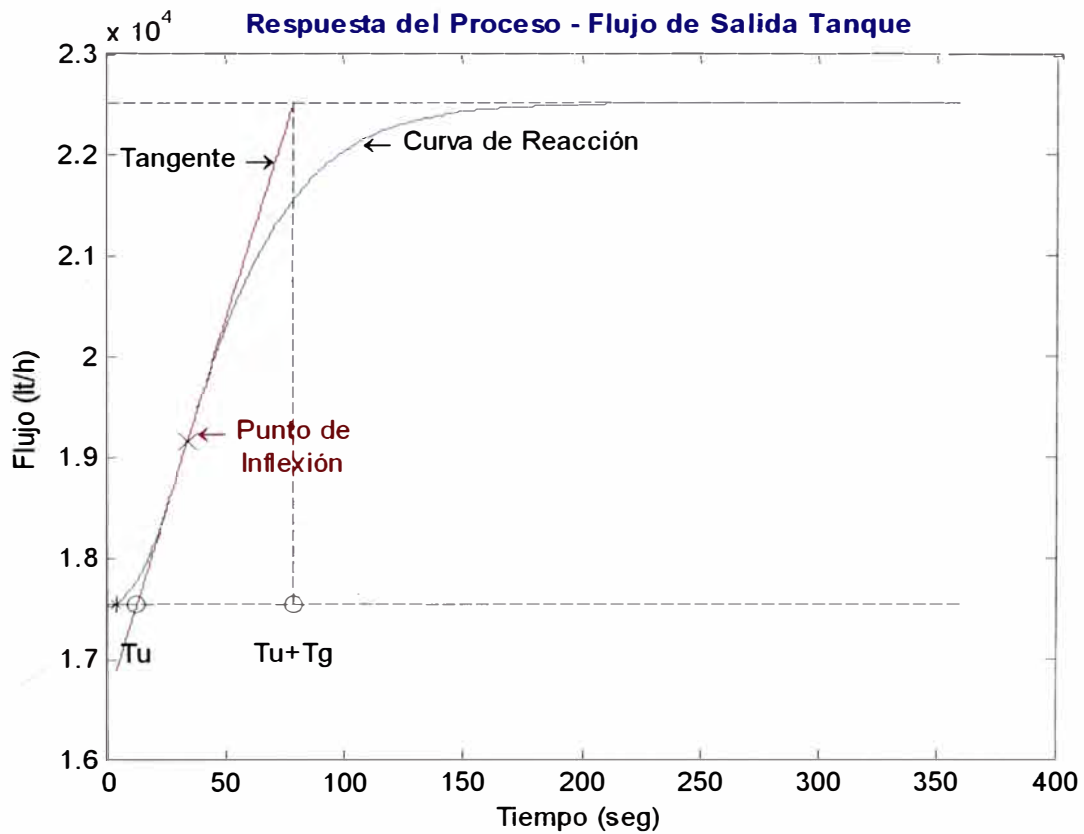
Con lo cual, si usamos un bloque tipo  $PT_n$ , según la ecuación 2.3 y la tabla 2.1, hallamos la siguiente función de transferencia para el proceso:

$$T = 0.368 \cdot T_G = 24.3432 \text{ seg. (para } n = 2)$$

$$G_P(s) = \frac{500.56}{(1 + 24.3432 \cdot s)^2} \quad (4.1)$$



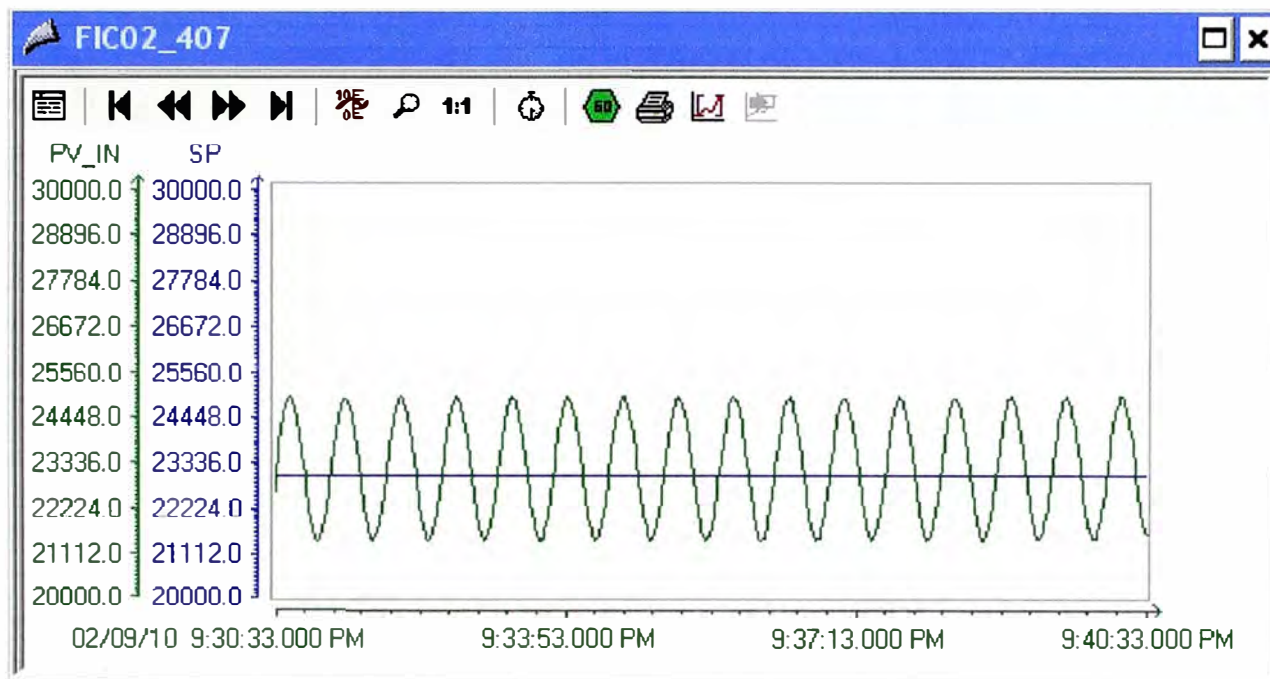
**Figura 4.4: Primera derivada del flujo de salida del tanque de mantenimiento**



**Figura 4.5: Determinación gráfica de los parámetros del proceso " $T_u$ " y " $T_G$ "**

### 4.1.3 Cálculo de Parámetros para el Controlador

Para determinar los parámetros del controlador, se usa el método de mínimo rebasamiento del punto de consiga, que se explicó en la sección 3.2.2. Siguiendo dicho procedimiento, colocamos el controlador en automático y aumentamos gradualmente la ganancia proporcional, estando los otros parámetros en su valor de menor influencia, hasta que la variable de proceso presente oscilaciones sostenidas (Figura 4.6).



**Figura 4.6: Oscilación sostenida con ganancia proporcional límite**

Estas oscilaciones mantenidas se dan para el siguiente valor de ganancia proporcional límite, hallado con ayuda del software de supervisión:

$$K_{CU} = 0.04 \quad (4.2)$$

Los datos son exportados nuevamente a MATLAB usando un modelo para comunicación con servidor OPC (archivo "fic407osc.mdl" en la sección A.1.3 del anexo) y haciendo el análisis de la curva se obtiene el valor del periodo de oscilación en segundos (archivo "fic407mgl.m" en la sección A.1.4 del anexo). La gráfica de los datos se muestra en la Figura 4.7 y el valor obtenido para el periodo de oscilación es:

$$P_{CU} = 38.2 \quad (4.3)$$

Luego, aplicando las fórmulas 3.8, 3.9 y 3.10 del método de mínimo rebasamiento, obtenemos los siguientes valores para los parámetros del controlador PID:

$$K_C = 0.032, T_I = 22.92 \text{ seg. y } T_D = 7.258 \text{ seg.} \quad (4.4)$$

Finalmente, la función de transferencia del controlador PID sería la siguiente:

$$G_C(s) = 0.032 \cdot \left(1 + \frac{1}{22.92 \cdot s} + 7.258 \cdot s\right) \quad (4.5)$$

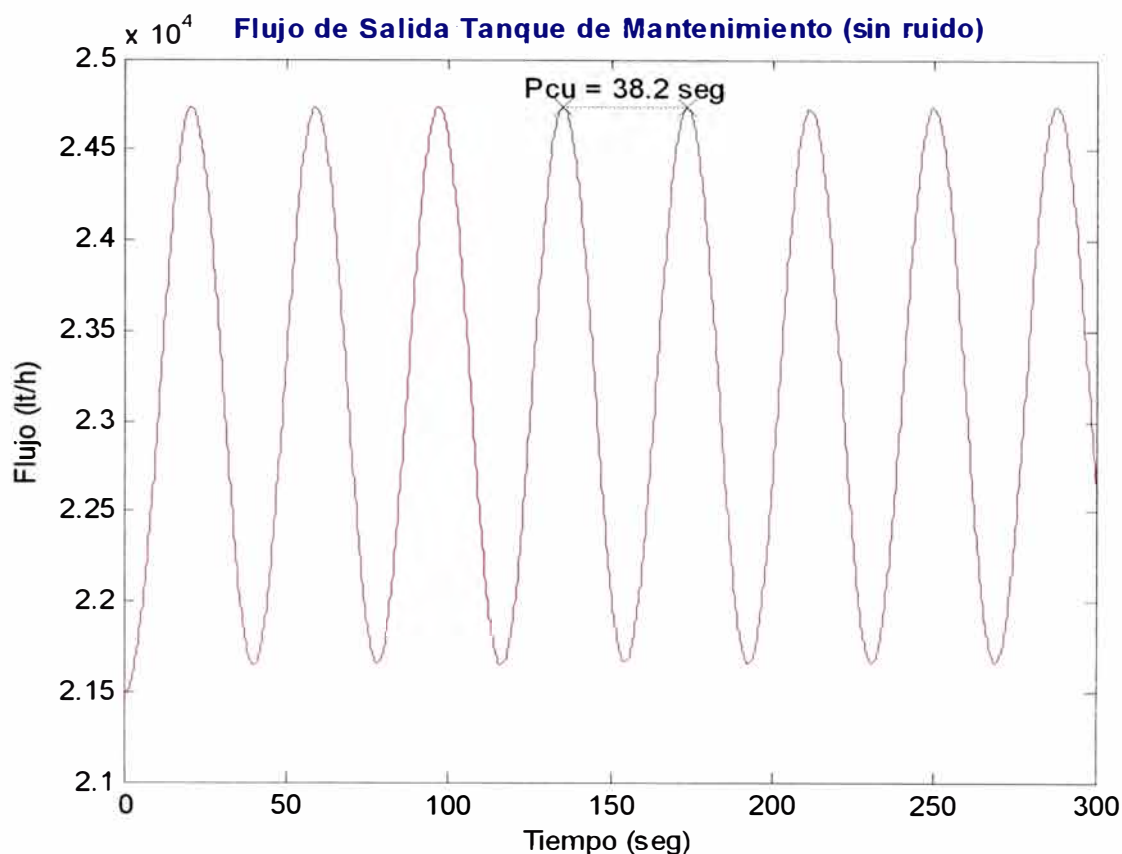


Figura 4.7: Cálculo del periodo de oscilación para ganancia proporcional límite

#### 4.1.4 Análisis de la Acción del Controlador sobre el Proceso

Teniendo la función de transferencia del proceso (ecuación 4.1) y la del controlador (ecuación 4.5) podemos implementar un modelo en Matlab para analizar la respuesta del proceso en lazo cerrado bajo la acción del controlador ante un cambio tipo escalón en el valor deseado (Figura 4.8). El cambio en escalón en el valor deseado se hace de 25,000lt/h a 32,000lt/h.

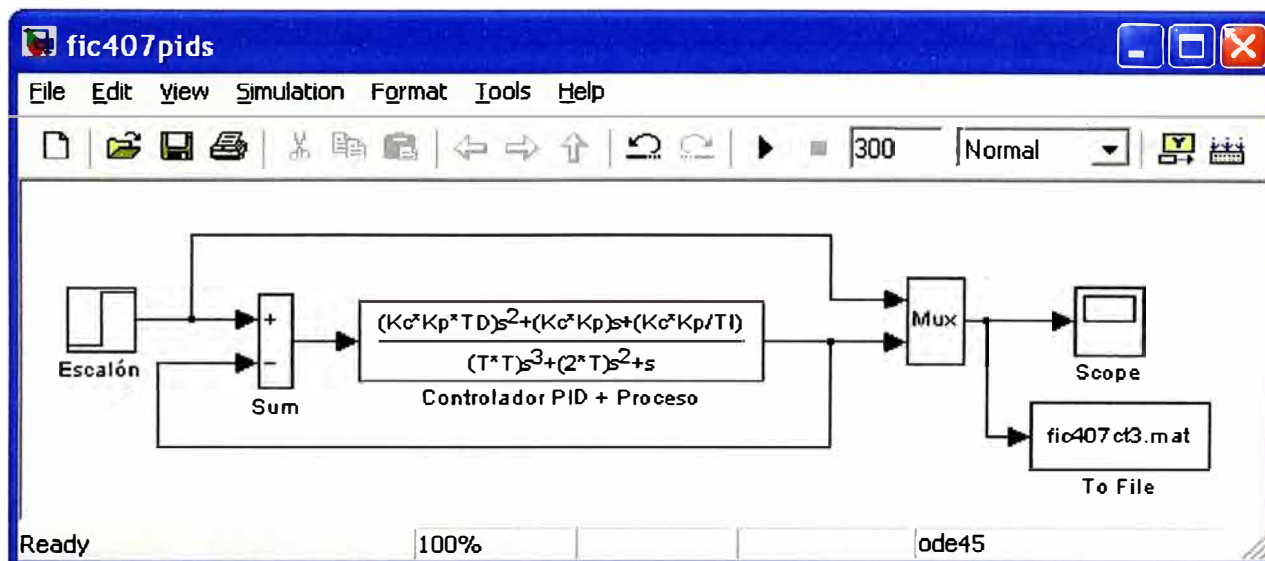


Figura 4.8: Modelo de proceso con acción de controlador en lazo cerrado

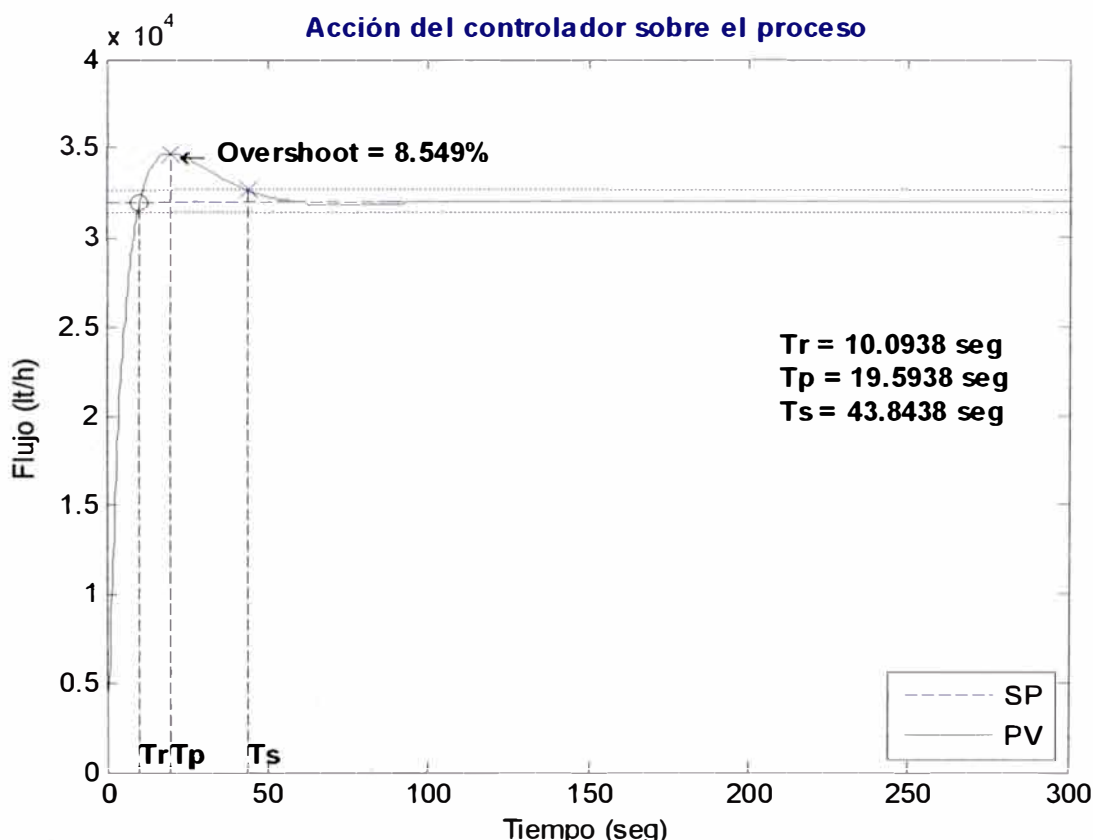
Para determinar si nuestro controlador cumple las especificaciones de diseño, realizamos el análisis de la respuesta obtenida (archivo "fic407rsl.m" en la sección A.1.7 del anexo). Los valores obtenidos de las características de la respuesta inicial del proceso son:

- Porcentaje de sobre-nivel ("overshoot"): 12.6864%
- Tiempo de subida ("Tr"): 9.85 seg.
- Tiempo pico ("Tp"): 18.6 seg.
- Tiempo de estabilización (para error en estado estacionario de 2%): 39.35 seg.

Por lo tanto, se puede apreciar que debemos hacer un ajuste adicional para tratar de mejorar el valor del porcentaje de sobre-nivel. Para esto simplemente aumentamos ligeramente el tiempo derivativo. Luego de realizar varias pruebas, se obtiene el siguiente valor óptimo para una respuesta mejorada:

$$T_d = 9.0 \text{ seg.} \quad (4.6)$$

La respuesta final del proceso bajo la acción del controlador PID se muestra en la Figura 4.9.



**Figura 4.9: Respuesta final de la acción del controlador sobre el proceso**

De esta forma, se cumple con todas las especificaciones de diseño planteadas inicialmente, tanto para el porcentaje de sobre-nivel como para el tiempo de estabilización, con lo cual se termina el diseño del controlador.

## **4.2 Temperatura de producto en ingreso al primer efecto**

### **4.2.1 Objetivos de Control y Especificaciones de Diseño**

El objetivo es mantener la temperatura de la leche en el ingreso al primer efecto cercana al punto de ebullición para la presión de vacío correspondiente (aproximadamente 72°C). La temperatura del producto se controla haciendo circular la leche por un intercambiador de calor tipo tubular antes de ingresar al evaporador. El medio de calentamiento usado es vapor de agua proveniente de una caldera. Por lo tanto, la variable a controlar es la temperatura de salida de la leche del intercambiador. La variable manipulada es el porcentaje de apertura de una válvula proporcional en el ingreso de vapor a la caldera, lo cual permitirá regular indirectamente la temperatura de salida de la leche del intercambiador. Se usará un controlador tipo PID.

El controlador diseñado debe ser capaz de llevar la variable de proceso al valor deseado sin que exista demasiado sobre-nivel, ya que se podría dañar el producto debido a una alta temperatura. Como los cambios de temperatura son lentos, se puede admitir un tiempo largo de estabilización. Por lo tanto, para cumplir con el objetivo de control, planteamos las siguientes condiciones de diseño para el controlador:

- a) Porcentaje de sobre-nivel ("overshoot"): 7.5%.
- b) Tiempo de subida ("rise time"): 45 segundos.
- c) Tiempo pico ("peak time"): 90 segundos.
- d) Tiempo de estabilización ("settling time"): 180 segundos.
- e) Error en estado estacionario ("offset"): +/- 1% del valor deseado.

### **4.2.2 Obtención de la Función de Transferencia del Proceso**

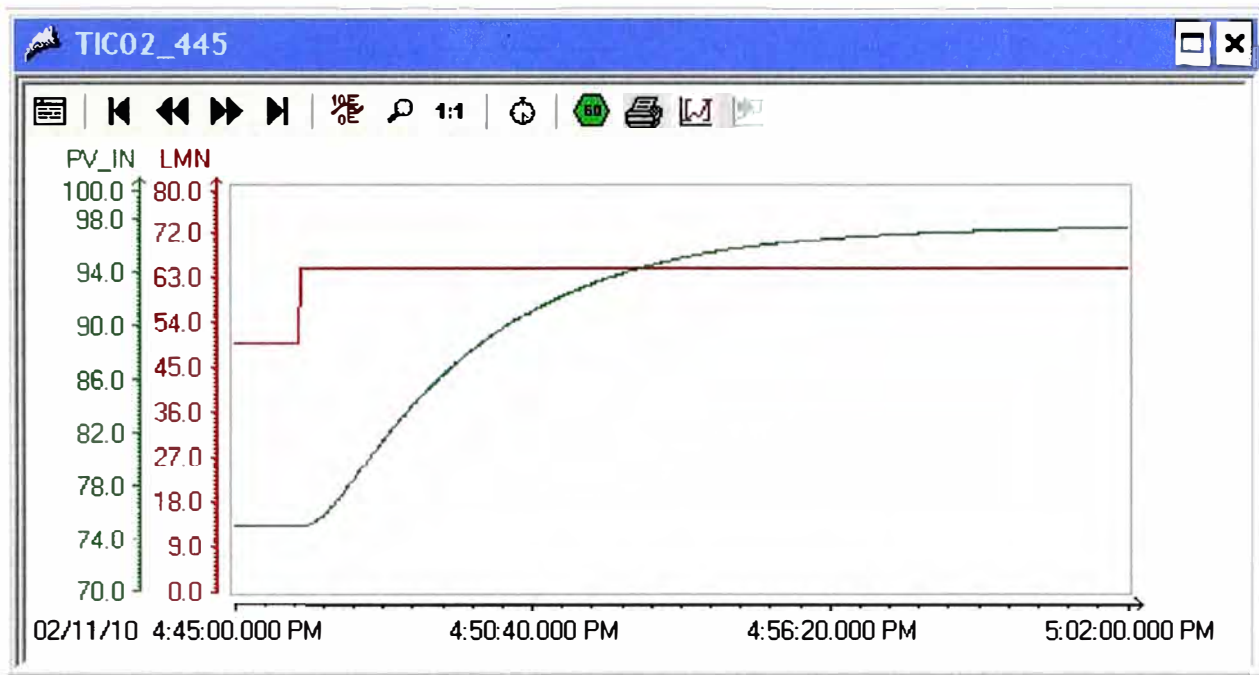
#### **Curva de reacción**

Para obtener la curva de reacción, aplicaremos una variación tipo escalón en la válvula proporcional de ingreso de vapor a la caldera, de 50% a 65%, estando el controlador en lazo abierto. La curva de tendencia obtenida se muestra en la Figura 4.10.

Los datos obtenidos se exportan hacia el software MATLAB mediante un modelo en SIMULINK para comunicación con un servidor OPC. El modelo empleado se encuentra en la sección A.2.1 del anexo (archivo "tic445opc.mdl"). Los datos obtenidos se grafican contra el tiempo en segundos (Figura 4.11 y Figura 4.12).

#### **Cálculo de parámetros para el modelo del proceso**

Para hallar los parámetros del modelo del proceso se seguirá el mismo procedimiento que en el caso anterior. El archivo MATLAB usado para dicho cálculo se encuentra en la sección A.2.2 del anexo ("tic445anl.m"). La curva de reacción con eliminación de ruido se muestra en la Figura 4.13. El punto de inflexión con el trazo de la recta tangente se muestra en la Figura 4.14.



**Figura 4.10: Curva de reacción para temperatura de leche en salida intercambiador**

Una vez hallado el punto de inflexión y la recta tangente a la curva de reacción en dicho punto, calculamos los valores de los parámetros " $T_U$ " y " $T_G$ ". Los valores obtenidos son los siguientes:

$$T_U = 19.35 \text{ seg}$$

$$T_G = 261.75 \text{ seg}$$

Y la ganancia del proceso sería:

$$K_p = \frac{\Delta Y}{\Delta u} = 1.4945$$

En este caso podemos apreciar que la relación de  $T_U$  a  $T_G$  es bastante pequeña:

$$\frac{T_U}{T_G} = 0.0739 \leq 0.104$$

Luego, podemos aplicar el caso especial para la función de transferencia usando un bloque tipo  $PT_n$  (Samal), es decir:

$$G_p(s) = \frac{K_p}{(1 + T_1 \cdot s) \cdot (1 + T_2 \cdot s)} \quad (4.7)$$

Por lo tanto, de la tabla 2.2 obtenemos los siguientes valores para  $K$ ,  $T_1$  y  $T_2$ :

$$K = 5$$

$$T_1 = 28.0072$$

$$T_2 = 140.0363$$

Finalmente, reemplazando los datos en (4.7) obtenemos la función de transferencia para el proceso de control de temperatura:

$$G_p(s) = \frac{1.4945}{(1 + 28.0072 \cdot s) \cdot (1 + 140.0363 \cdot s)} \quad (4.8)$$

#### 4.2.3 Cálculo de Parámetros para el Controlador

Para determinar los parámetros del controlador, usaremos esta vez el método de la curva de reacción descrito en la sección 3.2.3. Como se explicó en dicho capítulo, este método se realiza con una prueba a lazo abierto, a diferencia del método anterior que requiere una prueba a lazo cerrado, y se obtienen los parámetros de un análisis gráfico de la curva de reacción.

Como ya hemos obtenido los valores para  $K_p$ ,  $T_U$  y  $T_G$  simplemente usamos las ecuaciones 3.15, 3.16 y 3.17 para hallar los parámetros del controlador PID. Obtenemos los siguientes valores:

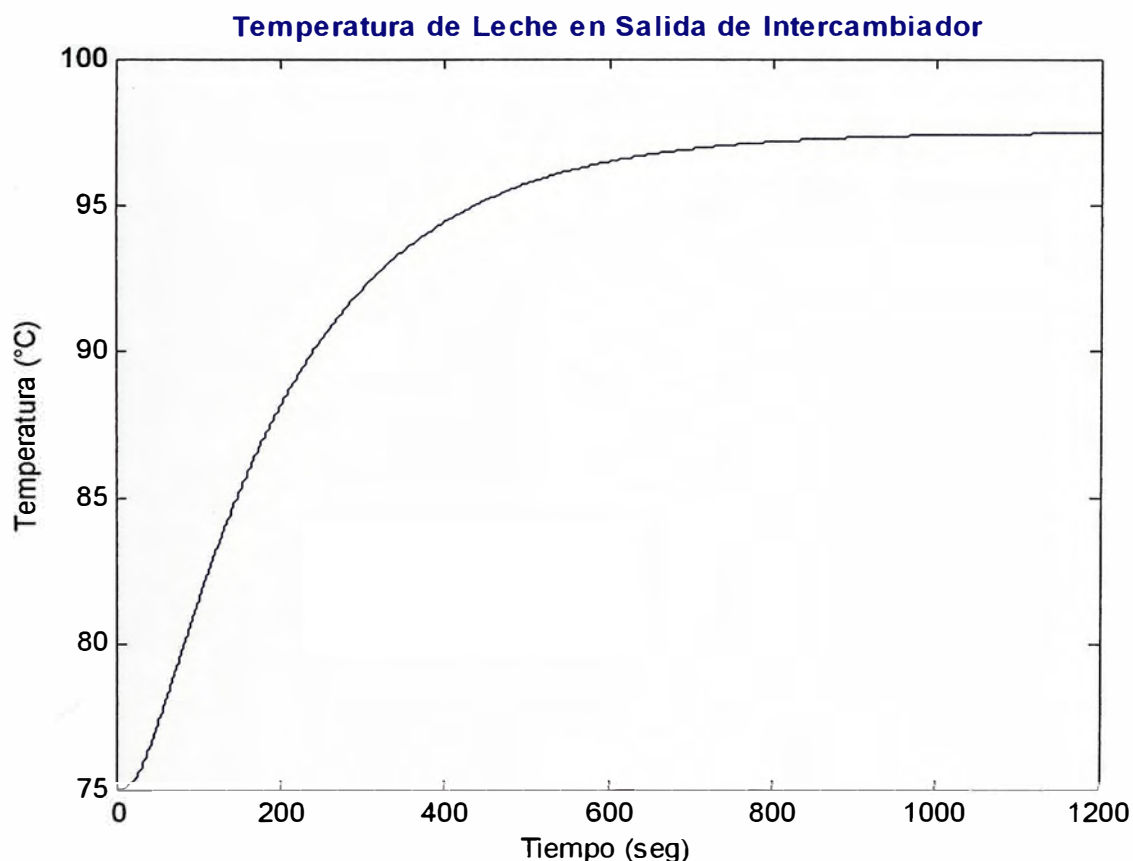
Ganancia proporcional:  $K_c = 10.8613$

Tiempo integral:  $T_i = 38.7 \text{ seg}$

Tiempo derivativo:  $T_d = 9.675 \text{ seg}$

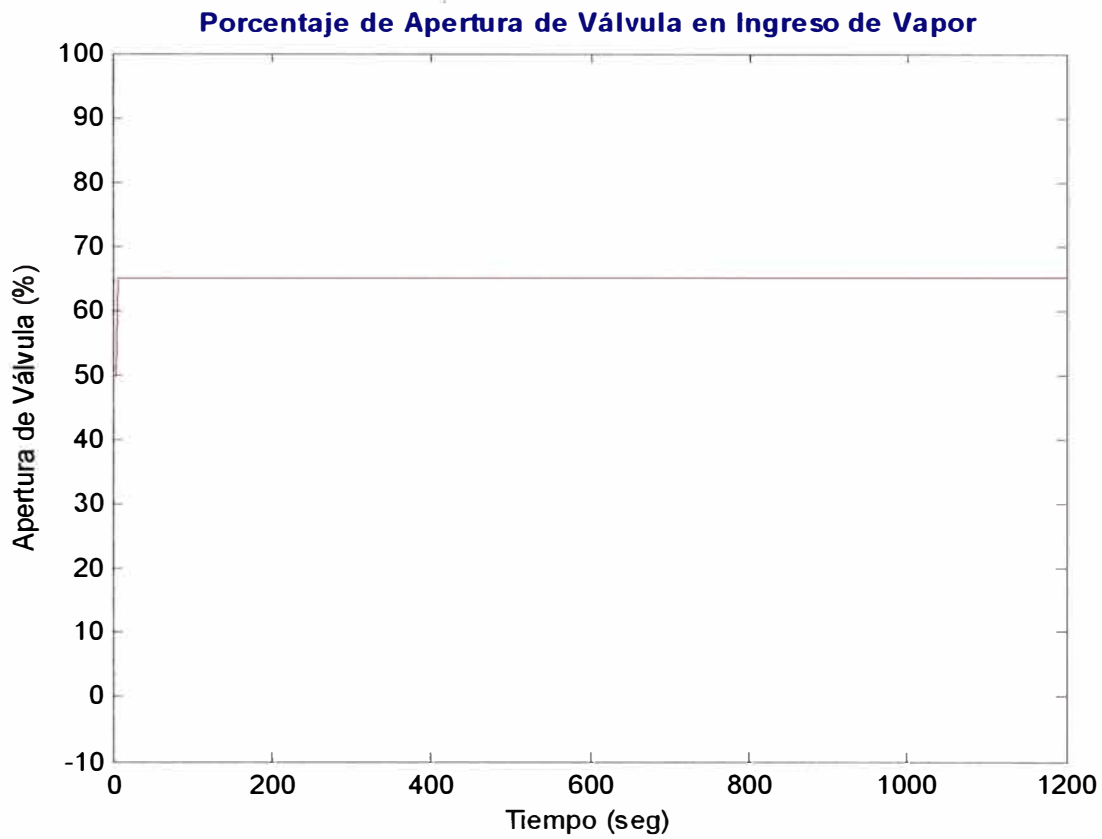
Y por lo tanto, la función de transferencia del controlador es la siguiente:

$$G_c(s) = 10.8613 \cdot \left(1 + \frac{1}{38.7 \cdot s} + 9.675 \cdot s\right) \quad (4.9)$$

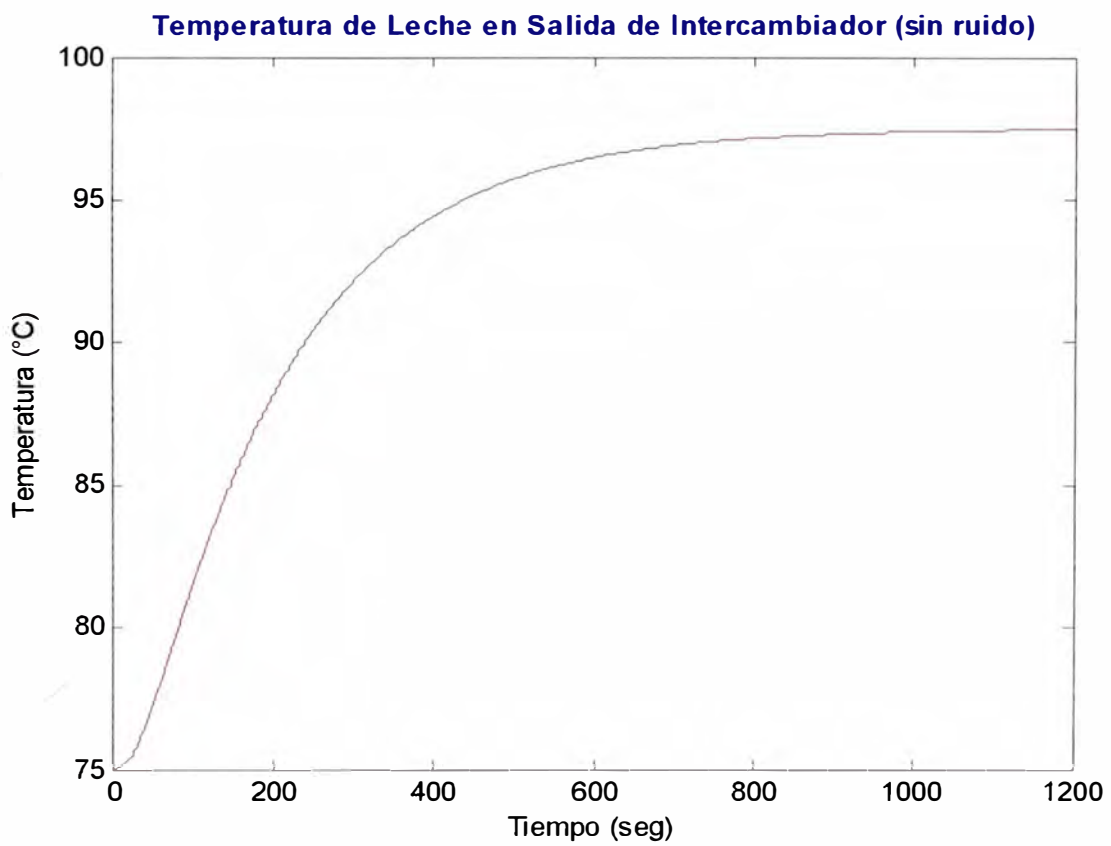


**Figura 4.11: Temperatura de producto en salida de intercambiador**

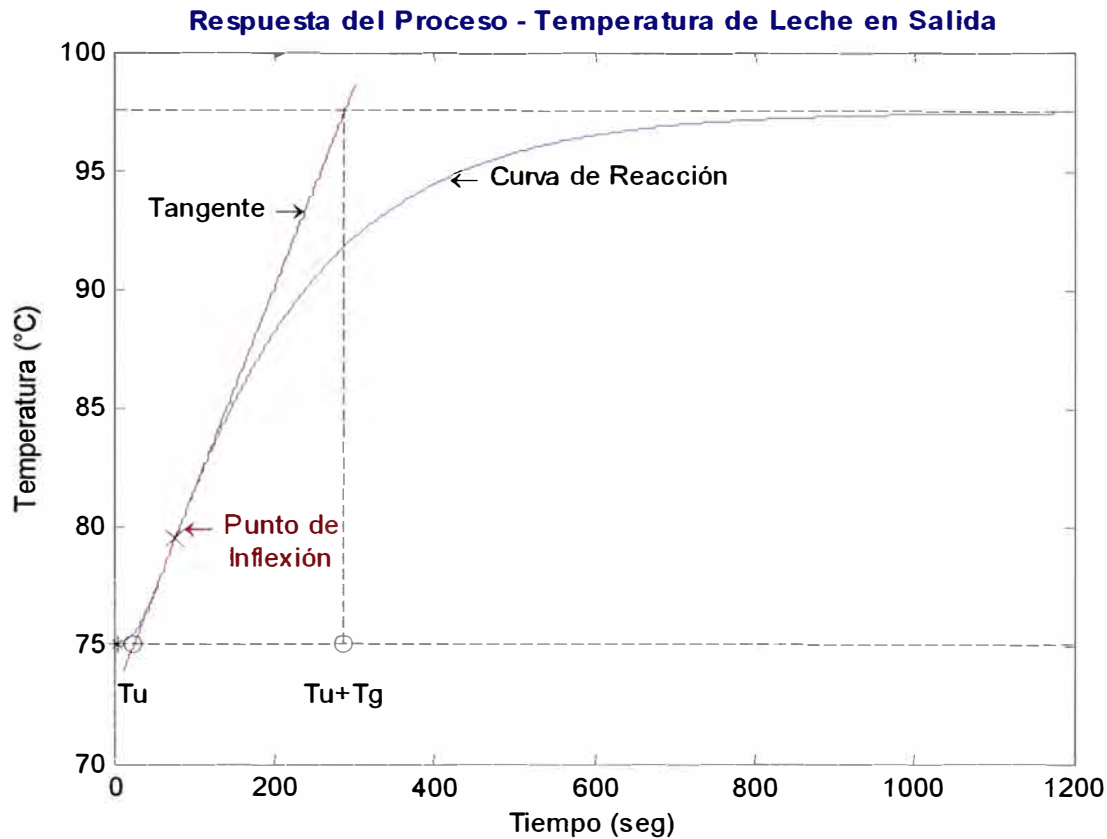




**Figura 4.12: Variación tipo escalón en apertura de válvula de ingreso de vapor**



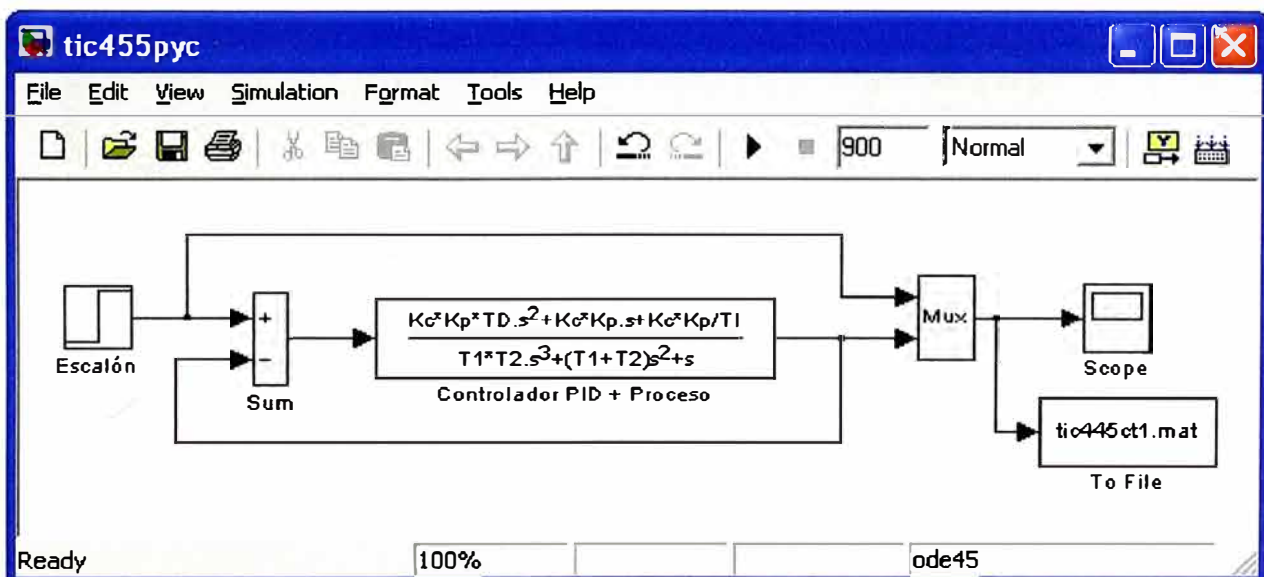
**Figura 4.13: Temperatura de producto en salida de intercambiador sin ruido**



**Figura 4.14: Determinación gráfica de los parámetros del proceso “ $T_u$ ” y “ $T_g$ ”**

#### 4.2.4 Análisis de la Acción del Controlador sobre el Proceso

Teniendo la función de transferencia del proceso (ecuación 4.8) y la del controlador (ecuación 4.9) implementamos en MATLAB un modelo de simulación para analizar la respuesta del proceso en lazo cerrado bajo la acción del controlador. Se aplica un cambio tipo escalón en el valor deseado de 70°C a 110°C. Este modelo de simulación se ilustra en la Figura 4.15.



**Figura 4.15: Modelo de proceso con acción de controlador PID en lazo cerrado**

El análisis de las características de la respuesta obtenida se realiza con el programa "tic445rsl.m" que se encuentra en la sección A.2.5 del anexo. Los valores obtenidos de las características de la respuesta inicial del proceso son:

- Porcentaje de sobre-nivel ("overshoot"): 31.09%.
- Tiempo de subida ("Tr"): 25.6 seg.
- Tiempo pico ("Tp"): 50.1 seg.
- Tiempo de estabilización (para error en estado estacionario de 2%): 205.1 seg.

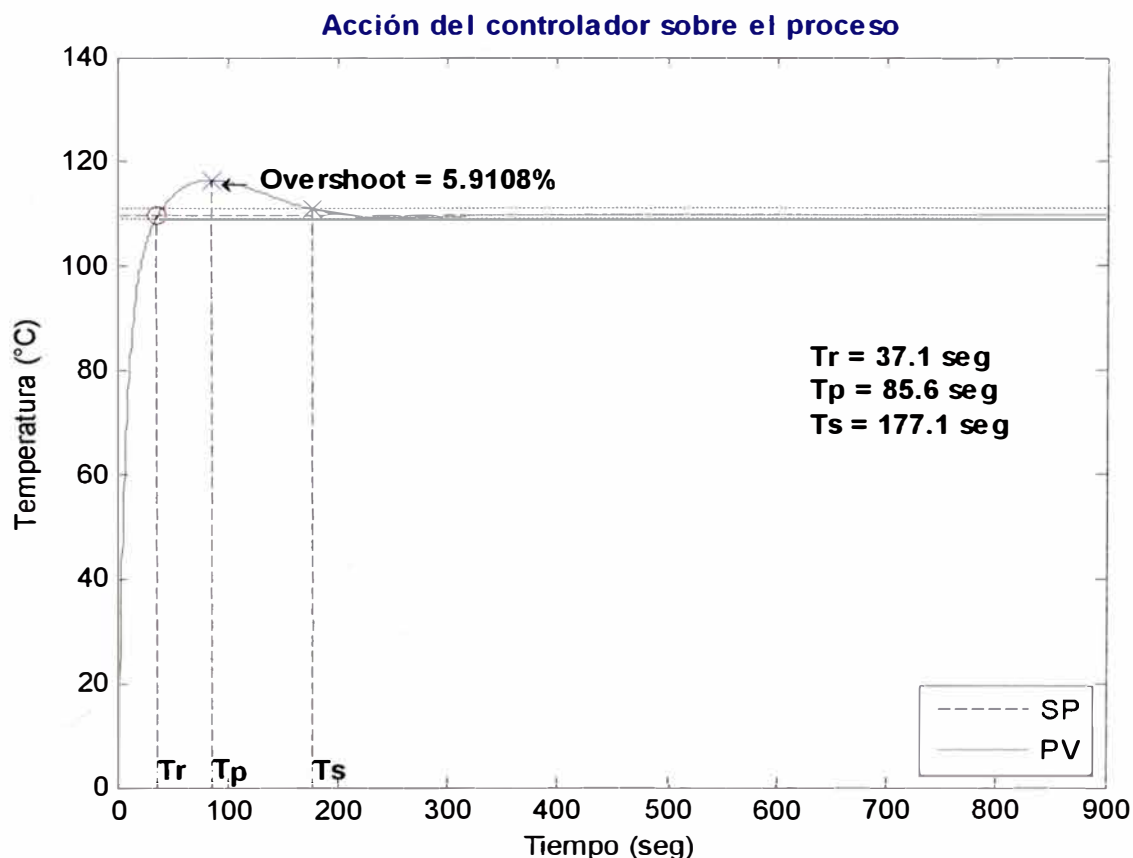
Es entonces necesario hacer un ajuste adicional por tanteo de los parámetros del controlador, para mejorar el porcentaje de sobre-nivel. Haciendo un ajuste manual adicional de los parámetros del controlador, se obtiene la respuesta final mostrada en la Figura 4.16. Esta respuesta se logra variando el tiempo integral y el tiempo derivativo adecuadamente. Los parámetros finales del controlador son los siguientes:

Ganancia proporcional:  $K_c = 10.8613$

Tiempo integral:  $T_i = 55.0 \text{ seg}$  (4.10)

Tiempo derivativo:  $T_d = 30.0 \text{ seg}$

De esta forma se cumple con las especificaciones de diseño planteadas inicialmente, con lo cual se termina el diseño del controlador.



**Figura 4.16: Respuesta final de la acción del controlador sobre el proceso**

### **4.3 Presión de Vapor en Termocompresor**

#### **4.3.1 Objetivos de Control y Especificaciones de Diseño**

El objetivo es mantener constante la presión de vapor en el termocompresor, para garantizar una mezcla correcta del vapor de caldera con el vapor obtenido de la evaporización en los efectos, siendo ésta mezcla la que finalmente sirve como medio de calentamiento en los efectos durante todo el proceso de evaporación. Se hace importante entonces mantener dicha presión en el valor de operación para un correcto proceso de evaporación. Por lo tanto, la variable a controlar en éste caso es la presión en el termocompresor. El valor deseado debe estar entre 4.5 a 5.0 bar. La variable manipulada es el porcentaje de apertura de una válvula proporcional que regula la cantidad de vapor de caldera que ingresa al termocompresor.

El controlador diseñado debe ser capaz de llevar la variable de proceso al valor deseado en un tiempo no muy largo. En este caso se puede admitir un porcentaje de sobre-nivel más elevado que en los casos anteriores, debido a que esto no es tan perjudicial para el proceso.

El controlador a usar será de tipo PI, que es el más adecuado para procesos de presión. Para cumplir con el objetivo de control, planteamos las siguientes condiciones de diseño para el controlador:

- a) Porcentaje de sobre-nivel (“overshoot”): 20%.
- b) Tiempo de subida (“rise time”): 40 segundos.
- c) Tiempo pico (“peak time”): 60 segundos.
- d) Tiempo de estabilización (“settling time”): 150 segundos.
- e) Error en estado estacionario (“offset”): +/- 2.5% del valor deseado.

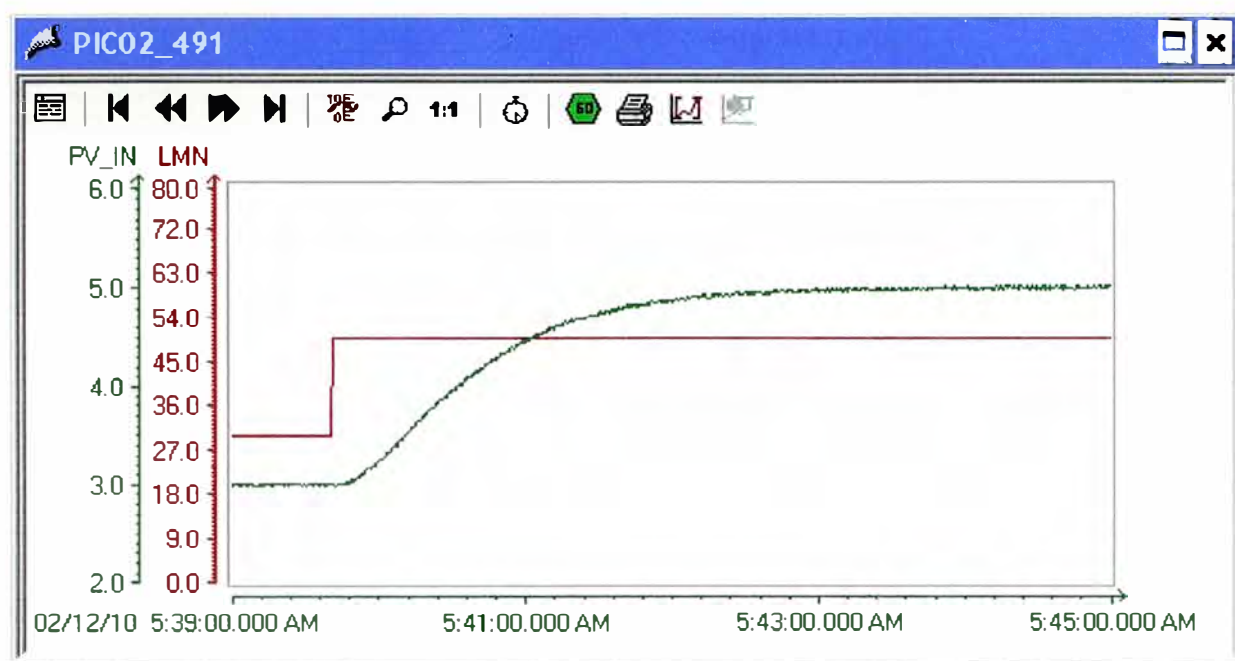
#### **4.3.2 Obtención de la Función de Transferencia del Proceso**

##### **Curva de reacción**

Para obtener la curva de reacción, aplicaremos una variación tipo escalón en la válvula proporcional de ingreso de vapor al termocompresor, desde un valor inicial de 30% hasta un valor final de 50%, estando el controlador en manual y en lazo abierto. La curva de tendencia obtenida en el software de supervisión y monitoreo (SCADA) se muestra en la Figura 4.17.

Los datos obtenidos se exportan hacia el software MATLAB mediante un modelo en SIMULINK para comunicación con un servidor OPC. Dicho modelo se encuentra en la sección A.3.1 en los anexos con el nombre “pic491opc.mdl”.

Los datos obtenidos para la curva de reacción y la variación en el porcentaje de apertura de la válvula en el ingreso de vapor se grafican contra el tiempo en segundos en la Figura 4.18 y Figura 4.19, respectivamente.



**Figura 4.17: Curva de reacción para presión de vapor en termocompresor**

### Cálculo de parámetros para el modelo del proceso

Se seguirá el mismo procedimiento que en el caso del lazo de control anterior. El archivo usado para el cálculo del modelo del proceso se encuentra en la sección A.3.2 del anexo bajo el nombre de "pic491anl.m".

La curva de reacción con eliminación de ruido se muestra en la Figura 4.20. El punto de inflexión con el trazo de la recta tangente en dicho punto se muestra en la Figura 4.21. Una vez hallado el punto de inflexión y la recta tangente a la curva de reacción en dicho punto, podemos calcular los valores de los parámetros " $T_U$ " y " $T_G$ ". Los valores obtenidos son los siguientes:

$$T_U = 8.7 \text{ seg}$$

$$T_G = 85.2 \text{ seg}$$

Y la ganancia del proceso sería:

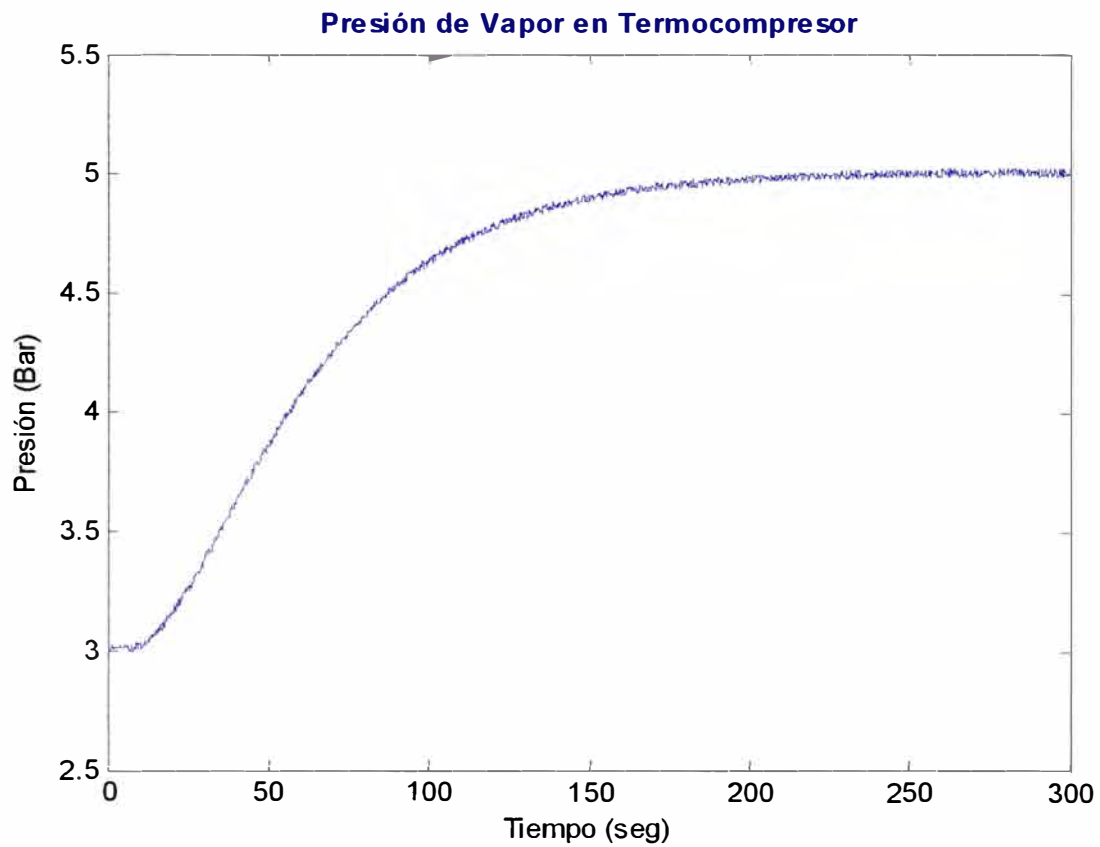
$$K_p = \frac{\Delta Y}{\Delta u} = 0.099$$

Con lo cual, si usamos un bloque tipo  $PT_n$ , según la ecuación 2.3 y la tabla 2.1, hallamos los siguientes parámetros para el modelo del proceso:

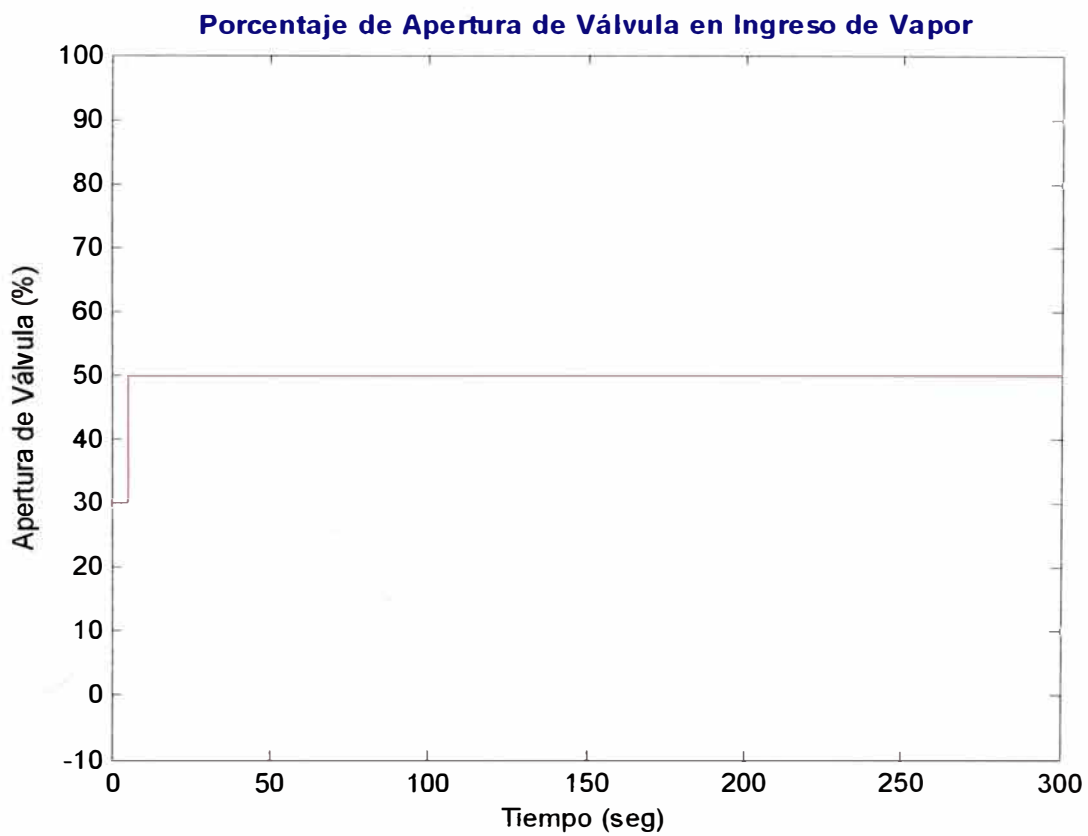
$$\frac{T_U}{T_G} = 0.1021 \Rightarrow n = 2$$

$$T = 0.368 \cdot T_G = 31.3536 \text{ seg.}$$

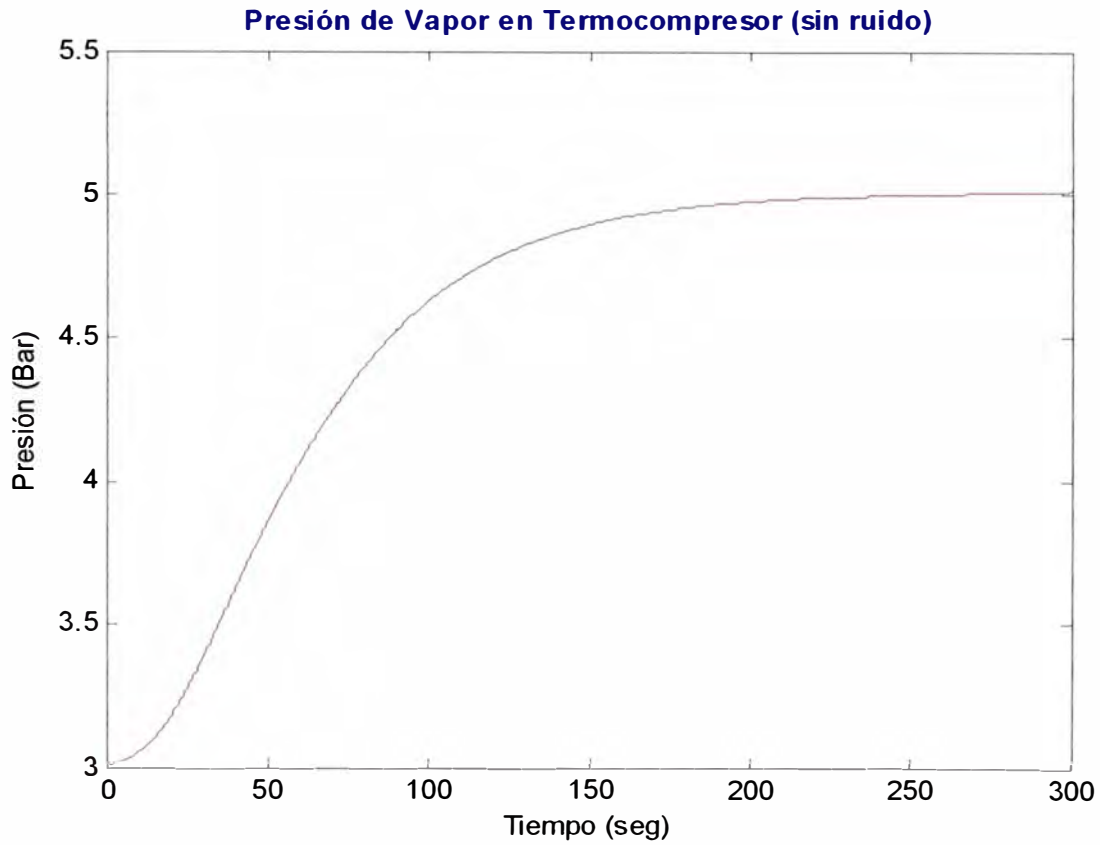
$$G_p(s) = \frac{0.099}{(1 + 31.3536 \cdot s)^2} \quad (4.11)$$



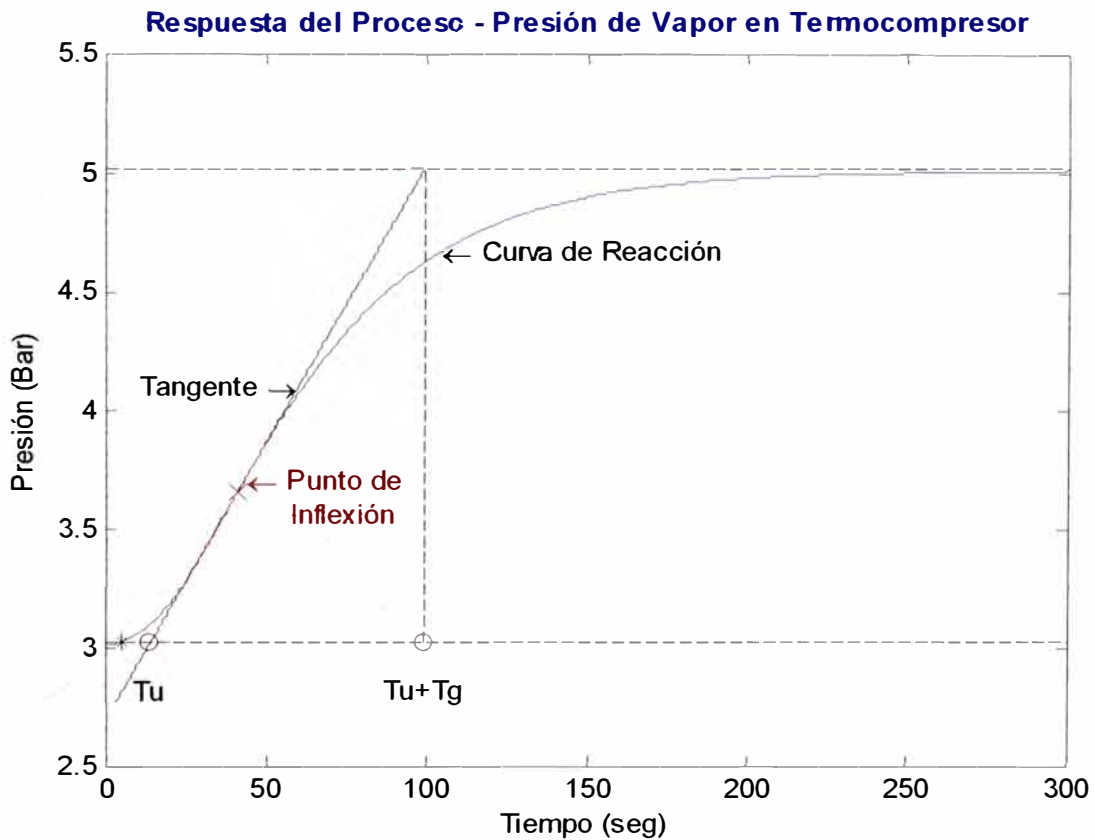
**Figura 4.18: Presión de vapor en Termocompresor**



**Figura 4.19: Variación tipo escalón en apertura de válvula de ingreso de vapor**



**Figura 4.20: Presión de vapor en Termocompresor sin ruido**



**Figura 4.21: Determinación gráfica de los parámetros del proceso “ $T_u$ ” y “ $T_g$ ”**

### 4.3.3 Cálculo de Parámetros para el Controlador

Para determinar los parámetros del controlador, usaremos nuevamente el método de la curva de reacción descrito en el capítulo 3.2.3. Como ya hemos obtenido los valores para  $K_P$ ,  $T_U$  y  $T_G$  simplemente usamos las ecuaciones 3.13 y 3.14 para hallar los parámetros del controlador PI:

Ganancia proporcional:  $K_C = 88.1834$

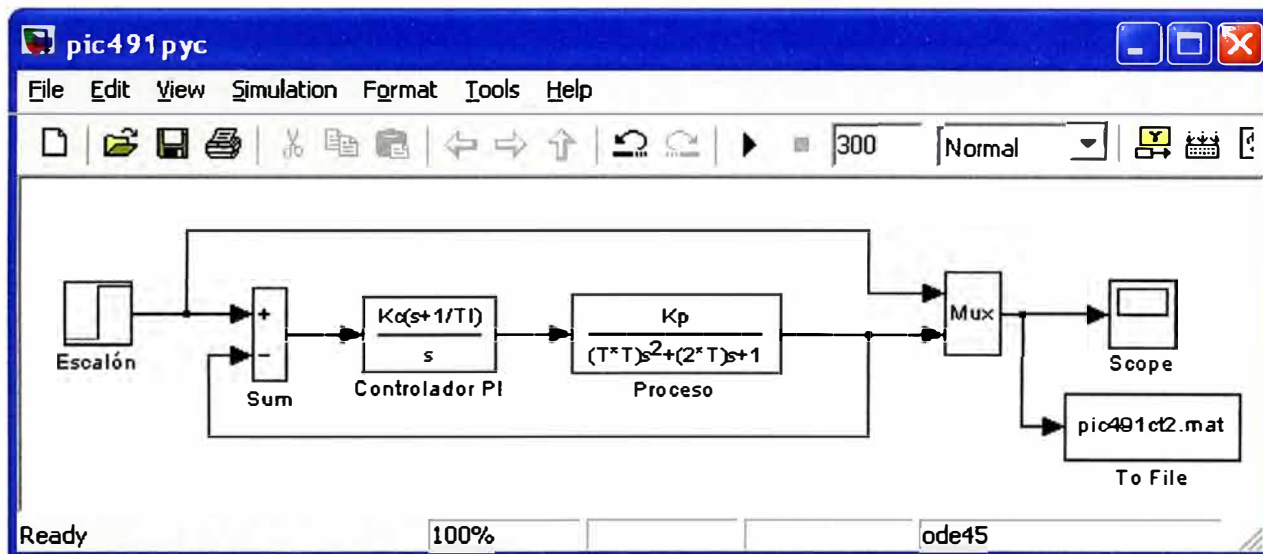
Tiempo integral:  $T_I = 28.971 \text{ seg}$

Y por lo tanto, la función de transferencia del controlador es la siguiente:

$$G_C(s) = 88.1834 \cdot \left(1 + \frac{1}{28.971 \cdot s}\right) \quad (4.12)$$

### 4.3.4 Análisis de la Acción del Controlador sobre el Proceso

Teniendo la función de transferencia del proceso (ecuación 4.11) y la del controlador (ecuación 4.12) implementamos en MATLAB un modelo de simulación para analizar la respuesta del proceso en lazo cerrado bajo la acción del controlador, ante un cambio tipo escalón en el valor deseado de 4 bar a 5 bar. Este modelo de simulación se ilustra en la Figura 4.22.



**Figura 4.22: Modelo de proceso con acción de controlador PID en lazo cerrado**

El análisis de las características de la respuesta obtenida se realiza en el archivo MATLAB "pic491rsl.m" que se encuentra en la sección A.3.5 del anexo. De esta forma, los valores obtenidos de las características de la respuesta inicial del proceso ante la acción del controlador son los siguientes:

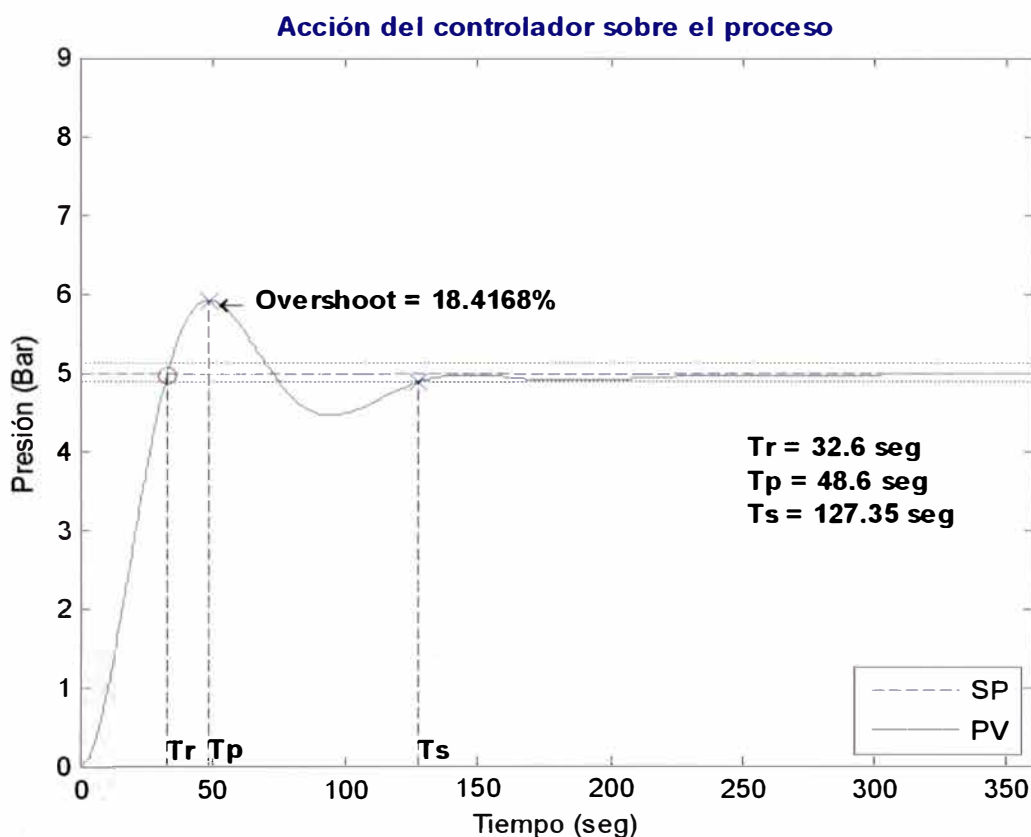
- Porcentaje de sobre-nivel ("overshoot"): 61.08%.
- Tiempo de subida ("Tr"): 18.35 seg.
- Tiempo pico ("Tp"): 33.6 seg.
- Tiempo de estabilización (para error en estado estacionario de 2.5%): 308.35 seg.



En este caso, el controlador con los parámetros iniciales no cumple las condiciones de diseño en cuanto a porcentaje de sobre-nivel y tiempo de estabilización. Es necesario hacer un ajuste adicional por tanteo de los parámetros del controlador, obteniéndose los siguientes valores para la respuesta final mejorada:

$$K_C = 45 \text{ y } T_I = 90.5 \text{ seg} \quad (4.13)$$

La gráfica de la respuesta mejorada se muestra en la Figura 4.23.



**Figura 4.23: Mejora de la acción del controlador sobre el proceso**

De esta forma se cumple con las especificaciones de diseño planteadas inicialmente, con lo cual se termina el diseño del controlador.

## CONCLUSIONES Y RECOMENDACIONES

### Conclusiones:

En el presente informe se han mostrado tres métodos de diseño para controladores PID: el método de ganancia límite, el método de mínimo rebasamiento del punto de consigna y el método de la curva de reacción. El uso de cada método depende básicamente de las características que tenga el proceso que se requiere controlar, en especial, el tipo de variable a controlar y las especificaciones de diseño que debe cumplir el proceso controlado. Se han analizado las respuestas de tres clases principales de procesos: control de flujo, control de temperatura y control de presión, que son las variables principales que intervienen para un funcionamiento correcto del evaporador de efectos múltiples: flujo de leche en el ingreso al primer efecto, temperatura de leche en el ingreso al primer efecto y presión de vapor en el termocompresor. Se observó en las Figuras 4.1, Figura 4.10 y Figura 4.17 que cada uno de dichos procesos presenta una respuesta diferente ante los cambios escalón en la señal de entrada. Y efectivamente, esto se comprobó posteriormente al calcular los parámetros de los procesos, tal como explicamos a continuación.

En el caso del controlador de flujo de leche en ingreso al primer efecto, se observa en la Figura 4.2.a que la respuesta ante la variación tipo escalón de la señal de entrada (Figura 4.2.b) provoca un cambio rápido en la variable de salida. Al realizar el análisis de la curva de reacción, se obtuvo una relación de los parámetros del proceso  $T_U/T_G$  igual a 0.1247, lo cual nos indica que es un proceso de respuesta rápida. Como modelo matemático para este proceso se usó un bloque tipo  $PT_n$ . Se utilizó el método de diseño de mínimo rebasamiento del punto de consigna para el cálculo del controlador PID, para poder cumplir con las condiciones de diseño planteadas de mínimo porcentaje de sobre-nivel y corto tiempo de estabilización. Los parámetros del controlador hallados con dicho método fueron los siguientes: ganancia proporcional: 0.032, tiempo integral: 22.92seg. y tiempo derivativo: 7.258seg. El controlador diseñado proporciona una respuesta inicial aceptable ante un cambio escalón del punto de consigna de 25,000lt/h a 32,000lt/h, con los siguientes resultados: porcentaje de sobre-nivel: 12.68%, tiempo de estabilización: 39.35seg y error en estado estacionario casi nulo. Luego, simplemente se realizaron

ajustes adicionales para disminuir el sobre-impulso y cumplir así con todas las condiciones de diseño. Solo fue necesario aumentar ligeramente el tiempo derivativo. El nuevo valor de éste parámetro es: 9seg. y la respuesta final se aprecia en la figura 4.9. Los parámetros finales de la respuesta del proceso bajo la acción del controlador son los siguientes: porcentaje de sobre-nivel: 8.549%, tiempo de estabilización: 43.84seg. y error en estado estacionario nulo. Con ésta respuesta final se cumplen todas las condiciones de diseño planteadas.

Para el lazo de control de temperatura de leche en ingreso al primer efecto, se observa en la Figura 4.11 que la respuesta del proceso ante la variación tipo escalón de la señal de entrada (Figura 4.12) es más lenta, lo cual es de esperarse por ser un proceso térmico de intercambio de calor. Al realizar el análisis de la curva de reacción, se obtuvo una relación de los parámetros del proceso  $T_U/T_G$  igual a 0.0739, lo cual nos indica que es un proceso de alta resistencia. Como modelo matemático para este proceso se usó un bloque tipo  $PT_n$ , con el caso especial para procesos con valores bajos de la relación  $T_U/T_G$ . Para el diseño del controlador PID se utilizó el método de la curva de reacción. Los parámetros iniciales del controlador hallados con dicho método fueron los siguientes: ganancia proporcional: 10.86, tiempo integral: 38.7seg. y tiempo derivativo: 9.675seg. El controlador diseñado proporciona una respuesta inicial aceptable ante un cambio escalón del punto de consigna de 70°C a 110°C, con los siguientes resultados: porcentaje de sobre-nivel: 31.09%, tiempo de estabilización: 205.1seg y error en estado estacionario casi nulo. Por lo tanto, se tuvieron que realizar ajustes adicionales para disminuir el sobre-impulso y cumplir así con todas las condiciones de diseño. En éste caso se modificaron los tiempos integral y derivativo adecuadamente hasta alcanzar una respuesta mejorada. Los nuevos valores para dichos parámetros son tiempo integral: 55seg. y tiempo derivativo: 30seg. La respuesta final se aprecia en la figura 4.16, donde se puede apreciar que los parámetros finales de la respuesta del proceso son los siguientes: porcentaje de sobre-nivel: 5.91%, tiempo de estabilización: 177.1seg y error en estado estacionario nulo. Con ésta respuesta final se cumplen todas las condiciones de diseño.

En el controlador de presión de vapor en el termocompresor la respuesta del proceso ante una variación tipo escalón en la señal de entrada (Figura 4.18) presenta un comportamiento intermedio entre los procesos de flujo y temperatura. Al realizar el análisis de la curva de reacción, se obtuvo una relación de los parámetros del proceso  $T_U/T_G$  igual a 0.1021, lo cual nos indica que es un proceso de resistencia media. Como modelo matemático para este proceso se usó nuevamente un bloque tipo  $PT_n$ . En éste caso se empleó un controlador tipo PI, debido a que el proceso no presenta cambios

bruscos ni cambios demasiado lentos. Para el diseño del controlador se utilizó el método de la curva de reacción. Los parámetros iniciales del controlador hallados con dicho método fueron los siguientes: ganancia proporcional: 88.18 y tiempo integral: 28.971seg. El controlador diseñado no proporciona una respuesta inicial aceptable para las condiciones de diseño. En efecto, ante un cambio escalón del punto de consigna de 4bar a 5bar se obtuvieron los siguientes resultados: porcentaje de sobre-nivel: 61.08%, tiempo de estabilización: 308.351seg y error en estado estacionario casi nulo. Por lo tanto, se tuvieron que realizar ajustes adicionales para disminuir el sobre-impulso y acortar el tiempo de estabilización para cumplir así con las condiciones de diseño. En éste caso se modificó la ganancia proporcional y el tiempo integral adecuadamente hasta alcanzar una respuesta aceptable. Los valores finales de dichos parámetros son: ganancia: 45 y tiempo integral: 90.5seg. La respuesta final se aprecia en la figura 4.23, donde se puede apreciar que los parámetros finales de la respuesta del proceso son los siguientes: porcentaje de sobre-nivel: 18.41%, tiempo de estabilización: 127.35seg y error en estado estacionario nulo. Con ésta respuesta final se cumplen todas las condiciones de diseño.

#### **Recomendaciones:**

De los métodos estudiados para hallar la función de transferencia del proceso se recomienda usar el modelo con bloque tipo  $PT_n$  (ecuación 2.3) en lugar del modelo con bloque tipo  $PT_1T_t$  (ecuación 2.1), ya que el primero proporciona resultados más reales en la simulación del comportamiento del proceso. Esto se comprobó al realizar reiteradas simulaciones usando la herramienta SIMULINK del programa MATLAB.

En cuanto a los métodos de diseño para los controladores PID, se recomienda usar el método de mínimo rebasamiento del punto de consigna, ya que es el que mejor resultados proporciona, dando siempre una respuesta inicial con mínimo porcentaje de sobre-nivel, corto tiempo de estabilización y error en estado estacionario nulo. Hay que tener en cuenta que para aplicar este método es necesario realizar una prueba a lazo cerrado del sistema, y llevarlo a condiciones de oscilación (inestabilidad) para obtener los parámetros del controlador. En algunos casos, dicho procedimiento no se puede realizar en una planta en producción, debido a que se podría dañar algún equipo o el producto en procesamiento, y por lo tanto, habría que recurrir a otros métodos de ajuste, como por ejemplo, el método de la curva de reacción. Para cualquiera de los métodos usados, siempre es necesario realizar un ajuste adicional por tanteo, para optimizar el control y cumplir con todas las condiciones de diseño.

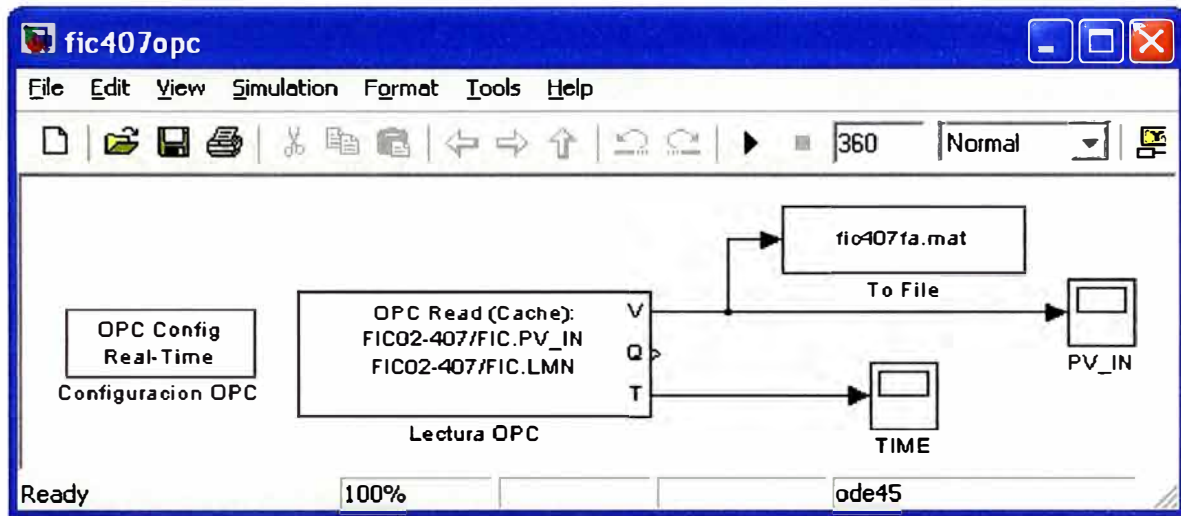
Los lazos de control estudiados en el presente informe fueron implementados en un sistema automatizado con PLC y software SCADA, y se encuentra operativo en la actualidad. Esto demuestra que los controladores PID siguen siendo ampliamente usados

en la automatización de procesos industriales, debido a su buen desempeño, precisión aceptable y simplicidad.

**ANEXO**  
**LISTADO DE PROGRAMAS EN MATLAB Y SIMULINK**

## A.1) Modelos y programas para controlador de flujo

### A.1.1) Modelo para comunicación con servidor OPC



### A.1.2) Programa para análisis de datos y obtención de parámetros del proceso

% fic407anl.m: Análisis de datos obtenidos de FIC02-407 por OPC Server WinCC

```
clc;
```

```
% Calculo de vectores LMN y PV_IN
```

```
PID_TIME = fic407va(1,:);
```

```
PID_PVIN = fic407va(2,:);
```

```
PID_LMN = fic407va(3,:);
```

```
% parametros varios
```

```
n = max(size(fic407va));
```

```
PID_PVIN_SM = fastsmooth2(PID_PVIN,100,3,1);
```

```
% Cálculo de la 1era. derivada
```

```
d1_PVIN = zeros(1,n);
```

```
for i=1:n-1
```

```
    d1_PVIN(1,i) = PID_PVIN_SM(1,i+1)-PID_PVIN_SM(1,i);
```

```
end;
```

```
d1_PVIN(1,n) = d1_PVIN(1,n-1);
```

```
% eliminación de ruido
```

```
d1_PVIN_SM = fastsmooth2(d1_PVIN,100,3,1);
```

```
% Cálculo de la 2da. derivada
```

```
d2_PVIN = zeros(1,n);
```

```
for i=1:n-1
```

```
    d2_PVIN(1,i) = d1_PVIN_SM(1,i+1)-d1_PVIN_SM(1,i);
```

```
end;
```

```
d2_PVIN(1,n) = d2_PVIN(1,n-1);
```

```

figure;
plot(PID_TIME,d2_PVIN);
% eliminación de ruido
d2_PVIN_SM = fastsmooth2(d2_PVIN,90,3,1);
% Búsqueda del punto de inflección en 2da. derivada = 0
for i=1:n-1
    if (d2_PVIN_SM(1,i)>=0) && (d2_PVIN_SM(1,i+1)<=0) && (i>20)
        i_pf1 = i;
        break;
    else
        i_pf1 = -1;
    end;
end;
% Búsqueda del punto de inflección en 1ra. derivada = max
for i=1:n-1
    if (d1_PVIN_SM(1,i)==max(d1_PVIN_SM))
        i_pf2 = i;
        break;
    else
        i_pf2 = -1;
    end;
end;
i_pf = i_pf2;
% Recta tangente en el punto de inflexión
fcorr = 1.05; % factor de corrección
Y_tg = fcorr*d1_PVIN_SM(1,i_pf)*(PID_TIME-PID_TIME(1,i_pf))/(PID_TIME(1,i_pf+1)-
PID_TIME(1,i_pf))+PID_PVIN_SM(1,i_pf);
% Cálculo de Tu y Tg
for i=1:n-1
    if ((PID_LMN(i+1)-PID_LMN(i))>0)
        i_Tinit = i;
        break;
    else
        i_Tinit = -1;
    end;
end;
for i=1:n-1
    if (Y_tg(1,i)<=PID_PVIN_SM(1,i_Tinit)) && (Y_tg(1,i+1)>=PID_PVIN_SM(1,i_Tinit))
        i_Tu = i;
    end;
end;

```



```

        break;
    else
        i_Tu = -1;
    end;
end;
for i=1:n-1
    if (Y_tg(1,i)>=max(PID_PVIN_SM))
        i_Tg = i;
        break;
    else
        i_Tg = -1;
    end;
end;
Tu = PID_TIME(1,i_Tu)-PID_TIME(1,i_Tinit);
Tg = PID_TIME(1,i_Tg)-PID_TIME(1,i_Tu);
% Cálculo de parámetros del proceso
Kp = (max(PID_PVIN_SM)-min(PID_PVIN_SM))/(max(PID_LMN)-min(PID_LMN));
T = 0.368*Tg;
%Gráficos
figure;
plot(PID_TIME,PID_PVIN);
xlabel('Tiempo (seg)');
ylabel('Flujo (lt/h)');
title('Flujo de Salida Tanque de Mantenimiento','Color','b',...
    'FontWeight','bold');
figure;
plot(PID_TIME,PID_LMN,'r');
axis([PID_TIME(1,1) PID_TIME(1,n) -10 100]);
xlabel('Tiempo (seg)');
ylabel('Referencia de Velocidad (%)');
title('Referencia de Velocidad de Variador de Bomba de Descarga','Color','b',...
    'FontWeight','bold');
figure;
plot(PID_TIME,PID_PVIN_SM,'r');
xlabel('Tiempo (seg)');
ylabel('Flujo (lt/h)');
title('Flujo de Salida Tanque de Mantenimiento (sin ruido)','Color','b',...
    'FontWeight','bold');
figure;

```

```

plot(PID_TIME,d1_PVIN,'b');
xlabel('Tiempo (seg)');
ylabel('Derivada de Flujo ((lt/h)/s)');
title('Primera Derivada del Flujo Salida Tanque Mantenimiento','Color','b',...
      'FontWeight','bold');
figure;
plot(PID_TIME,d1_PVIN_SM,'r');
xlabel('Tiempo (seg)');
ylabel('Derivada de Flujo ((lt/h)/s)');
title('Primera Derivada del Flujo Salida Tanque Mantenimiento "Suavizada"', 'Color','b',...
      'FontWeight','bold');
hold on;
plot(PID_TIME(1,i_pf),d1_PVIN_SM(1,i_pf),'x','MarkerEdgeColor','b',...
      'MarkerSize',10);
text(PID_TIME(1,i_pf),d1_PVIN_SM(1,i_pf),'\leftarrow Máximo valor',...
      'HorizontalAlignment','left');
figure;
plot(PID_TIME,PID_PVIN_SM);
xlabel('Tiempo (seg)');
ylabel('Flujo (lt/h)');
title('Respuesta del Proceso - Flujo de Salida Tanque','Color','b',...
      'FontWeight','bold');
text(PID_TIME(1,round(30*n/100)),PID_PVIN_SM(1,round(30*n/100)),'\leftarrow Curva de
Reacción',...
      'HorizontalAlignment','left');
str1(1) ={' \leftarrow Punto de'};
str1(2) ={'  Inflexión'};
text(PID_TIME(1,i_pf+1),PID_PVIN_SM(1,i_pf-4),str1,...
      'HorizontalAlignment','left','Color','r',...
      'VerticalAlignment','Middle');
hold on;
plot(PID_TIME(1,i_pf),PID_PVIN_SM(1,i_pf),'x','MarkerEdgeColor','k',...
      'MarkerSize',10);
hold on;
plot(PID_TIME(1,round(1*n/100):round(22*n/100)),Y_tg(1,round(1*n/100):round(22*n/100)), 'r');
text(PID_TIME(1,round(20*n/100)),Y_tg(1,round(20*n/100)), 'Tangente \rightarrow',...
      'HorizontalAlignment','right');
hold on;
plot([PID_TIME(1,1) PID_TIME(1,n)],[PID_PVIN_SM(1,i_Tinit) PID_PVIN_SM(1,i_Tinit)], '--k');
hold on;

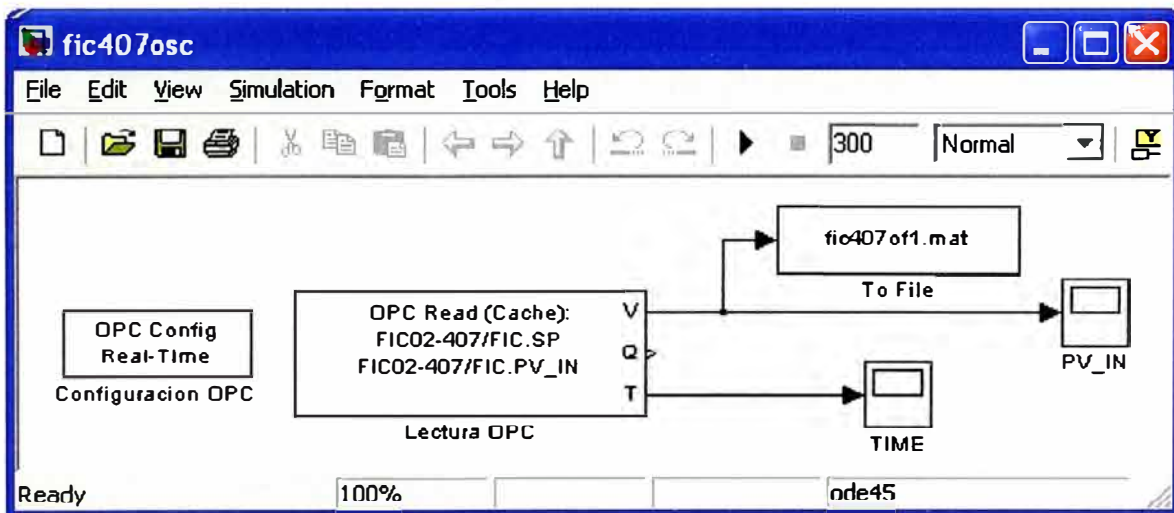
```

```

plot([PID_TIME(1,1) PID_TIME(1,n)],[max(PID_PVIN_SM) max(PID_PVIN_SM)],'--k');
hold on;
plot([PID_TIME(1,i_Tg) PID_TIME(1,i_Tg)],[PID_PVIN_SM(1,i_Tinit) max(PID_PVIN_SM)],'--k');
hold on;
plot(PID_TIME(1,i_Tinit),PID_PVIN_SM(1,i_Tinit),'*k');
hold on;
plot(PID_TIME(1,i_Tu),PID_PVIN_SM(1,i_Tinit),'ok');
hold on;
plot(PID_TIME(1,i_Tg),PID_PVIN_SM(1,i_Tinit),'ok');
str2(1) ={' '};
str2(2) ={'Tu'};
text(PID_TIME(1,i_Tu),PID_PVIN_SM(1,i_Tinit),str2,...
    'HorizontalAlignment','center','Color','k',...
    'VerticalAlignment','Top');
str3(1) ={' '};
str3(2) ={'Tu+Tg'};
text(PID_TIME(1,i_Tg),PID_PVIN_SM(1,i_Tinit),str3,...
    'HorizontalAlignment','center','Color','k',...
    'VerticalAlignment','Top');

```

### A.1.3) Modelo para Obtener respuesta oscilante de proceso por comunicación OPC



### A.1.4) Programa para obtención de ganancia límite y periodo de oscilación

```

% fic407mgl.m: Método de ganancia límite para hallar controlador
clc;
% Experimentalmente
Kcu = 0.04;
% Calculo de vectores LMN y PV_IN
PID_TIME = fic407ov1(1,:);

```

```

PID_SP = fic407ov1(2,:);
PID_PVIN = fic407ov1(3,:);
% parametros varios
n = max(size(fic407ov1));
PID_PVIN_SM = fastsmooth2(PID_PVIN,30,3,1);
% Búsqueda del periodo de oscilación
j = 0;
MPu = zeros(1,10);
for i=2:n-1
    if (PID_PVIN_SM(1,i)-PID_PVIN_SM(1,i-1)>=0) && (PID_PVIN_SM(1,i)-
PID_PVIN_SM(1,i+1)>=0)
        j = j+1;
        MPu(1,j) = i;
        if (j==10)
            break;
        end;
    end;
end;
end;
% Cálculo de Pcu
Pcu = PID_TIME(1,MPu(1,5))-PID_TIME(1,MPu(1,4));
%Gráficos
figure;
plot(PID_TIME,PID_PVIN);
xlabel('Tiempo (seg)');
ylabel('Flujo (lt/h)');
title('Flujo de Salida Tanque de Mantenimiento','Color','b','FontWeight','bold');
figure;
plot(PID_TIME,PID_PVIN_SM,'r');
xlabel('Tiempo (seg)');
ylabel('Flujo (lt/h)');
title('Flujo de Salida Tanque de Mantenimiento (sin ruido)','Color','b','FontWeight','bold');
hold on;
plot(PID_TIME(1,MPu(1,4)),PID_PVIN_SM(1,MPu(1,4)),'x','MarkerEdgeColor','b','MarkerSize',10);
hold on;
plot(PID_TIME(1,MPu(1,5)),PID_PVIN_SM(1,MPu(1,5)),'x','MarkerEdgeColor','b','MarkerSize',10);
hold on;
plot([PID_TIME(1,MPu(1,4)) PID_TIME(1,MPu(1,5))],[PID_PVIN_SM(1,MPu(1,4)) ...
PID_PVIN_SM(1,MPu(1,5))],':k');
txt1 = {'Pcu = ' num2str(Pcu) ' seg'};
text(PID_TIME(1,round((MPu(1,4)+MPu(1,5))/2)),PID_PVIN_SM(1,MPu(1,4)),txt1,...

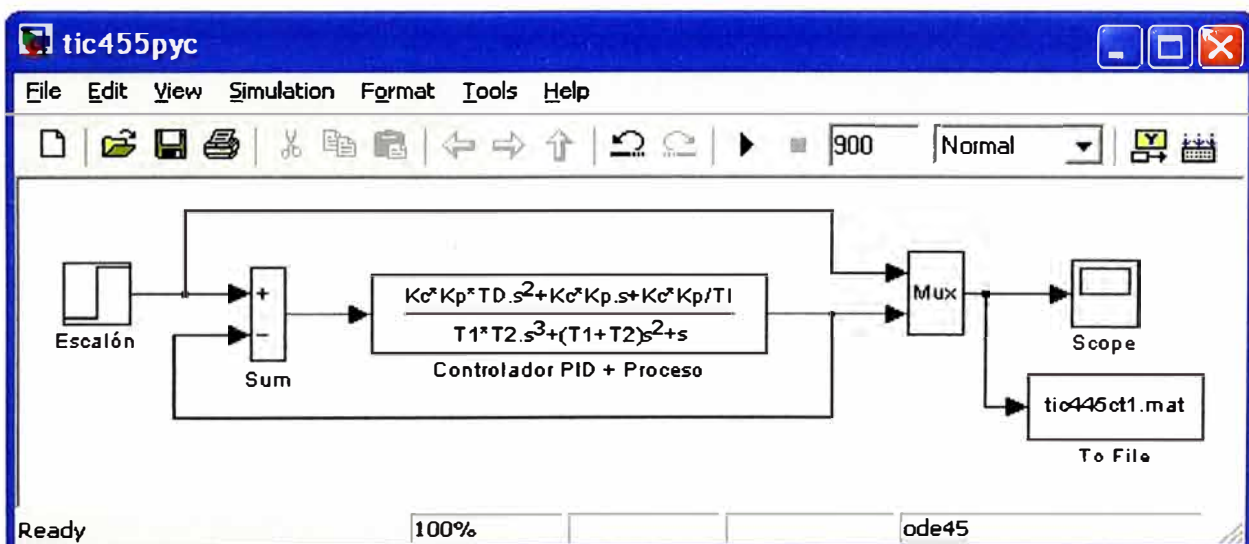
```

'HorizontalAlignment','center','VerticalAlignment','bottom');

### A.1.5) Programa para obtención de parámetros del controlador

```
% fic407cdat.m / Datos para el controlador PID FIC02-407
clc;
% Parámetros del controlador iniciales por método de ganancia limite
ctrl = 3;
% ctrl = 1 : controlador PI por método de ganancia limite
% ctrl = 2 : controlador PI por método de curva de reacción
% ctrl = 3 : controlador PID para mínimo rebasamiento
if (ctrl == 1)
    Kc = 0.45*Kcu;
    TI = Pcu/1.2;
    TD = 0;
elseif (ctrl == 2)
    Kc = 0.9*Tg/(Kp*Tu);
    TI = 3.33*Tu;
    TD = 0;
elseif (ctrl == 3)
    Kc = 0.8*Kcu;
    TI = 0.6*Pcu;
    TD = 0.19*Pcu;
end;
% parámetros óptimos
% Kc = 0.028;
% TI = 36.5;
TD = 9;
```

### A.1.6) Modelo de simulación de acción del controlador sobre el proceso



**A.1.7) Programa para análisis de la respuesta del proceso con acción del controlador**

```

% tic445rsl.m : Acción del controlador sobre el proceso
clc;
% Vectores dato obtenidos de la simulación
tic445TM = tic445rsl1(1,:);
tic445SP = tic445rsl1(2,:);
tic445PV = tic445rsl1(3,:);
n = max(size(tic445PV));
% Cálculo de parámetros de respuesta
for i=1:n
    if (tic445PV(1,i)<=max(tic445SP) && tic445PV(1,i+1)>=max(tic445SP))
        i_TR = i;
        break;
    else
        i_TR = -1;
    end;
end;
TR = tic445TM(1,i_TR);
for i=1:n
    if (tic445PV(1,i)==max(tic445PV))
        i_PO = i;
        break;
    else
        i_PO = -1;
    end;
end;
PO = tic445PV(1,i_PO);
POP = 100*(tic445PV(1,i_PO)-max(tic445SP))/max(tic445SP);
TP = tic445TM(1,i_PO);
Ess = 0.01;
Lim_sup = max(tic445SP)*(1+Ess);
Lim_inf = max(tic445SP)*(1-Ess);
for i=n:-1:1
    if (tic445PV(1,i)>=Lim_sup || tic445PV(1,i)<=Lim_inf)
        i_Tss = i;
        break;
    else
        i_Tss = -1;
    end;
end;
end;

```

```

Tss = tic445TM(1,i_Tss);
% Gráfica de resultados
figure;
plot(tic445TM,tic445SP,'--',tic445TM,tic445PV);
xlabel('Tiempo (seg)');
ylabel('Temperatura (°C)');
title('Acción del controlador sobre el proceso','Color','b',...
      'FontWeight','bold');
legend('SP','PV'.4);
hold on;
texto1 = [' ← Overshoot = ' num2str(POP) '%'];
plot(tic445TM(1,i_PO),tic445PV(1,i_PO),'x','MarkerEdgeColor','b',...
      'MarkerSize',10);
text(tic445TM(1,i_PO),tic445PV(1,i_PO),texto1,...
      'HorizontalAlignment','left','FontWeight','bold');
hold on;
texto2(1) = {'Tp'};
% texto2(2) = {'(' num2str(TP) ')'};
plot([tic445TM(1,i_PO) tic445TM(1,i_PO)],[0 tic445PV(1,i_PO)],'k--');
text(tic445TM(1,i_PO),0,texto2,'HorizontalAlignment','Left',...
      'VerticalAlignment','Bottom','FontWeight','bold');
hold on;
plot(tic445TM(1,i_TR),tic445PV(1,i_TR),'ro');
hold on;
texto3(1) = {'Tr'};
% texto3(2) = {'(' num2str(TR) ')'};
plot([tic445TM(1,i_TR) tic445TM(1,i_TR)],[0 tic445PV(1,i_TR)],'k--');
text(tic445TM(1,i_TR),0,texto3,'HorizontalAlignment','Left',...
      'VerticalAlignment','Bottom','FontWeight','bold');
hold on;
plot([tic445TM(1,1) tic445TM(1,n)],[Lim_sup Lim_sup],'k:');
hold on;
plot([tic445TM(1,1) tic445TM(1,n)],[Lim_inf Lim_inf],'k:');
hold on;
plot(tic445TM(1,i_Tss),tic445PV(1,i_Tss),'x','MarkerEdgeColor','b',...
      'MarkerSize',10);
hold on;
texto4(1) = {'Ts'};
% texto3(2) = {'(' num2str(TR) ')'};
plot([tic445TM(1,i_Tss) tic445TM(1,i_Tss)],[0 tic445PV(1,i_Tss)],'k--');

```

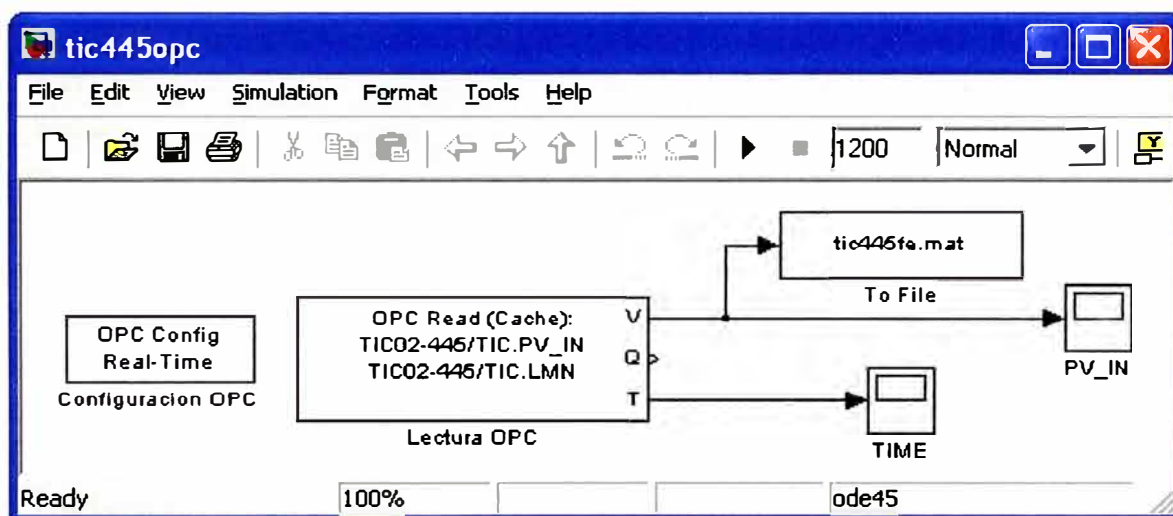
```

axis([0 900 0 140]);
text(tic445TM(1,i_Tss),0,texto4,'HorizontalAlignment','Left',...
    'VerticalAlignment','Bottom','FontWeight','bold');
texto5(1) = {'Tr = ' num2str(TR) ' seg'};
texto5(2) = {'Tp = ' num2str(TP) ' seg'};
texto5(3) = {'Ts = ' num2str(Tss) ' seg'};
text(tic445TM(1,round(n*70/100)),round(max(tic445SP)*60/100),texto5,'HorizontalAlignment','Left',
    'VerticalAlignment','Bottom','FontWeight','bold');

```

## A.2) Modelos y programas para controlador de temperatura

### A.2.1) Modelo para comunicación con servidor OPC



### A.2.2) Programa para análisis de datos y obtención de parámetros del proceso

% tic445anl.m: Análisis de datos obtenidos de TIC02-445 por OPC Server WinCC

```

clc;
% Calculo de vectores LMN y PV_IN
PID_TIME = tic445ve(1,:);
PID_PVIN = tic445ve(2,:);
PID_LMN = tic445ve(3,:);
% parametros varios
n = max(size(tic445ve));
PID_PVIN_SM = fastsmooth2(PID_PVIN,100,3,1);
% Cálculo de la 1era. derivada
d1_PVIN = zeros(1,n);
for i=1:n-1
    d1_PVIN(1,i) = PID_PVIN_SM(1,i+1)-PID_PVIN_SM(1,i);
end;
d1_PVIN(1,n) = d1_PVIN(1,n-1);
% eliminación de ruido

```



```

d1_PVIN_SM = fastsmooth2(d1_PVIN,150,3,1);
% Cálculo de la 2da. derivada
d2_PVIN = zeros(1,n);
for i=1:n-1
    d2_PVIN(1,i) = d1_PVIN_SM(1,i+1)-d1_PVIN_SM(1,i);
end;
d2_PVIN(1,n) = d2_PVIN(1,n-1);
% eliminación de ruido
d2_PVIN_SM = fastsmooth2(d2_PVIN,90,3,1);
% Búsqueda del punto de inflección en 2da. derivada = 0
for i=1:n-1
    if (d2_PVIN_SM(1,i)>=0) && (d2_PVIN_SM(1,i+1)<=0) && (i>20)
        i_pf1 = i;
        break;
    else
        i_pf1 = -1;
    end;
end;
% Búsqueda del punto de inflección en 1ra. derivada = max
for i=1:n-1
    if (d1_PVIN_SM(1,i)==max(d1_PVIN_SM(1,10:n)) && i>10)
        i_pf2 = i;
        break;
    else
        i_pf2 = -1;
    end;
end;
i_pf = i_pf2;
% Recta tangente en el punto de inflexión
fcorr = 1.0; % factor de corrección
Y_tg = fcorr*d1_PVIN_SM(1,i_pf)*(PID_TIME-PID_TIME(1,i_pf))/(PID_TIME(1,i_pf+1)-
PID_TIME(1,i_pf))+PID_PVIN_SM(1,i_pf);
% Cálculo de Tu y Tg
for i=1:n-1
    if ((PID_LMN(i+1)-PID_LMN(i))>0)
        i_Tinit = i;
        break;
    else
        i_Tinit = -1;
    end;
end;

```

```

end;
for i=1:n-1
    if (Y_tg(1,i)<=PID_PVIN_SM(1,i_Tinit)) && (Y_tg(1,i+1)>=PID_PVIN_SM(1,i_Tinit))
        i_Tu = i;
        break;
    else
        i_Tu = -1;
    end;
end;
for i=1:n-1
    if (Y_tg(1,i)>=max(PID_PVIN_SM))
        i_Tg = i;
        break;
    else
        i_Tg = -1;
    end;
end;
end;
Tu = PID_TIME(1,i_Tu)-PID_TIME(1,i_Tinit);
Tg = PID_TIME(1,i_Tg)-PID_TIME(1,i_Tu);
% Cálculo de parámetros del proceso
Kp = (max(PID_PVIN_SM)-min(PID_PVIN_SM))/(max(PID_LMN)-min(PID_LMN));
K = 5;
T1 = 0.107*Tg;
T2 = K*T1;
%Gráficos
figure;
plot(PID_TIME,PID_PVIN);
xlabel('Tiempo (seg)');
ylabel('Temperatura (°C)');
title('Temperatura de Leche en Salida de Intercambiador','Color','b',...
    'FontWeight','bold');
figure;
plot(PID_TIME,PID_LMN,'r');
axis([PID_TIME(1,1) PID_TIME(1,n) -10 100]);
xlabel('Tiempo (seg)');
ylabel('Apertura de Válvula (%)');
title('Porcentaje de Apertura de Válvula en Ingreso de Vapor','Color','b',...
    'FontWeight','bold');
figure;
plot(PID_TIME,PID_PVIN_SM,'r');

```

```

xlabel('Tiempo (seg)');
ylabel('Temperatura (°C)');
title('Temperatura de Leche en Salida de Intercambiador (sin ruido)','Color','b',...
      'FontWeight','bold');
figure;
plot(PID_TIME,d1_PVIN,'b');
xlabel('Tiempo (seg)');
ylabel('Derivada de Temperatura (°C/s)');
title('Primera Derivada de Temperatura de Leche en Salida Intercambiador','Color','b',...
      'FontWeight','bold');
figure;
plot(PID_TIME,d1_PVIN_SM,'r');
xlabel('Tiempo (seg)');
ylabel('Derivada de Temperatura (°C/s)');
title('Primera Derivada de Temperatura de Leche "Suavizada"', 'Color','b',...
      'FontWeight','bold');
hold on;
plot(PID_TIME(1,i_pf),d1_PVIN_SM(1,i_pf),'x','MarkerEdgeColor','b',...
      'MarkerSize',10);
text(PID_TIME(1,i_pf),d1_PVIN_SM(1,i_pf),'\leftarrow Máximo valor',...
      'HorizontalAlignment','left');
figure;
plot(PID_TIME,PID_PVIN_SM);
xlabel('Tiempo (seg)');
ylabel('Temperatura (°C)');
title('Respuesta del Proceso - Temperatura de Leche en Salida','Color','b',...
      'FontWeight','bold');
text(PID_TIME(1,round(35*n/100)),PID_PVIN_SM(1,round(35*n/100)),'\leftarrow Curva de
Reacción', 'HorizontalAlignment','left');
str1(1)={' \leftarrow Punto de};
str1(2)={'   Inflexión'};
text(PID_TIME(1,i_pf+1),PID_PVIN_SM(1,i_pf-4),str1,...
      'HorizontalAlignment','left','Color','r',...
      'VerticalAlignment','Middle');
hold on;
plot(PID_TIME(1,i_pf),PID_PVIN_SM(1,i_pf),'x','MarkerEdgeColor','k',...
      'MarkerSize',10);
hold on;
plot(PID_TIME(1,round(1*n/100):round(25*n/100)),Y_tg(1,round(1*n/100):round(25*n/100)), 'r');
text(PID_TIME(1,round(20*n/100)),Y_tg(1,round(20*n/100)), 'Tangente \rightarrow',...

```

```

    'HorizontalAlignment','right');
hold on;
plot([PID_TIME(1,1) PID_TIME(1,n)],[PID_PVIN_SM(1,i_Tinit) PID_PVIN_SM(1,i_Tinit)],'--k');
hold on;
plot([PID_TIME(1,1) PID_TIME(1,n)],[max(PID_PVIN_SM) max(PID_PVIN_SM)],'--k');
hold on;
plot([PID_TIME(1,i_Tg) PID_TIME(1,i_Tg)],[PID_PVIN_SM(1,i_Tinit) max(PID_PVIN_SM)],'--k');
hold on;
plot(PID_TIME(1,i_Tinit),PID_PVIN_SM(1,i_Tinit),'*k');
hold on;
plot(PID_TIME(1,i_Tu),PID_PVIN_SM(1,i_Tinit),'ok');
hold on;
plot(PID_TIME(1,i_Tg),PID_PVIN_SM(1,i_Tinit),'ok');
str2(1)={' '};
str2(2)={'Tu'};
text(PID_TIME(1,i_Tu),PID_PVIN_SM(1,i_Tinit),str2,...
    'HorizontalAlignment','center','Color','k',...
    'VerticalAlignment','Top');
str3(1)={' '};
str3(2)={'Tu+Tg'};
text(PID_TIME(1,i_Tg),PID_PVIN_SM(1,i_Tinit),str3,...
    'HorizontalAlignment','center','Color','k',...
    'VerticalAlignment','Top');

```

### A.2.3) Programa para obtención de parámetros del controlador

```

% tic491cdat.m / Datos para el controlador PID PIC02-491
clc;
% Parámetros del controlador por el método de curva de reacción
ctrl = 2;
% ctrl = 1 : controlador PI
% ctrl = 2 : controlador PID
if (ctrl == 1)
    Kc = 0.9*Tg/(Kp*Tu);
    TI = 3.33*Tu;
    TD = 0;
elseif (ctrl == 2)
    Kc = 1.2*Tg/(Kp*Tu);
    TI = 2*Tu;
    TD = 0.5*Tu;
end;

```

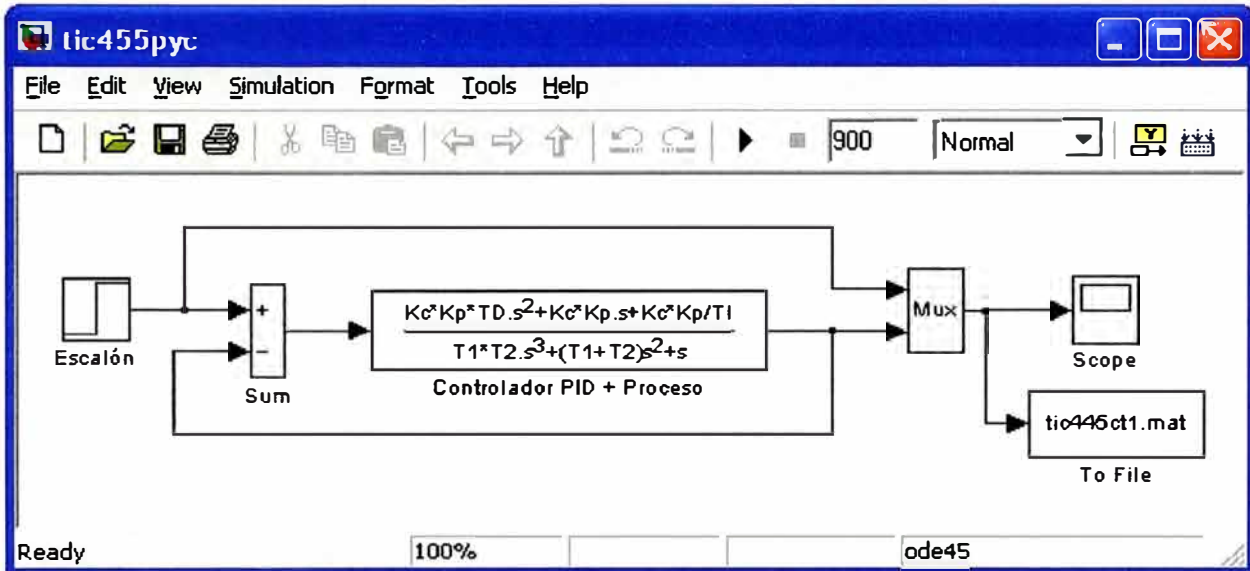
% parámetros óptimos

% Kc = 20.8;

TI = 55;

TD = 30;

#### A.2.4) Modelo de simulación de acción del controlador sobre el proceso



#### A.2.5) Programa para análisis de la respuesta del proceso con acción del controlador

% tic445rsl.m : Acción del controlador sobre el proceso

clc;

% Vectores dato obtenidos de la simulación

tic445TM = tic445rsl1(1,:);

tic445SP = tic445rsl1(2,:);

tic445PV = tic445rsl1(3,:);

n = max(size(tic445PV));

% Cálculo de parámetros de respuesta

for i=1:n

if (tic445PV(1,i) <= max(tic445SP) && tic445PV(1,i+1) >= max(tic445SP))

    i\_TR = i;

    break;

else

    i\_TR = -1;

end;

end;

TR = tic445TM(1,i\_TR);

for i=1:n

if (tic445PV(1,i) == max(tic445PV))

    i\_PO = i;

```

    break;
else
    i_PO = -1;
end;
end;
PO = tic445PV(1,i_PO);
POP = 100*(tic445PV(1,i_PO)-max(tic445SP))/max(tic445SP);
TP = tic445TM(1,i_PO);
Ess = 0.01;
Lim_sup = max(tic445SP)*(1+Ess);
Lim_inf = max(tic445SP)*(1-Ess);
for i=n:-1:1
    if (tic445PV(1,i)>=Lim_sup || tic445PV(1,i)<=Lim_inf)
        i_Tss = i;
        break;
    else
        i_Tss = -1;
    end;
end;
Tss = tic445TM(1,i_Tss);
% Gráfica de resultados
figure;
plot(tic445TM,tic445SP,'--',tic445TM,tic445PV);
xlabel('Tiempo (seg)');
ylabel('Temperatura (°C)');
title('Acción del controlador sobre el proceso','Color','b','FontWeight','bold');
legend('SP','PV',4);
hold on;
texto1 = [' ← Overshoot = ' num2str(POP) '%'];
plot(tic445TM(1,i_PO),tic445PV(1,i_PO),'x','MarkerEdgeColor','b',...
    'MarkerSize',10);
text(tic445TM(1,i_PO),tic445PV(1,i_PO),texto1,...
    'HorizontalAlignment','left','FontWeight','bold');
hold on;
texto2(1) = {'Tp'};
% texto2(2) = {'(' num2str(TP) ')'};
plot([tic445TM(1,i_PO) tic445TM(1,i_PO)],[0 tic445PV(1,i_PO)],'k--');
text(tic445TM(1,i_PO),0,texto2,'HorizontalAlignment','Left',...
    'VerticalAlignment','Bottom','FontWeight','bold');
hold on;

```

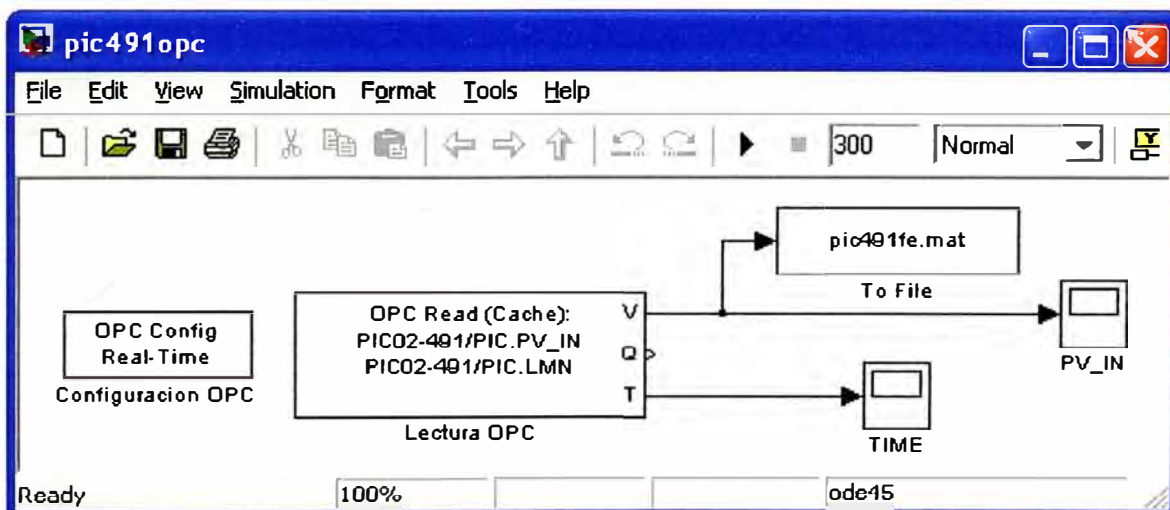
```

plot(tic445TM(1,i_TR),tic445PV(1,i_TR),'ro');
hold on;
texto3(1) = {'Tr'};
% texto3(2) = {'(' num2str(TR) ')'};
plot([tic445TM(1,i_TR) tic445TM(1,i_TR)],[0 tic445PV(1,i_TR)],'k--');
text(tic445TM(1,i_TR),0,texto3,'HorizontalAlignment','Left',...
    'VerticalAlignment','Bottom','FontWeight','bold');
hold on;
plot([tic445TM(1,1) tic445TM(1,n)],[Lim_sup Lim_sup],'k:');
hold on;
plot([tic445TM(1,1) tic445TM(1,n)],[Lim_inf Lim_inf],'k:');
hold on;
plot(tic445TM(1,i_Tss),tic445PV(1,i_Tss),'x','MarkerEdgeColor','b',...
    'MarkerSize',10);
hold on;
texto4(1) = {'Ts'};
% texto3(2) = {'(' num2str(TR) ')'};
plot([tic445TM(1,i_Tss) tic445TM(1,i_Tss)],[0 tic445PV(1,i_Tss)],'k--');
axis([0 900 0 140]);
text(tic445TM(1,i_Tss),0,texto4,'HorizontalAlignment','Left',...
    'VerticalAlignment','Bottom','FontWeight','bold');
texto5(1) = {'Tr = ' num2str(TR) ' seg'};
texto5(2) = {'Tp = ' num2str(TP) ' seg'};
texto5(3) = {'Ts = ' num2str(Tss) ' seg'};
text(tic445TM(1,round(n*70/100)),round(max(tic445SP)*60/100),texto5,'HorizontalAlignment','Left',
    'VerticalAlignment','Bottom','FontWeight','bold');

```

### A.3) Modelos y programas para controlador de presión PIC02-491

#### A.3.1) Modelo para comunicación con servidor OPC



**A.3.2) Programa para análisis de datos y obtención de parámetros del proceso**

```
% pic491anl.m: Análisis de datos obtenidos de PIC02-491 por OPC Server WinCC
```

```
clc;
% Calculo de vectores LMN y PV_IN
PID_TIME = pic491ve(1,:);
PID_PVIN = pic491ve(2,:);
PID_LMN = pic491ve(3,:);
% parametros varios
n = max(size(pic491ve));
PID_PVIN_SM = fastsmooth2(PID_PVIN,100,3,1);
% Cálculo de la 1era. derivada
d1_PVIN = zeros(1,n);
for i=1:n-1
    d1_PVIN(1,i) = PID_PVIN_SM(1,i+1)-PID_PVIN_SM(1,i);
end;
d1_PVIN(1,n) = d1_PVIN(1,n-1);
% eliminación de ruido
d1_PVIN_SM = fastsmooth2(d1_PVIN,150,3,1);
% Cálculo de la 2da. derivada
d2_PVIN = zeros(1,n);
for i=1:n-1
    d2_PVIN(1,i) = d1_PVIN_SM(1,i+1)-d1_PVIN_SM(1,i);
end;
d2_PVIN(1,n) = d2_PVIN(1,n-1);
% eliminación de ruido
d2_PVIN_SM = fastsmooth2(d2_PVIN,90,3,1);
% Búsqueda del punto de inflección en 2da. derivada = 0
for i=1:n-1
    if (d2_PVIN_SM(1,i)>=0) && (d2_PVIN_SM(1,i+1)<=0) && (i>20)
        i_pf1 = i;
        break;
    else
        i_pf1 = -1;
    end;
end;
% Búsqueda del punto de inflección en 1ra. derivada = max
for i=1:n-1
    if (d1_PVIN_SM(1,i)==max(d1_PVIN_SM(1,10:n)) && i>10)
        i_pf2 = i;
        break;
    end;
end;
```



```

else
    i_pf2 = -1;
end;
end;
i_pf = i_pf2;
% Recta tangente en el punto de inflexión
fcorr = 1.05; % factor de corrección
Y_tg = fcorr*d1_PVIN_SM(1,i_pf)*(PID_TIME-PID_TIME(1,i_pf))/(PID_TIME(1,i_pf+1)-
PID_TIME(1,i_pf))+PID_PVIN_SM(1,i_pf);
% Cálculo de Tu y Tg
for i=1:n-1
    if ((PID_LMN(i+1)-PID_LMN(i))>0)
        i_Tinit = i;
        break;
    else
        i_Tinit = -1;
    end;
end;
for i=1:n-1
    if (Y_tg(1,i)<=PID_PVIN_SM(1,i_Tinit)) && (Y_tg(1,i+1)>=PID_PVIN_SM(1,i_Tinit))
        i_Tu = i;
        break;
    else
        i_Tu = -1;
    end;
end;
for i=1:n-1
    if (Y_tg(1,i)>=max(PID_PVIN_SM))
        i_Tg = i;
        break;
    else
        i_Tg = -1;
    end;
end;
Tu = PID_TIME(1,i_Tu)-PID_TIME(1,i_Tinit);
Tg = PID_TIME(1,i_Tg)-PID_TIME(1,i_Tu);
% Cálculo de parámetros del proceso
Kp = (max(PID_PVIN_SM)-min(PID_PVIN_SM))/(max(PID_LMN)-min(PID_LMN));
T = 0.368*Tg;
%Gráficos

```

```

figure;
plot(PID_TIME,PID_PVIN);
xlabel('Tiempo (seg)');
ylabel('Presión (Bar)');
title('Presión de Vapor en Termocompresor','Color','b',...
      'FontWeight','bold');
figure;
plot(PID_TIME,PID_LMN,'r');
axis([PID_TIME(1,1) PID_TIME(1,n) -10 100]);
xlabel('Tiempo (seg)');
ylabel('Apertura de Válvula (%)');
title('Porcentaje de Apertura de Válvula en Ingreso de Vapor','Color','b',...
      'FontWeight','bold');
figure;
plot(PID_TIME,PID_PVIN_SM,'r');
xlabel('Tiempo (seg)');
ylabel('Presión (Bar)');
title('Presión de Vapor en Termocompresor (sin ruido)','Color','b',...
      'FontWeight','bold');
figure;
plot(PID_TIME,d1_PVIN,'b');
xlabel('Tiempo (seg)');
ylabel('Derivada de Presión (Bar/s)');
title('Primera Derivada de Presión de Vapor en Termocompresor','Color','b',...
      'FontWeight','bold');
figure;
plot(PID_TIME,d1_PVIN_SM,'r');
xlabel('Tiempo (seg)');
ylabel('Derivada de Presión (Bar/s)');
title('Primera Derivada Presión de Vapor en Termocompresor "Suavizada"', 'Color','b',...
      'FontWeight','bold');
hold on;
plot(PID_TIME(1,i_pf),d1_PVIN_SM(1,i_pf),'x','MarkerEdgeColor','b',...
      'MarkerSize',10);
text(PID_TIME(1,i_pf),d1_PVIN_SM(1,i_pf),'leftarrow Máximo valor',...
      'HorizontalAlignment','left');
figure;
plot(PID_TIME,PID_PVIN_SM);
xlabel('Tiempo (seg)');
ylabel('Presión (Bar)');

```

```

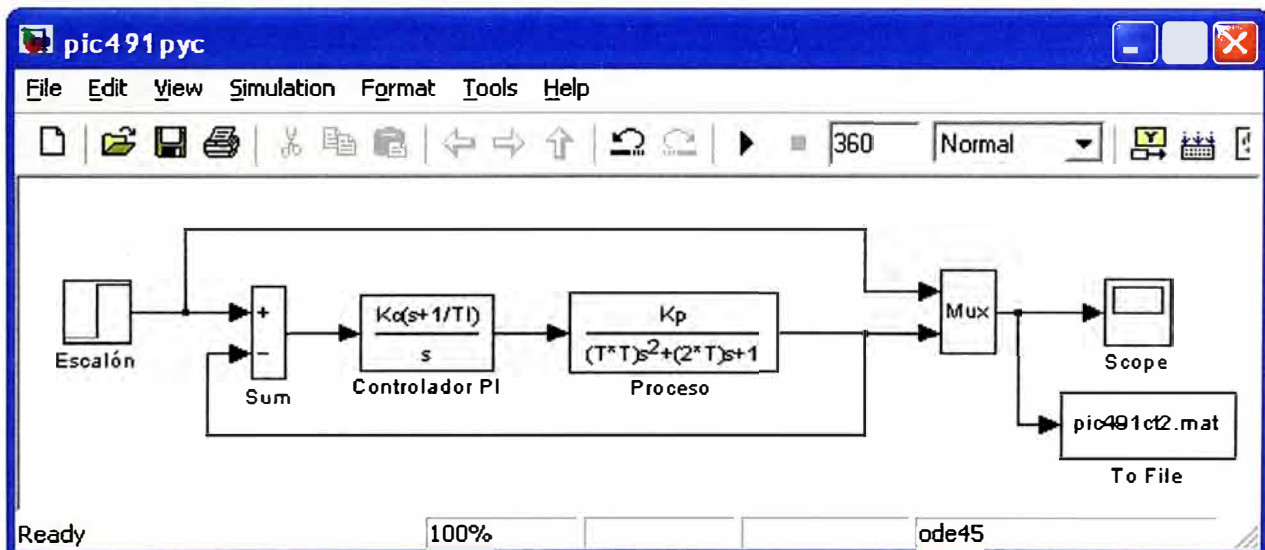
title('Respuesta del Proceso - Presión de Vapor en Termocompresor','Color','b',...
      'FontWeight','bold');
text(PID_TIME(1,round(35*n/100)),PID_PVIN_SM(1,round(35*n/100)),'\leftarrow Curva de
Reacción',...
      'HorizontalAlignment','left');
str1(1) ={' \leftarrow Punto de};
str1(2) ={'   Inflexión'};
text(PID_TIME(1,i_pf+1),PID_PVIN_SM(1,i_pf-4),str1,...
      'HorizontalAlignment','left','Color','r',...
      'VerticalAlignment','Middle');
hold on;
plot(PID_TIME(1,i_pf),PID_PVIN_SM(1,i_pf),'x','MarkerEdgeColor','k',...
      'MarkerSize',10);
hold on;
plot(PID_TIME(1,round(1*n/100):round(33*n/100)),Y_tg(1,round(1*n/100):round(33*n/100)), 'r');
text(PID_TIME(1,round(20*n/100)),Y_tg(1,round(20*n/100)), 'Tangente \rightarrow',...
      'HorizontalAlignment','right');
hold on;
plot([PID_TIME(1,1) PID_TIME(1,n)], [PID_PVIN_SM(1,i_Tinit) PID_PVIN_SM(1,i_Tinit)], '--k');
hold on;
plot([PID_TIME(1,1) PID_TIME(1,n)], [max(PID_PVIN_SM) max(PID_PVIN_SM)], '--k');
hold on;
plot([PID_TIME(1,i_Tg) PID_TIME(1,i_Tg)], [PID_PVIN_SM(1,i_Tinit) max(PID_PVIN_SM)], '--k');
hold on;
plot(PID_TIME(1,i_Tinit),PID_PVIN_SM(1,i_Tinit),'*k');
hold on;
plot(PID_TIME(1,i_Tu),PID_PVIN_SM(1,i_Tinit),'ok');
hold on;
plot(PID_TIME(1,i_Tg),PID_PVIN_SM(1,i_Tinit),'ok');
str2(1) ={' '};
str2(2) ={'Tu'};
text(PID_TIME(1,i_Tu),PID_PVIN_SM(1,i_Tinit),str2,...
      'HorizontalAlignment','center','Color','k',...
      'VerticalAlignment','Top');
str3(1) ={' '};
str3(2) ={'Tu+Tg'};
text(PID_TIME(1,i_Tg),PID_PVIN_SM(1,i_Tinit),str3,...
      'HorizontalAlignment','center','Color','k',...
      'VerticalAlignment','Top');

```

### A.3.3) Programa para obtención de parámetros del controlador

```
% pic491cdat.m / Datos para el controlador PID PIC02-491
clc;
% Parámetros del controlador por el método de curva de reacción
ctrl = 1;
% ctrl = 1 : controlador PI
% ctrl = 2 : controlador PID
if (ctrl == 1)
    Kc = 0.9*Tg/(Kp*Tu);
    TI = 3.33*Tu;
    TD = 0;
elseif (ctrl == 2)
    Kc = 1.2*Tg/(Kp*Tu);
    TI = 2*Tu;
    TD = 0.5*Tu;
end;
% parámetros óptimos
Kc = 45;
TI = 90.5;
```

### A.3.4) Modelo de simulación de acción del controlador sobre el proceso



### A.3.5) Programa para análisis de la respuesta del proceso con acción del controlador

```
% pic491rsl.m : Acción del controlador sobre el proceso
clc;
% Vectores dato obtenidos de la simulación
pic491TM = pic491rsl2(1,:);
pic491SP = pic491rsl2(2,:);
```

```

pic491PV = pic491rsI2(3,:);
n = max(size(pic491PV));
% Cálculo de parámetros de respuesta
for i=1:n
    if (pic491PV(1,i)<=max(pic491SP) && pic491PV(1,i+1)>=max(pic491SP))
        i_TR = i;
        break;
    else
        i_TR = -1;
    end;
end;
TR = pic491TM(1,i_TR);
for i=1:n
    if (pic491PV(1,i)==max(pic491PV))
        i_PO = i;
        break;
    else
        i_PO = -1;
    end;
end;
PO = pic491PV(1,i_PO);
POP = 100*(pic491PV(1,i_PO)-max(pic491SP))/max(pic491SP);
TP = pic491TM(1,i_PO);
Ess = 0.025;
Lim_sup = max(pic491SP)*(1+Ess);
Lim_inf = max(pic491SP)*(1-Ess);
for i=n:-1:1
    if (pic491PV(1,i)>=Lim_sup || pic491PV(1,i)<=Lim_inf)
        i_Tss = i;
        break;
    else
        i_Tss = -1;
    end;
end;
Tss = pic491TM(1,i_Tss);
% Gráfica de resultados
figure;
plot(pic491TM,pic491SP,'--',pic491TM,pic491PV);
xlabel('Tiempo (seg)');
ylabel('Presión (Bar)');

```

```

title('Acción del controlador sobre el proceso','Color','b',...
      'FontWeight','bold');
legend('SP','PV'.4);
hold on;
texto1 = [' ← Overshoot = ' num2str(POP) '%'];
plot(pic491TM(1,i_PO),pic491PV(1,i_PO),'x','MarkerEdgeColor','b',...
      'MarkerSize',10);
text(pic491TM(1,i_PO),pic491PV(1,i_PO),texto1,...
      'HorizontalAlignment','left','FontWeight','bold');
hold on;
texto2(1) = {'Tp'};
% texto2(2) = {'(' num2str(TP) ')'};
plot([pic491TM(1,i_PO) pic491TM(1,i_PO)],[0 pic491PV(1,i_PO)],'k--');
text(pic491TM(1,i_PO),0,texto2,'HorizontalAlignment','Left',...
      'VerticalAlignment','Bottom','FontWeight','bold');
hold on;
plot(pic491TM(1,i_TR),pic491PV(1,i_TR),'ro');
hold on;
texto3(1) = {'Tr'};
% texto3(2) = {'(' num2str(TR) ')'};
plot([pic491TM(1,i_TR) pic491TM(1,i_TR)],[0 pic491PV(1,i_TR)],'k--');
text(pic491TM(1,i_TR),0,texto3,'HorizontalAlignment','Left',...
      'VerticalAlignment','Bottom','FontWeight','bold');
hold on;
plot([pic491TM(1,1) pic491TM(1,n)],[Lim_sup Lim_sup],'k:');
hold on;
plot([pic491TM(1,1) pic491TM(1,n)],[Lim_inf Lim_inf],'k:');
hold on;
plot(pic491TM(1,i_Tss),pic491PV(1,i_Tss),'x','MarkerEdgeColor','b',...
      'MarkerSize',10);
hold on;
texto4(1) = {'Ts'};
% texto3(2) = {'(' num2str(TR) ')'};
plot([pic491TM(1,i_Tss) pic491TM(1,i_Tss)],[0 pic491PV(1,i_Tss)],'k--');
axis([0 360 0 9]);
text(pic491TM(1,i_Tss),0,texto4,'HorizontalAlignment','Left',...
      'VerticalAlignment','Bottom','FontWeight','bold');
texto5(1) = {'Tr = ' num2str(TR) ' seg'};
texto5(2) = {'Tp = ' num2str(TP) ' seg'};

```

```

texto5(3) = {'Ts = ' num2str(Tss) ' seg'};
text(pic491TM(1,round(n*70/100)),round(max(pic491SP)*60/100),texto5,'HorizontalAlignment','Left'
,'VerticalAlignment','Bottom','FontWeight','bold');

```

#### A.4) Programa para realizar la función de eliminación de ruido (“smoothing”)

```

function SmoothY=fastsmooth2(Y,w,type,ends)
% fastsmooth(Y,w,type,ends) smooths vector Y with smooth
% of width w. Version 2.0, May 2008.
% The argument "type" determines the smooth type:
% If type=1, rectangular (sliding-average or boxcar)
% If type=2, triangular (2 passes of sliding-average)
% If type=3, pseudo-Gaussian (3 passes of sliding-average)
% The argument "ends" controls how the "ends" of the signal
% (the first w/2 points and the last w/2 points) are handled.
% If ends=0, the ends are zero. (In this mode the elapsed
% time is independent of the smooth width). The fastest.
% If ends=1, the ends are smoothed with progressively
% smaller smooths the closer to the end. (In this mode the
% elapsed time increases with increasing smooth widths).
% fastsmooth(Y,w,type) smooths with ends=0.
% fastsmooth(Y,w) smooths with type=1 and ends=0.
% Example:
% fastsmooth([1 1 1 10 10 10 1 1 1 1],3)= [0 1 4 7 10 7 4 1 1 0]
% fastsmooth([1 1 1 10 10 10 1 1 1 1],3,1,1)= [1 1 4 7 10 7 4 1 1 1]
% T. C. O'Haver, 2008.
if nargin==2, ends=0; type=1; end
if nargin==3, ends=0; end
switch type
case 1
SmoothY=sa(Y,w,ends);
case 2
SmoothY=sa(sa(Y,w,ends),w,ends);
case 3
SmoothY=sa(sa(sa(Y,w,ends),w,ends),w,ends);
end
function SmoothY=sa(Y,smoothwidth,ends)
w=round(smoothwidth);
SumPoints=sum(Y(1:w));
s=zeros(size(Y));
halfw=round(w/2);

```

```

L=length(Y);
for k=1:L-w,
    s(k+halfw-1)=SumPoints;
    SumPoints=SumPoints-Y(k);
    SumPoints=SumPoints+Y(k+w);
end
s(k+halfw)=sum(Y(L-w+1:L));
SmoothY=s./w;
% Taper the ends of the signal if ends=1.
if ends==1,
    startpoint=(smoothwidth + 1)/2;
    SmoothY(1)=(Y(1)+Y(2))./2;
    for k=2:startpoint,
        SmoothY(k)=mean(Y(1:(2*k-1)));
        SmoothY(L-k+1)=mean(Y(L-2*k+2:L));
    end
    SmoothY(L)=(Y(L)+Y(L-1))./2;
end

```



## BIBLIOGRAFÍA

- [1] Gösta Bylund, M.Sc. (Dairy Techn.), "Dairy processing handbook", Tetra Pak Processing Systems AB S-221 86 Lund, Sweden, capítulo 6.5 (1995).
- [2] Ph.D. Arturo Rojas Moreno, 11vo. Programa de Titulación por Actualización de Conocimientos (Enero – Abril 2009), Curso: Ingeniería de Control Moderna.
- [3] Siemens AG Industry Sector, Process Control System PCS 7 - PCS 7 Standard Library V71, capítulo 15.2.1 (Marzo-2009).
- [4] Antonio Creus Solé, Instrumentación Industrial – 6ta. Edición, capítulo 9.5 a 9.7, Alfaomega Grupo Editor (1998).
- [5] The MathWorks Inc., SIMULINK - Model-Based and System-Based Design, Version 5 (Julio 2002).
- [6] Professor Emeritus, University of Maryland at College Park, Department of Chemistry and Biochemistry, paper: "An Introduction to Signal Processing in Chemical Analysis", capítulo: "Smoothing" – Septiembre 2009.  
Enlace web: <http://terpconnect.umd.edu/~toh/spectrum/Smoothing.html>.