

Trabajo Fin de Máster
Máster Universitario en Ingeniería Industrial

Control de un reactor continuo de tanque agitado

Autor: Jesús Fernández Romero

Tutor: Francisco Javier Gutiérrez Ortiz

Dpto. de Ingeniería Química y Ambiental
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla



Trabajo Fin de Máster
Máster Universitario en Ingeniería Industrial

Control de un reactor continuo de tanque agitado

Autor:

Jesús Fernández Romero

Tutor:

Francisco Javier Gutiérrez Ortiz

Catedrático de Universidad

Dpto. de Ingeniería Química y Ambiental

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2022

Trabajo Fin de Máster: Control de un reactor continuo de tanque agitado

Autor: Jesús Fernández Romero

Tutor: Francisco Javier Gutiérrez Ortiz

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2022

El Secretario del Tribunal

Agradecimientos

Quiero agradecer este Trabajo Fin de Máster a:

Mis profesores, por su trabajo y dedicación.

Mi tutor, por su constante esfuerzo dirigiéndome este TFM, respondiéndome incluso los fines de semana.

Mis amigos, por ayudarme en los malos momentos que he tenido.

Mi familia, por ser un pilar fundamental en mi vida, apoyándome siempre en todas mis decisiones.

En especial, quiero dedicarle “In memoriam” este Trabajo Fin de Master a mi hermano Alejandro, quien siempre ha estado y estará a mi lado en todo momento.

Jesús Fernández Romero

Sevilla, 2022

En este trabajo se desarrolla una aplicación informática con una interfaz gráfica de usuario que facilitará el aprendizaje de los fundamentos del control automático aplicado a procesos químicos, específicamente, en reactores de tanque agitado con dinámica no lineal. El desarrollo se ha hecho mediante la programación en Matlab como entorno de simulación.

Para ello, en primer lugar, se han obtenido tres modelos dinámicos de un reactor de tanque agitado continuo con diferentes métodos de refrigeración: mediante un serpentín y mediante una camisa con y sin recirculación de refrigerante. Posteriormente, se ha programado un simulador para integrar las ecuaciones de los modelos a partir de unas condiciones iniciales, mediante los métodos de Euler y Runge-Kutta de orden 4. Asimismo, se ha usado un temporizador para permitir la simulación en tiempo real o en tiempo de simulación, con una velocidad del proceso más rápida. Por otro lado, se ha diseñado un algoritmo para la identificación de la dinámica del proceso. El ajuste de los parámetros se realiza aplicando el método de mínimos cuadrados, sobre modelos lineales de primer orden con tiempo muerto, partiendo de los datos obtenidos por simulación. Estos modelos simplifican la dinámica no lineal del proceso real.

Para cada modelo, se han programado las ecuaciones de distintos reguladores, con el objetivo de controlar la temperatura del reactor. Se han utilizado algoritmos PID con y sin acción anticipativa, realimentación del estado y control predictivo DMC monovariable. Además, para el reactor encamisado, se han desarrollado las estructuras de control en cascada, control adaptativo de ajuste por tabla y control autoajustable basado en mínimos cuadrados recursivos. Todos los parámetros de ajuste han sido diseñados para proporcionar al usuario una solución por defecto.

Finalmente, se han integrado todas estas funcionalidades en una interfaz gráfica de usuario multiventana, que permite el ajuste de las diferentes opciones de simulación y la selección del algoritmo de control. Además, se ha incluido una ventana para realizar ensayos de entrada en escalón, permitiendo exportar a una hoja de cálculo los datos obtenidos para realizar la identificación. Con fines didácticos, se ha creado un manual en el que se detalla el procedimiento a seguir para la instalación y manejo del software diseñado.

Abstract

In this work, an application focused on the study and learning of the fundamentals of automatic control applied to chemical processes has been developed, specifically for jacketed reactors with non-linear dynamics, by programming in Matlab that is a simulation environment for nonlinear dynamic reactors.

For this purpose, three models of continuous stirred tank reactors with different cooling of the coolant have been obtained: by means of a coil and using a cooling jacket with and without recirculation of the coolant. Subsequently, in order to integrate the equations of the models from some initial conditions, a simulator has been programmed through Euler and 4th order Runge-Kutta methods. In addition, a timer has been used to enable a real time simulation or to speed up the time simulation of the process. On the other hand, an algorithm has been designed for the identification of approximate first order systems plus dead time, using least squares method from data obtained by simulation. These models simplify the nonlinear dynamics of the real process.

For each model, the equations of different temperature regulators have been programmed. The algorithms used were standard PID structure with and without feedforward action, state feedback and monovariable DMC predictive control. Furthermore, a cascade control structure, an adaptive Gain Scheduling technique and a self-tuning control based on recursive least squares method have been developed for the case of jacketed reactor. All of the tuning parameters have been designed to provide the user a functional default solution.

Finally, all these functionalities have been integrated into a multi-window graphical user interface, which also allows the adjustment of different simulation options and the selection of the control algorithm. A dedicated window for steps input experiments has also been included and the data and the output obtained can be exported to a spreadsheet for identification purposes. For didactic reasons, a user manual has been created detailing the procedure to follow for the installation and operation of the designed software.

Agradecimientos	v
Resumen	vii
Abstract	ix
Índice	xi
Índice de Tablas	xiii
Índice de Figuras	xv
Notación	xix
<i>Símbolos</i>	<i>xix</i>
<i>Variables y propiedades fisicoquímicas</i>	<i>xx</i>
<i>Variables de modelos lineales e identificación</i>	<i>xxi</i>
<i>Variables de simulación y control</i>	<i>xxii</i>
<i>Subíndices</i>	<i>xxiii</i>
Glosario	xxv
1 Introducción	1
1.1 <i>Introducción</i>	<i>1</i>
2 Modelado del reactor de tanque agitado continuo	3
2.1 <i>Modelo del reactor CSTR con serpentín de refrigeración</i>	<i>3</i>
2.2 <i>Modelo del reactor CSTR encamisado</i>	<i>6</i>
2.3 <i>Modelo del reactor CSTR encamisado con recirculación de refrigerante</i>	<i>8</i>
2.4 <i>Aproximación de retardos presentes en el proceso real</i>	<i>10</i>
2.5 <i>Linealización del modelo en torno al punto de funcionamiento lineal</i>	<i>10</i>
3 Simulación del proceso	13
3.1 <i>Simulación de las ecuaciones mediante integración numérica</i>	<i>13</i>
3.2 <i>Simulación dinámica</i>	<i>17</i>
4 Identificación del proceso	21
4.1 <i>Identificación paramétrica de sistemas lineales aproximados</i>	<i>21</i>
4.2 <i>Método de identificación paramétrica por mínimos cuadrados</i>	<i>25</i>
4.3 <i>Transformación del modelo de tiempo discreto a tiempo continuo</i>	<i>27</i>
4.4 <i>Herramienta de ensayos con entrada en escalón para identificación de modelos aproximados</i>	<i>29</i>
4.5 <i>Detección y prevención de errores en la identificación</i>	<i>31</i>
4.6 <i>Modificación de funciones de transferencia asociadas al proceso</i>	<i>32</i>
5 Sistema de control por realimentación	35
5.1 <i>Control PID</i>	<i>35</i>
5.1.1 <i>Algoritmo de control PID en tiempo continuo</i>	<i>35</i>
5.1.2 <i>Discretización del controlador PID</i>	<i>38</i>
5.1.3 <i>Implementación en la GUI del algoritmo de control PID</i>	<i>40</i>
5.1.4 <i>Diseño de un controlador P/PI/PID mediante reglas de sintonización</i>	<i>40</i>
5.1.5 <i>Acción anticipativa</i>	<i>42</i>
5.1.6 <i>Estrategia de control en cascada</i>	<i>45</i>
5.2 <i>Control por realimentación del vector de estados</i>	<i>48</i>

5.2.1	Control proporcional con pre-compensador en la entrada.....	48
5.2.2	Realimentación del estado con acción integral	49
5.2.3	Diseño de un controlador óptimo por realimentación de estado mediante LQR	50
5.2.4	Herramienta de sintonización de controladores de realimentación de estado	52
5.3	<i>Control predictivo DMC</i>	54
5.3.1	Algoritmo de control DMC para sistemas lineales.....	54
5.3.2	Algoritmos para la implementación en la GUI de un control DMC para sistemas no lineales	57
6	Control adaptativo y autoajutable	63
6.1	<i>Estrategia de control adaptativo mediante ajustable por tabla</i>	<i>63</i>
6.2	<i>Controlador autoajutable basado en mínimos cuadrados recursivos</i>	<i>66</i>
7	Mejoras futuras	75
7.1	<i>Opciones de simulación</i>	<i>75</i>
7.2	<i>Identificación</i>	<i>75</i>
7.3	<i>Opciones de control</i>	<i>76</i>
7.4	<i>Visualización</i>	<i>76</i>
	Referencias.....	77
	Anexo A. Manual de la interfaz gráfica de usuario.....	79
1.	<i>Instalación</i>	<i>79</i>
2.	<i>Inicio</i>	<i>81</i>
3.	<i>Configuración de la simulación.....</i>	<i>84</i>
4.	<i>Simulación del proceso</i>	<i>86</i>
4.1.	Simulación I. Algoritmo de simulación estática	86
4.2.	Simulación II. Algoritmo de simulación dinámica	87
5.	<i>Realización de ensayos con entrada en escalón para identificación paramétrica</i>	<i>88</i>
5.1.	Obtención de datos para identificación	90
5.2.	Exportación de datos de simulación	91
5.3.	Identificación paramétrica	91
5.4.	Herramienta de análisis, guardado en memoria y exportación de funciones de transferencia	93
6.	<i>Selección del algoritmo de control</i>	<i>96</i>
6.1.	Sintonización de un controlador PID	98
6.2.	Sintonización de la acción anticipativa	101
6.3.	Sintonización de un controlador por realimentación del estado	104
6.4.	Sintonización de un controlador predictivo DMC.....	106
6.5.	Configuración del control en cascada	107
7.	<i>Ensayos con control adaptativo y autoajutable.....</i>	<i>108</i>
7.1.	Configuración del control Gain Scheduling PI	109
7.2.	Configuración del control autoajutable RLS	110
7.3.	Configuración del control Gain Scheduling DMC.....	111
7.4.	Sintonización de controladores adaptativos.....	112
	Anexo B. Resultados experimentales de algoritmos de identificación.....	113
1.	<i>Experimento de identificación paramétrica por mínimos cuadrados</i>	<i>113</i>
1.1.	Descripción del experimento	113
1.2.	Configuración de la simulación	113
1.3.	Resultados.....	114
2.	<i>Experimento de identificación paramétrica por mínimos cuadrados recursivos</i>	<i>115</i>
2.1.	Descripción del experimento	115
2.2.	Configuración de la simulación	116
2.3.	Resultados.....	118

Índice de Tablas

Tabla 1. Datos de los parámetros del reactor CSTR con serpentín de refrigeración y condiciones de operación nominal	6
Tabla 2. Datos de los parámetros de la camisa de refrigeración y condiciones de operación nominal	8
Tabla 3. Rango de operación del reactor	12
Tabla 4. Algoritmo 1: Simulación del sistema controlado	15
Tabla 5. Algoritmo 2: Preparación del control DMC	58
Tabla 6. Algoritmo 3: Cálculo de la acción de control óptima mediante DMC	59
Tabla 7. Valores de conmutación del control adaptativo mediante ajuste por tabla	65
Tabla 8. Algoritmo 4: Actualización de los parámetros de un PI autoajutable mediante RLS y sintonización IMC	72
Tabla A1. Rango de medición de los sensores-transmisores	84
Tabla B1. Configuración de la simulación en el experimento de identificación paramétrica mediante LS (I)	114
Tabla B2. Resultados del experimento de identificación paramétrica LS	115
Tabla B3. Configuración de la simulación en el experimento de identificación paramétrica RLS	116
Tabla B4. Configuración de la simulación en el experimento de identificación paramétrica mediante LS (II)	117
Tabla B5. Resultados de la identificación paramétrica RLS	118

Índice de Figuras

Figura 1. Esquema de un reactor de tanque agitado continuo y refrigerado mediante un serpentín.....	3
Figura 2. Esquema de un reactor de tanque agitado continuo con camisa de refrigeración	7
Figura 3. Esquema de un reactor de tanque agitado continuo encamisado con recirculación de refrigerante .	9
Figura 4. Diagrama de flujo del algoritmo de simulación dinámica en Matlab	19
Figura 5. Diagrama de bloques de un modelo con estructura ARX	22
Figura 6. Diagrama de bloques de un modelo con estructura ARMAX	23
Figura 7. Diagrama de bloques de un modelo con estructura Box and Jenkins	24
Figura 8. Suavizado de datos mediante filtro de media móvil.....	26
Figura 9. Comparación entre proceso modelado y proceso identificado empíricamente para distintos modelos de reactor CSTR.....	27
Figura 10. Diagrama de bloques del sistema lineal completo	29
Figura 11. Realización en la GUI de una identificación paramétrica por cada punto de operación del ensayo	31
Figura 12. Herramienta de análisis y modificación de funciones de transferencia	33
Figura 13. Comparación de desempeño entre distintos tipos de acción derivativa.....	39
Figura 14. Diagrama de bloques de la implementación en la GUI implementado en la GUI	40
Figura 15. Comparación de desempeño entre distintos controladores PI obtenidos mediante sintonización IMC.....	42
Figura 16. Diagrama de bloques del control en lazo cerrado con acción anticipativa	43
Figura 17. Comparación de las respuestas temporales ante una perturbación para los diferentes modelos controlados por un PI con y sin acción anticipativa.....	44
Figura 18. Implementación en la GUI del control en lazo cerrado con acción anticipativa.....	45
Figura 19. Diagrama de bloques del algoritmo de control en cascada.....	46
Figura 20. Rechazo de una perturbación (temperatura del refrigerante a la entrada de la camisa del reactor) utilizando diferentes variaciones del algoritmo PI.....	47
Figura 21. Rechazo de una perturbación externa (temperatura a la entrada del reactor) para distintas estrategias de control	47
Figura 22. Desempeño de un controlador “ <i>state feedback</i> ” proporcional para distintos valores de μ	51
Figura 23. Desempeño de un controlador “ <i>state feedback</i> ” con acción integral para distintos valores de γ ..	51
Figura 24. Desempeño en el rechazo de perturbaciones de un controlador “ <i>state feedback</i> ” con acción integral para distintos valores de γ	52
Figura 25. Herramienta para la sintonización de controladores de realimentación de estado	53
Figura 26. Diagrama de funcionamiento de un controlador predictivo basado en modelo	55
Figura 27. Desempeño de un controlador DMC para distintos valores de λ	60

Figura 28. Comparación del rechazo de una perturbación para distintas variaciones del algoritmo DMC.....	60
Figura 29. Comparación de la respuesta temporal obtenida mediante control PID y DMC.....	61
Figura 30. Diagrama de bloques de un control adaptativo mediante ajuste por tabla	64
Figura 31. Diagrama de conmutación de la estrategia de control adaptativo de ajuste por tabla.....	65
Figura 32. Comparación entre las respuestas temporales obtenidas para el caso del reactor CSTR encamisado, aplicando control PI y DMC con y sin estrategia de ajuste por tabla.	67
Figura 33. Diagrama de bloques de un controlador autoajustable	68
Figura 34. Resultados de una identificación paramétrica mediante mínimos cuadrados recursivos	70
Figura 35. Respuesta temporal y evolución de los parámetros kc y Ti para un control PI autoajustable, aplicado al caso del reactor CSTR encamisado	73
Figura A1. Instalación del software. Inicio.....	79
Figura A2. Instalación del software. Selección de ruta de instalación	80
Figura A3. Instalación del software Matlab Runtime.....	80
Figura A4. Instalación del software. Resumen de la instalación	81
Figura A5. Ventana de inicio de la interfaz gráfica de usuario	81
Figura A6. Ventana principal de la interfaz gráfica de usuario.....	82
Figura A7. Evolución del estado interno del proceso obtenido mediante el botón “Variables de proceso”	83
Figura A8. Evolución del error y de la integral del error obtenidos mediante el botón “Índices de error”	83
Figura A9. Evolución de las señales eléctricas obtenidas mediante el botón “Señales eléctricas”	84
Figura A10. Ventana gráfica de configuración de la simulación	85
Figura A11. Opciones de gráficas	87
Figura A12. Guardado de datos en el modo de simulación dinámica	88
Figura A13. Ventana gráfica de identificación paramétrica.....	89
Figura A14. Importación de datos.....	90
Figura A15. Mensaje de confirmación de exportación correcta de datos y directorio de ubicación	91
Figura A16. Resultados de la identificación paramétrica mediante la opción “Identificación por puntos”	92
Figura A17. Modelos lineales obtenidos con una identificación paramétrica mediante la opción “Identificación por puntos”	93
Figura A18. Herramienta para el análisis de funciones de transferencia	94
Figura A19. Ventana gráfica de opciones de control	96
Figura A20. Previsualización de la respuesta temporal del proceso controlado para una referencia unitaria .	97
Figura A21. Ventana gráfica de información sobre el controlador seleccionado	97
Figura A22. Sintonización de un controlador PID.....	98
Figura A23. Configuración del tipo de acción proporcional.....	99
Figura A24. Configuración del tipo de acción derivativa.....	99
Figura A25. Configuración del tipo de acción derivativa con filtro	100
Figura A26. Diagrama de bode del proceso controlado en lazo abierto (I)	100
Figura A27. Diagrama de bode del proceso controlado en lazo abierto (II)	101
Figura A28. Sintonización de un controlador PI con acción anticipativa	102

Figura A29. Herramienta para sintonizar la acción anticipativa	103
Figura A30. Lazo de control configurado con acción anticipativa	104
Figura A31. Sintonización de un controlador por realimentación del estado con acción integral	104
Figura A32. Herramienta para sintonización de un controlador por realimentación del estado.....	105
Figura A33. Sintonización de un controlador predictivo DMC	106
Figura A34. Ventana gráfica de opciones de control para el reactor encamisado con estructura de control en cascada.....	107
Figura A35. Ventana gráfica de control adaptativo y autoajustable	109
Figura A36. Parámetros de ajuste para el control PI con estrategia “Gain Scheduling”	110
Figura A37. Parámetros de ajuste para el control autoajustable basado en RLS	111
Figura A38. Parámetros de ajuste para el control predictivo DMC con estrategia “Gain Scheduling”	111
Figura B1. Resultados de la identificación paramétrica LS	115
Figura B2. Resultados de la identificación paramétrica RLS y comparación con el método LS.....	119

Símbolos

\bar{a}	Una barra localizada sobre una letra indica un vector
\dot{b}	Un punto sobre una letra indica una variable derivada con respecto al tiempo
$Ceil(x)$	Redondeo al entero más próximo por exceso
cte	Valor constante
$f(x) _{x=x_0}$	Evaluación en el punto de equilibrio de una función de x
\bar{I}	Matriz identidad
$J^*(x)$	Valor óptimo de la función de coste $J(x)$
$J^*(\bar{x}^+)$	Valor óptimo de la función de coste $J(\bar{x})$, analizado en $\bar{x}(k + 1)$.
$\ln(\cdot)$	Logaritmo neperiano
$\bar{\bar{M}}$	Una doble barra localizada sobre una letra mayúscula indica una matriz
$\bar{\bar{M}}_{j,*}$	Fila j de la matriz M
$\bar{\bar{M}}(i, j: k)$	Elementos de M localizados en la fila i , entre las columnas desde j hasta k
$\min J(x)$	Valor mínimo de la función $J(x)$
P_0	Valor inicial de la variable P
s	Variable compleja usada en la transformada de Laplace
x^*	Valor que optimiza a una función objetivo $J(x)$
z	Variable compleja usada en la transformada Z
$Z\{\cdot\}$	Transformada Z de la función $\{\cdot\}$
Δ	Incremento
$(\cdot)^T$	Transpuesta de (\cdot)
$ \cdot $	Valor absoluto de (\cdot)
$\xi(\cdot)$	Ruido blanco de media cero y varianza σ^2
$\phi(s)$	Polinomio característico evaluado en s

Variables y propiedades fisicoquímicas

A	Área del reactor (m^2)
c_A	Concentración molar de reactivo (mol/m^3)
C_p	Capacidad calorífica a presión constante del reactivo y producto ($kJ/(kg\ ^\circ C)$)
E	Energía de activación (J/mol)
F	Caudal de reactivo y producto (m^3/s)
k	Constante cinética (s^{-1})
k_0	Factor de frecuencia (s^{-1})
K_{vs}	Coefficiente de caudal máximo de la válvula ($m^3 h^{-1} bar^{-1/2}$)
Q	Potencia calorífica liberada hacia el refrigerante (W)
r	Velocidad de reacción ($mol/(m^3 s)$)
R	Constante universal de los gases ideales ($8,314 J/(mol K)$)
T	Temperatura del reactor ($^\circ C$)
U	Coefficiente global de transferencia de calor ($W/(m^2\ ^\circ C)$)
V	Volumen del reactor (m^3)
V_c	Volumen de la camisa (m^3)
ΔH_R	Calor de reacción (J/mol)
$\Delta P_v(x)$	Caída de presión en la válvula (bar)
γ	Peso específico del refrigerante
ρ	Densidad del reactivo y producto (consideradas iguales) (kg/m^3)

Variables de modelos lineales e identificación

\bar{A}	Matriz de estados del modelo en espacio de estados
\bar{B}	Matriz de entrada del modelo en espacio de estados
\bar{C}	Matriz de salida del modelo en espacio de estados
\bar{D}	Matriz de transmisión directa del modelo en espacio de estados
\bar{E}	Vector de error de modelado
f_{pct}	Sensibilidad del bloque conversor de la salida del transmisor (en mA o V) a un valor porcentual
f_c	Conversión de la señal de control calculada en señal eléctrica (de % a V o mA)
f_{st}	Conversión de la señal eléctrica en magnitud física
f	Respuesta libre
\bar{g}	Vector de ganancias del modelo de respuesta en escalón
\overline{gd}	Vector de ganancias del modelo de respuesta en escalón para una entrada de perturbación
\bar{G}	Matriz dinámica
G	Función de transferencia
K	Ganancia estática
\bar{M}_c	Matriz de controlabilidad
N_1	Longitud del vector de estados
n_a	Orden de la parte autorregresiva de una estructura de modelo lineal en tiempo discreto
n_b	Orden de la parte exógena de una estructura de modelo lineal en tiempo discreto
n_c	Orden de la parte de media móvil de una estructura de modelo lineal en tiempo discreto
n_{det}	Número de pasos que tarda un cambio en la entrada en afectar a la señal de salida
\bar{x}	Vector de estados
\bar{X}	Base de datos de entrada al proceso
$\hat{y}(t+k t)$	Valor predicho de la variable "y" en el instante " $t+k$ " calculado en el instante actual, " t ", con $k \in \mathbb{N}^+$. Cada número natural se corresponde con un intervalo de tiempo igual al periodo de muestro
y	Salida del proceso
\bar{Y}	Base de datos de salida al proceso
$\bar{\Gamma}$	Matriz de varianza-covarianza
ϵ	Error de modelado
ζ	Constante de amortiguamiento
$\bar{\theta}$	Vector paramétrico
λ_o	Factor de olvido
τ	Constante de tiempo (<i>min</i>)
τ_m	Tiempo muerto
$v(t)$	Término de media móvil asociado al ruido en una estructura de tipo ARMAX
$\bar{\varphi}$	Vector de regresión

Variables de simulación y control

\bar{d}	Vector de perturbación
Dy	Derivada de la salida del proceso con respecto al tiempo
e_r	Error de seguimiento
u_D	Término de la acción derivativa de una estructura PID estándar
\bar{H}	Matriz de históricos
\bar{K}_c	Matriz de ganancia de realimentación
\bar{K}_I	Elementos de la matriz de ganancia de realimentación correspondientes a la integral del error
IE	Integral del error
K_c	Constante proporcional de una estructura PID estándar
K_{cd}	Constante proporcional de una estructura PID estándar en tiempo discreto
m	Horizonte de control
N_2	Longitud del vector de referencia
N_3	Longitud del vector de perturbación
n_1	Número de filas de la matriz de históricos
n_2	Número de columnas de la matriz de históricos
p	Horizonte de predicción
$PreC$	Precompensador
\bar{Q}	Matriz de ponderación de desviación respecto a la posición de equilibrio
\bar{r}	Vector de referencias
\bar{R}	Matriz de ponderación de la energía de control
\bar{t}_d	Vector de instantes de aplicación de las perturbaciones en la simulación
\bar{t}_r	Vector de instantes de aplicación de las referencias en la simulación
T_d	Tiempo derivativo (<i>min</i>)
T_{dd}	Tiempo derivativo de una estructura PID estándar en tiempo discreto (<i>min</i>)
T_{DMC}	Tiempo de muestreo del algoritmo de control DMC (<i>min</i>)
T_i	Tiempo integral (<i>min</i>)
T_{id}	Tiempo integral de una estructura PID estándar en tiempo discreto (<i>min</i>)
t_m	Tiempo de muestreo (<i>min</i>)
T_s	Tiempo de simulación (<i>min</i>)
u	Señal de control
u_{os}	Valor nominal de salida de un controlador proporcional
\bar{w}	Vector de referencias futuras
Δt_k	Paso de integración (<i>min</i>)
λ	Parámetro de ponderación de los esfuerzos de control

Subíndices

<i>a</i>	El subíndice <i>a</i> sobre una función de transferencia hace referencia al actuador (p.e. G_{av})
<i>c</i>	El subíndice <i>c</i> sobre un parámetro o variable fisicoquímica hace referencia al caudal de refrigeración (p.e. F_c)
<i>d</i>	El subíndice <i>d</i> sobre una variable de modelo lineal hace referencia a la perturbación (p.e. G_d)
<i>des</i>	El subíndice <i>des</i> sobre un parámetro o variable hace referencia al valor deseado (p.e. $K_{c,des}$)
<i>e</i>	El subíndice <i>e</i> sobre un parámetro o variable fisicoquímica hace referencia la entrada del sistema (p.e. T_{ce})
<i>j</i>	El subíndice <i>j</i> sobre un parámetro o variable fisicoquímica hace referencia al caudal de mezcla (p.e. T_j)
<i>k</i>	El subíndice <i>k</i> sobre una variable de simulación hace referencia al valor en el instante actual (p.e. r_k)
<i>máx</i>	El subíndice <i>máx</i> sobre una variable hace referencia al valor máximo alcanzable o valor máximo establecido (p.e. $p_{máx}$)
<i>os</i>	El subíndice <i>os</i> sobre una variable de control hace referencia al valor de “offset”
<i>out</i>	El subíndice <i>out</i> sobre una variable de simulación hace referencia al argumento de salida del algoritmo de simulación (p.e. \bar{H}_{out})
<i>p</i>	El subíndice <i>p</i> sobre una variable de modelo lineal hace referencia al proceso (p.e. G_p)
<i>r</i>	El subíndice <i>r</i> sobre un parámetro o variable fisicoquímica hace referencia al caudal recirculado (p.e. F_{rc})
<i>req</i>	El subíndice <i>req</i> sobre un parámetro o variable fisicoquímica hace referencia al caudal requerido (p.e. $F_{c,req}$)
<i>s</i>	El subíndice <i>s</i> sobre un parámetro o variable fisicoquímica hace referencia al caudal de purga (p.e. F_{cs}). El subíndice <i>s</i> sobre una constante de tiempo hace referencia una función de transferencia de segundo orden.
<i>T</i>	El subíndice <i>T</i> sobre una función de transferencia hace referencia a la combinación de las dinámicas dadas por f_c , G_{av} , G_v y G_p (p.e. G_T)
<i>Td</i>	El subíndice <i>Td</i> sobre una función de transferencia hace referencia al transmisor (p.e. G_{Td})
<i>v</i>	El subíndice <i>v</i> sobre una variable de modelo lineal hace referencia a la válvula de control (p.e. G_v)

ARMAX	Auto regresivo y exógeno de media móvil
ARX	Auto regresivo y exógeno
CSTR	Reactor de tanque agitado continuo
DMC	Control por matriz dinámica
GUI	Interfaz gráfica de usuario
LQR	Regulador lineal cuadrático
LS	Mínimos cuadrados
P	Proporcional
PI	Proporcional-integral
PID	Proporcional-integral-derivativo
RLS	Mínimos cuadrados recursivos
SISO	Sistema de una entrada y una salida
STR	Regulador autoajutable
ZOH	Retenedor de orden cero

1 INTRODUCCIÓN

El objetivo de este trabajo es crear una aplicación informática orientada al estudio, aprendizaje y puesta en práctica de nociones básicas y avanzadas de control de procesos químicos, proporcionando las herramientas necesarias para la simulación y el diseño de controladores de temperatura en un reactor químico de tanque agitado continuo (CSTR) exotérmico.

En el aspecto configurable, se incluyen diferentes posibilidades en la selección del algoritmo de control, con aplicación a tres casos de estudio:

- Reactor con camisa de refrigeración
- Reactor encamisado controlado en cascada
- Reactor con refrigeración por serpentín

El alcance de este Trabajo Fin de Máster implica dos aspectos principales: por un lado, la obtención de modelos no lineales que describan la dinámica de cada tipo de reactor CSTR considerado en el objetivo del proyecto y, por otro, el diseño de una interfaz gráfica de usuario (GUI) que presente una disposición coherente e intuitiva de sus elementos constituyentes. Dicha interfaz abarca aspectos fundamentales para el diseño de un lazo de control, como son la inclusión de una herramienta de identificación de sistemas lineales aproximados, la programación de los algoritmos de control más representativos, tanto a nivel industrial como académico y la simulación del proceso controlado, a partir de unos parámetros y condiciones iniciales establecidas. El diseño y programación de la interfaz gráfica, así como la creación de la aplicación “*standalone*” se ha llevado a cabo con Matlab [1].

Asimismo, este proyecto también contempla la programación de un algoritmo de simulación con tiempo de ejecución variable, de forma que proporcione los datos a una velocidad previamente fijada por el usuario, además de permitir una simulación en tiempo real.

Con respecto a los parámetros de control, se incluye la programación de un algoritmo PID con y sin acción anticipativa, una estructura de control en cascada, algoritmos de realimentación de estado, estrategias de control adaptativo y autoajutable, y un algoritmo de control predictivo basado en modelo (MPC). Otro aspecto del trabajo consiste en el ajuste de los parámetros de los algoritmos propuestos, ofreciendo al usuario, por medio de la interfaz gráfica, una solución funcional de los mismos. Por otro lado, para proporcionar al usuario documentación sobre la instalación y manejo de la aplicación, se ha desarrollado un manual que se incluye en el Anexo A.

Dado que el eje central de este trabajo es el control de procesos químicos, no se persigue que las variables de proceso se correspondan al de un caso concreto de aplicación real, priorizándose en su lugar, el carácter pedagógico en lo relativo al campo de estudio en cuestión.

1.1 Introducción

El reactor de tanque agitado continuo representa uno de los modelos de reactores clásicos más utilizados tradicionalmente en la industria. Consiste en un depósito donde se mezclan de manera homogénea los reactivos a partir de un agitador movido por un motor, al que se le incorpora un sistema de intercambio de calor para permitir la refrigeración, en caso de reacción exotérmica, o aportación energética, en caso de reacción endotérmica. Los reactivos y productos son alimentados y retirados continuamente y pueden estar presentes en forma de sólido, líquido, gas, o combinación de éstas, siendo la aplicación más común en sistemas de fase líquida a presiones bajas o medias [2].

Una de las ventajas de este tipo de reactores reside en la uniformidad de presión, composición y temperatura, lo que permite ser operados en condiciones isotermas, aun cuando el calor de reacción sea alto, resultando útil al

operar en intervalos pequeños de temperatura para reducir las reacciones secundarias [2]. Debido a la homogeneidad en la mezcla, se evitan puntos calientes en reacciones altamente exotérmicas, facilitando el control de temperatura y mejorándose la transferencia de calor. Además, los reactores CSTR permiten una operación económica cuando se opera con grandes volúmenes que requieren un tiempo de contacto elevado. Por otro lado, en comparación con otros tipos de reactores (como los tubulares), suelen requerirse volúmenes más grandes y, debido a la agitación necesaria de los reactivos para obtener una mezcla con temperatura homogénea, hay un consumo de energía extra, repercutiendo en mayores costes de operación [3].

Entre las diversas aplicaciones de este tipo de reactor, es de especial interés su utilización en la saponificación de grasas, polimerizaciones en suspensión y en emulsión y oxidación de ciclohexano. Además, se usan frecuentemente en la producción continua de hidrógeno, ya que las bacterias que producen este gas se suspenden bien en la mezcla agitada y sufren menos la resistencia a la transferencia de masa [4]. Desde un punto de vista más general, los reactores CSTR son útiles en reacciones de baja concentración y en sistemas heterogéneos donde un alto grado de mezcla da lugar a mayores tiempos de contacto entre fases [3].

Actualmente en la industria química, para alcanzar unos requisitos de seguridad y eficiencia económica, resulta imprescindible el diseño de un lazo de control, de forma que permita mantener determinadas variables del proceso en un nivel constante previamente establecido, aun en presencia de perturbaciones. Este controlador debe diseñarse en base a una información del proceso conocida, para lo que es necesaria la caracterización dinámica del proceso, que, generalmente, es de tipo no lineal (como es el caso del reactor CSTR).

En los siguientes capítulos se describe el modelado de los distintos casos de reactor CSTR considerados, así como el desarrollo de algoritmos para la simulación, identificación y control del proceso.

2 MODELADO DEL REACTOR DE TANQUE AGITADO CONTINUO

El reactor de tanque agitado continuo (CSTR) representa un modelo de reactor químico alimentado por un caudal de reactivo constante, en cuyo interior se realiza una mezcla perfectamente uniforme, de modo que la concentración de producto a la salida es igual a la del interior del reactor. Debido a su simplicidad, resultan muy convenientes para ilustrar los principios básicos del modelado de reactores químicos [5].

2.1 Modelo del reactor CSTR con serpentín de refrigeración

A continuación, se detalla el procedimiento de modelado de un reactor CSTR refrigerado mediante un serpentín de extracción de calor (Figura 1).

Asumiendo una reacción de la forma $A \rightarrow B$ se puede considerar una ecuación de primer orden para la velocidad de reacción:

$$r = kc_A \quad (1)$$

donde k es la constante cinética y c_A la concentración molar de reactivo.

Mediante la ecuación de Arrhenius se tiene que la constante k viene dada por:

$$k(T) = k_0 e^{-E/RT} \quad (2)$$

donde k_0 es el factor de frecuencia, E es la energía de activación y R es la constante universal de los gases ideales.

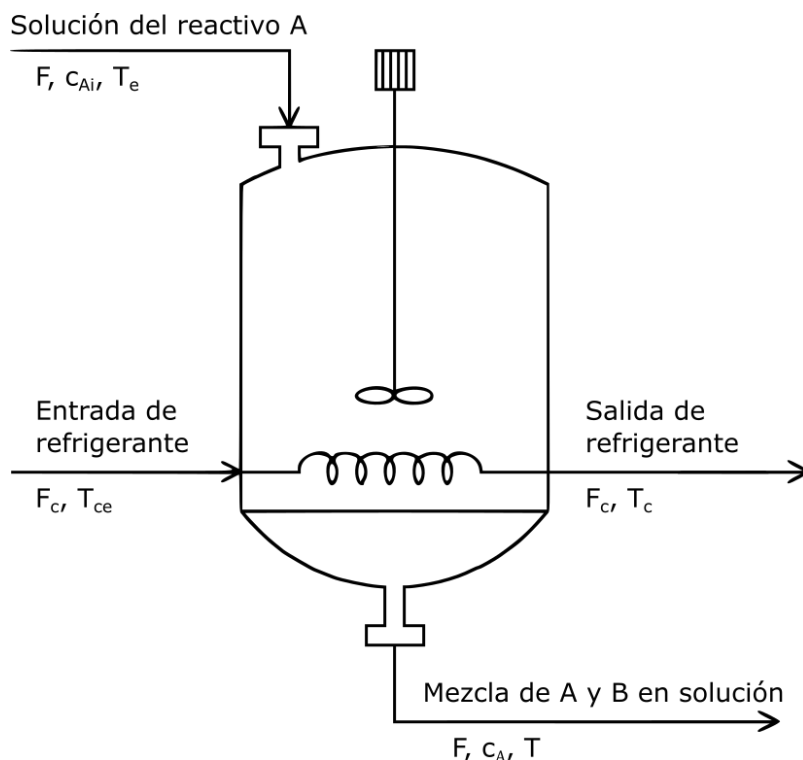


Figura 1. Esquema de un reactor de tanque agitado continuo y refrigerado mediante un serpentín

Para obtener un modelo matemático sencillo es necesario realizar algunas simplificaciones, que se detallan a continuación:

- La pureza del reactivo a la entrada del reactor es del 100%.
- La extracción de calor de la reacción exotérmica se lleva a cabo mediante un serpentín por el que circula un caudal de líquido de refrigeración.
- La mezcla dentro del reactor está perfectamente agitada.
- La densidad del reactivo a la entrada es igual a la densidad del producto a la salida.
- Existe un control de nivel en el tanque asociado al caudal de salida, de forma que el volumen de la mezcla en el interior se mantiene constante en el tiempo.
- Se desprecian las pérdidas de calor hacia el exterior.
- Los calores específicos a presión constante (C_p) no cambian con la temperatura dentro del intervalo de operación, y coinciden en las corrientes del proceso a la entrada y salida del reactor.

De esta forma, es posible realizar un balance de masa en el reactor:

$$V \frac{dc_A}{dt} = F(c_{Ae} - c_A) - Vkc_A \quad (3)$$

El calor cedido por el reactor hacia el refrigerante por unidad de tiempo (potencia calorífica) puede modelarse como:

$$Q = \rho_c C_{pc} F_c (T_c - T_{ce}) = UA \frac{T_c - T_{ce}}{\ln\left(\frac{T - T_{ce}}{T - T_c}\right)} \quad (4)$$

de donde se obtiene:

$$\ln\left(\frac{T - T_{ce}}{T - T_c}\right) = \frac{UA}{\rho_c C_{pc} F_c} \quad (5)$$

Por lo que Q puede expresarse como:

$$Q = \rho_c C_{pc} F_c \left(1 - e^{\frac{-UA}{\rho_c C_{pc} F_c}}\right) (T - T_{ce}) \quad (6)$$

Realizando un balance de energía se obtiene la siguiente ecuación diferencial:

$$V\rho C_p \frac{dT}{dt} = \rho C_p F (T_e - T) + (-\Delta H_R) V k c_A - \rho_c C_{pc} F_c \left(1 - e^{\frac{-UA}{\rho_c C_{pc} F_c}}\right) (T - T_{ce}) \quad (7)$$

Las ecuaciones (3) y (7) componen un sistema de dos ecuaciones diferenciales que describen la dinámica del reactor CSTR con refrigeración por serpentín. Las variables de salida son la temperatura del reactor y la concentración del reactivo A en el caudal saliente, y las entradas son el caudal de refrigerante, su temperatura de entrada, la temperatura de la corriente de alimentación al reactor, el caudal del proceso y la concentración a la entrada. En la Tabla 1 se muestran los parámetros del modelo implementado en la interfaz gráfica, así como las condiciones iniciales.

La temperatura del refrigerante y de la corriente de alimentación, así como el caudal del proceso y la concentración de reactivo a la entrada, son posibles perturbaciones.

Considerando que el caudal de refrigeración puede manipularse, dentro de un determinado intervalo, mediante una válvula de control automática situada a la entrada del serpentín, se tiene:

$$F_c = f(x)K_{vs} \sqrt{\frac{\Delta P_v(x)}{\gamma}} \quad (8)$$

donde:

- K_{vs} es el coeficiente de caudal máximo de la válvula y se obtiene cuando $x = 1$ y $f(x) = 1$.
- $f(x)$ es la característica inherente de la válvula.
- $K_v = K_{vs}f(x)$ es el coeficiente de válvula; varía con la abertura de la válvula en el intervalo $x \in [0,1]$.
- $\Delta P_v(x)$ es la caída de presión en la válvula.
- γ es el peso específico del refrigerante (se tomará igual a 1, por tratarse de agua).

Realizando las siguientes simplificaciones:

$$\Delta P_v(x) = cte \quad (9)$$

$$f(x) = x \quad (10)$$

La ecuación de la válvula puede aproximarse como sigue:

$$F_c = x \cdot F_{c,m\acute{a}x} \quad (11)$$

De forma que puede establecerse una relación entre la abertura de la válvula y el caudal solicitado correspondiente a dicho valor de abertura.

Para dotar al modelo de una mayor similitud con el proceso real, se ha añadido una dinámica de primer orden con una constante de tiempo (τ_v) de unos pocos segundos:

$$\tau_v \frac{dF_c}{dt} + F_c = F_{c,req} \quad (12)$$

donde $F_{c,req}$ es el caudal de refrigeración que se solicita con cada valor de abertura de la válvula. Nótese que al variar la abertura de la válvula (variable manipulable) se varía el caudal de refrigerante. De este modo, una nueva abertura (deseada) implicará un nuevo caudal (solicitado). Asimismo, la válvula modelada es de tipo “*air-to-open*”, es decir, la abertura de la misma aumenta de manera directamente proporcional a la señal de control.

Tabla 1. Datos de los parámetros del reactor CSTR con serpentín de refrigeración y condiciones de operación nominal

Variable	Condiciones de operación nominal
Concentración de reactivo a la salida (c_A)	55,50 mol/m ³
Temperatura del reactor (T)	356,43 K (83,28 °C)
Temperatura de refrigeración a la salida (T_c)	356,41 K (83,26 °C)
Caudal de reactivo/producto (F)	7,50 · 10 ⁻⁴ m ³ /s
Caudal de refrigerante (F_c)	1,50 · 10 ⁻³ m ³ /s
Concentración de reactivo a la entrada (c_{Ae})	930,00 mol/m ³
Temperatura a la entrada del reactor (T_e)	315,00 K (41,85 °C)
Temperatura de refrigeración a la entrada (T_{ce})	310,00 K (36,85 °C)
Volumen del reactor (V)	0,10 m ³
Término de transferencia de calor (UA)	4,88 · 10 ⁴ W/K
Factor de frecuencia de la reacción (k_0)	8,75 · 10 ⁸ s ⁻¹
Energía de activación (E)	67,34 · 10 ³ J/mol
Calor de reacción ($-\Delta H$)	8,37 x 10 ⁵ J/mol
Densidad (ρ)	1,00 · 10 ³ kg/m ³
Calor específico (C_p)	4,19 · 10 ³ J/(kg K)
Peso específico del refrigerante (γ)	1,00

2.2 Modelo del reactor CSTR encamisado

Suponiendo que la extracción de calor de la reacción exotérmica se lleva a cabo mediante una camisa de refrigeración (Figura 2) en lugar de a través de un serpentín, se asume lo siguiente:

- La pureza del reactivo a la entrada del reactor es del 100%
- La mezcla dentro del reactor está perfectamente agitada.
- El fluido de refrigeración en el interior de la camisa tiene una temperatura uniforme y la temperatura del caudal de refrigeración a la salida es igual a la temperatura en el interior de ésta.
- La densidad del reactivo a la entrada es igual a la densidad del producto a la salida.
- Existe un control de nivel en el tanque asociado al caudal de entrada, de forma que el volumen de la mezcla en el interior se mantiene constante en el tiempo.
- Se desprecian las pérdidas de calor hacia el exterior.
- Los calores específicos a presión constante (C_p) no cambian con la temperatura dentro del intervalo de operación, y coinciden en las corrientes del proceso a la entrada y salida del reactor.

El balance de masa en el interior del reactor no varía con respecto al caso del reactor refrigerado mediante serpentín, por lo que se tiene:

$$V \frac{dc_A}{dt} = F(c_{Ae} - c_A) - Vkc_A \quad (13)$$

El calor que se extrae del reactor por la camisa de refrigeración puede modelarse mediante la siguiente expresión:

$$Q = UA(T - T_c) \quad (14)$$

Por lo que la evolución de la temperatura en el reactor viene dada por:

$$V\rho C_p \frac{dT}{dt} = \rho C_p F(T_e - T) + (-\Delta H_R)Vkc_A + UA(T_c - T) \quad (15)$$

Y la evolución de la temperatura en la camisa de refrigeración puede expresarse como:

$$\frac{dT_c}{dt} = \frac{F_c}{V_c}(T_{ce} - T_c) + \frac{UA}{\rho_c C_{pc} V_c}(T - T_c) \quad (16)$$

Las variables de entrada y salida del proceso son las mismas que en el caso del reactor con serpentín de refrigeración.

Los parámetros de la camisa y las condiciones iniciales del modelo implementado en GUI se muestran en la Tabla 2. Nótese que el punto de equilibrio difiere respecto al caso del reactor refrigerado mediante serpentín (Tabla 1).

Al igual que en el reactor con serpentín de refrigeración, existe una válvula de control localizada a la entrada de la camisa y modelada mediante la ecuación (12).

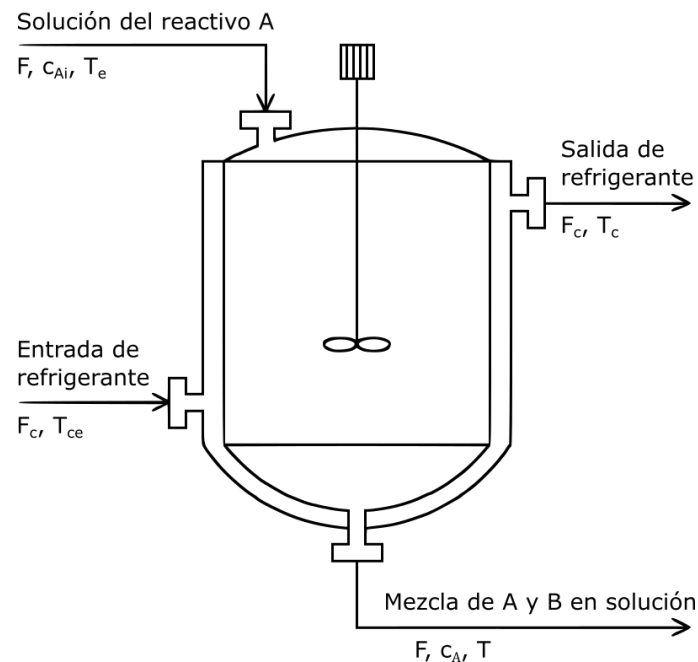


Figura 2. Esquema de un reactor de tanque agitado continuo con camisa de refrigeración

Tabla 2. Datos de los parámetros de la camisa de refrigeración y condiciones de operación nominal

Variable	Condiciones de operación nominal
Concentración de reactivo a la salida (c_A)	42,30 mol/m ³
Temperatura del reactor (T)	360,96 K (87,81 °C)
Temperatura de refrigeración a la salida (T_c)	355,16 K (82,01 °C)
Caudal de refrigerante (F_c)	1,50 · 10 ⁻³ m ³ /s
Temperatura de refrigeración a la entrada (T_{ce})	310,00 K (36,85 °C)
Término de transferencia de calor (UA)	4,88 · 10 ⁴ W/K
Volumen de la camisa del reactor (V_c)	0,01 m ³
Densidad (ρ_c)	1,00 · 10 ³ kg/m ³
Calor específico (C_c)	4,19 · 10 ³ J/(kg K)

2.3 Modelo del reactor CSTR encamisado con recirculación de refrigerante

La adición de un lazo de recirculación de caudal en la camisa (Figura 3) permite obtener un mejor comportamiento, ya que minimiza los efectos de:

- Generación de puntos fríos y calientes, que conducen a una pobre homogeneización de la temperatura del fluido en la camisa de refrigeración.
- Tiempo muerto y ganancia del proceso que serían variables.
- Bajo coeficiente global de transferencia de calor a caudales bajos.
- Ensuciamiento a caudales bajos.

Para el modelado de este proceso se tienen en cuenta las mismas consideraciones tomadas para el caso del reactor encamisado, además de las siguientes:

- Existe una bomba que mantiene constante el caudal de refrigeración en la camisa.
- En la zona de impulsión de la bomba, una vez derivado el caudal de recirculación, existe, en la purga de refrigerante, una válvula reguladora de presión aguas arriba de la misma.
- En la tubería de entrada a la camisa, aguas abajo del nodo de recirculación, existe una válvula que puede manipularse para variar el caudal de refrigerante fresco y así controlar la temperatura dentro del reactor.

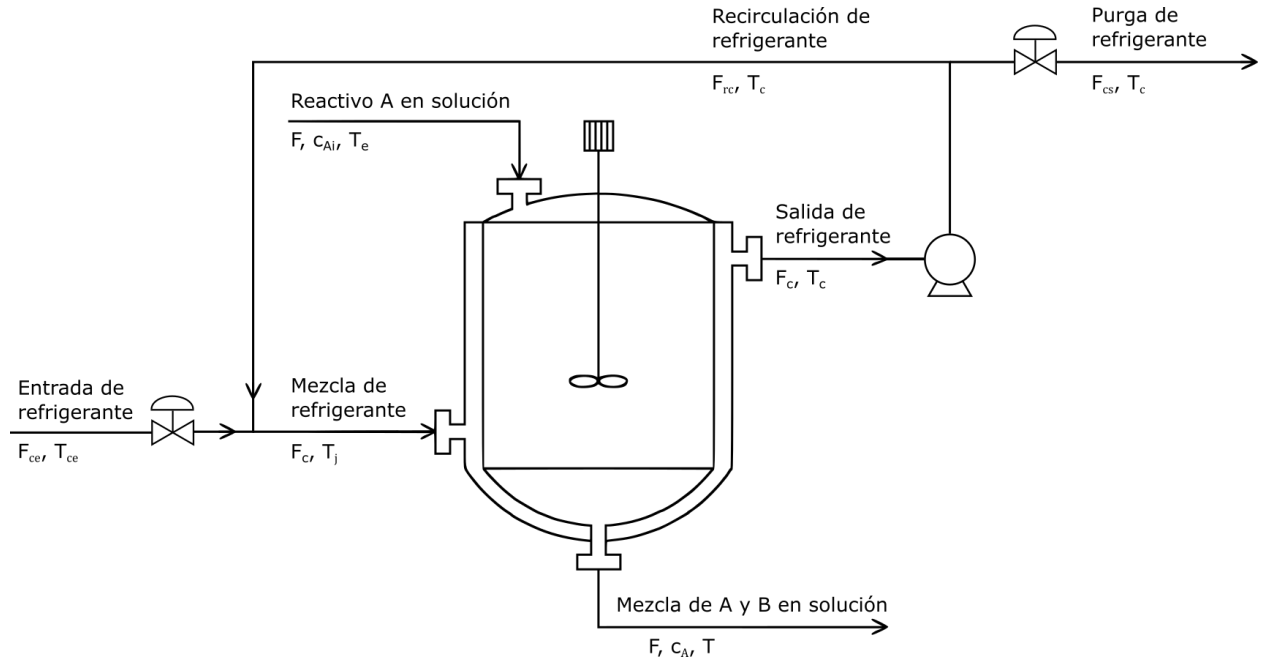


Figura 3. Esquema de un reactor de tanque agitado continuo encamisado con recirculación de refrigerante

La evolución de la temperatura del caudal de refrigerante viene dada por la siguiente ecuación diferencial:

$$\frac{dT_c}{dt} = \frac{F_c}{V_c} (T_j - T_c) + \frac{UA}{\rho_c C_{pc} V_c} (T - T_c) \quad (17)$$

donde T_j es la temperatura resultante de la mezcla entre el caudal de refrigeración de entrada al proceso y la fracción de flujo recirculado, procedente de la salida de la camisa.

Realizando un balance de masa en los nodos se tiene:

$$F_c = F_{ce} + F_{rc} \quad (18)$$

$$F_c = F_{cs} + F_{rc} \quad (19)$$

donde F_{rc} es el caudal recirculado y F_{cs} el caudal de purga a través de la válvula reguladora de presión.

El balance de energía del proceso proporciona la siguiente expresión:

$$\rho_c F_{ce} C_{pc} (T_{ce} - T_{ref}) + \rho_c F_{rc} C_{pc} (T_c - T_{ref}) = \rho_c F_c C_{pc} (T_j - T_{ref}) \quad (20)$$

Tomando como temperatura de referencia $T_{ref} = 273 \text{ K}$, se tiene:

$$F_c T_j = F_{ce} T_{ce} + F_{rc} T_c \quad (21)$$

$$T_j = \frac{F_{ce} T_{ce} + F_{rc} T_c}{F_c} = T_c + \frac{F_{ce}}{F_c} (T_{ce} - T_c) \quad (22)$$

Finalmente, sustituyendo T_j en la ecuación (17), se tiene:

$$\frac{dT_c}{dt} = \frac{F_{ce}}{V_c} (T_{ce} - T_c) + \frac{UA}{\rho_c C_{pc} V_c} (T - T_c) \quad (23)$$

Nótese que este sistema de ecuaciones coincide con el descrito en el apartado 2.2, aunque esta nueva configuración da como resultado una mayor similitud entre el proceso real y el modelo matemático. Esto es debido, en parte, a una menor variación del coeficiente global de transferencia de calor, obtenido al mantener un caudal de refrigeración constante a través de la camisa. En la GUI, se han implementado las ecuaciones correspondientes al caso con recirculación, sin embargo, se ha optado por representar en la misma el esquema correspondiente a la Figura 2, con el objetivo de simplificar la información visual de la ventana principal.

2.4 Aproximación de retardos presentes en el proceso real

Las ecuaciones anteriormente descritas han sido obtenidas a partir de simplificaciones y suposiciones iniciales, con el objetivo de obtener un equilibrio entre semejanza con la realidad y sencillez en la integración con métodos numéricos. Como consecuencia, las simulaciones realizadas no muestran el efecto de tiempo muerto característico de este tipo de procesos.

Para corregir esta desviación en el modelado, es posible incluir en el sistema de ecuaciones diferenciales la dinámica de la pared del reactor, de manera que se proporciona una nueva capacidad térmica al modelo, que conduciría a un sistema de tercer orden. Sin embargo, para no aumentar con ello la complejidad del sistema, se ha optado por incluir un retraso en los datos de entrada del simulador. Este retraso puro se ha dimensionado para incluir también la no linealidad asociada a una mezcla imperfecta y al retardo debido a una posición alejada del sensor-transductor, así como al efecto debido a algunas de las hipótesis simplificadoras consideradas en el modelado.

Además, para dotar al sistema de un mayor realismo, se ha dado un retraso diferente a cada variable interna del proceso (temperatura del reactor, concentración a la salida y temperatura de refrigeración), de forma que se obtienen tiempos muertos distintos para cada relación estado interno/entrada.

2.5 Linealización del modelo en torno al punto de funcionamiento lineal

Para realizar un análisis de la controlabilidad del proceso se ha realizado una previa linealización del modelo del reactor en torno al punto de operación nominal. Para ello, se han calculado las matrices Jacobianas de la función vectorial multivariable que describe la dinámica del sistema con respecto al estado del mismo y a la entrada, resultando en una expresión en el espacio de estados.

A continuación, se muestra el proceso de linealización del modelo de reactor CSTR encamisado. Partiendo del siguiente sistema de ecuaciones diferenciales:

$$\frac{dc_A}{dt} = f_1(c_A, T, T_c, F_c) \quad (24)$$

$$\frac{dT}{dt} = f_2(c_A, T, T_c, F_c) \quad (25)$$

$$\frac{dT_c}{dt} = f_3(c_A, T, T_c, F_c) \quad (26)$$

Se obtienen las matrices que describen la evolución del estado de un sistema lineal aproximado, a partir de las matrices Jacobianas:

$$\bar{\bar{A}} = \begin{bmatrix} \frac{\partial f_1}{\partial c_A} & \frac{\partial f_1}{\partial T} & \frac{\partial f_1}{\partial T_c} \\ \frac{\partial f_2}{\partial c_A} & \frac{\partial f_2}{\partial T} & \frac{\partial f_2}{\partial T_c} \\ \frac{\partial f_3}{\partial c_A} & \frac{\partial f_3}{\partial T} & \frac{\partial f_3}{\partial T_c} \end{bmatrix} \bigg|_{\substack{c_A=c_{A0} \\ T=T_0 \\ T_c=T_{c0} \\ F_c=F_{c0}}} \quad (27)$$

$$\bar{\bar{B}} = \begin{bmatrix} \frac{\partial f_1}{\partial F_c} \\ \frac{\partial f_2}{\partial F_c} \\ \frac{\partial f_3}{\partial F_c} \end{bmatrix} \bigg|_{\substack{c_A=c_{A0} \\ T=T_0 \\ T_c=T_{c0} \\ F_c=F_{c0}}} \quad (28)$$

Seguidamente, se muestran los sistemas obtenidos para cada modelo de reactor:

- Reactor CSTR con serpentín de refrigeración

$$\bar{\bar{A}} = \begin{bmatrix} -\frac{F}{V} - k_0 e^{-\frac{E}{RT_0}} & -\frac{E}{RT_0^2} c_{A0} k_0 e^{-\frac{E}{RT_0}} \\ \frac{(-\Delta Hr)}{\rho C_p} k_0 e^{-\frac{E}{RT_0}} & -\frac{F}{V} + \frac{(-\Delta Hr)}{\rho C_p} \frac{E}{RT_0^2} c_{A0} k_0 e^{-\frac{E}{RT_0}} - \frac{\rho_c C_{pc} F_{c0}}{V C_p \rho} \left(1 - e^{\frac{-UA}{\rho_c C_{pc} F_{c0}}}\right) \end{bmatrix} \quad (29)$$

$$\bar{\bar{B}} = \begin{bmatrix} 0 \\ \frac{\rho_c C_{pc}}{\rho C_p V} (T_0 - T_{ce}) \left(e^{-\frac{UA}{\rho_c C_{pc} F_{c0}}} \left(1 + \frac{UA}{\rho_c C_{pc} F_{c0}}\right) - 1 \right) \end{bmatrix} \quad (30)$$

$$\begin{bmatrix} \Delta \dot{c}_A \\ \Delta \dot{T} \end{bmatrix} = \bar{\bar{A}} \cdot \begin{bmatrix} \Delta c_A \\ \Delta T \end{bmatrix} + \bar{\bar{B}} \cdot \Delta F_c \quad (31)$$

- Reactor CSTR encamisado

$$\bar{\bar{A}} = \begin{bmatrix} -\frac{F}{V} - k_0 e^{-\frac{E}{RT_0}} & -\frac{E}{RT_0^2} c_{A0} k_0 e^{-\frac{E}{RT_0}} & 0 \\ \frac{(-\Delta Hr)}{\rho C_p} k_0 e^{-\frac{E}{RT_0}} & -\frac{F}{V} - \frac{UA}{V \rho C_p} + \frac{(-\Delta Hr)}{\rho C_p} \frac{E}{RT_0^2} c_{A0} k_0 e^{-\frac{E}{RT_0}} & \frac{UA}{V \rho C_p} \\ 0 & \frac{UA}{V_c \rho_c C_{pc}} & -\frac{F_{c0}}{V_c} - \frac{UA}{V_c \rho_c C_{pc}} \end{bmatrix} \quad (32)$$

$$\bar{\bar{B}} = \begin{bmatrix} 0 \\ 0 \\ \frac{T_{ce} - T_{c0}}{V_c} \end{bmatrix} \quad (33)$$

$$\begin{bmatrix} \Delta \dot{c}_A \\ \Delta \dot{T} \\ \Delta \dot{T}_c \end{bmatrix} = \bar{\bar{A}} \cdot \begin{bmatrix} \Delta c_A \\ \Delta T \\ \Delta T_c \end{bmatrix} + \bar{\bar{B}} \cdot \Delta F_c \quad (34)$$

- Reactor CSTR encamisado con recirculación del caudal de refrigeración:

$$\bar{\bar{A}} = \begin{bmatrix} -\frac{F}{V} - k_0 e^{-\frac{E}{RT_0}} & -\frac{E}{RT_0^2} c_{A0} k_0 e^{-\frac{E}{RT_0}} & 0 \\ \frac{(-\Delta Hr)}{\rho C_p} k_0 e^{-\frac{E}{RT_0}} & -\frac{F}{V} - \frac{UA}{V \rho C_p} + \frac{(-\Delta Hr)}{\rho C_p} \frac{E}{RT_0^2} c_{A0} k_0 e^{-\frac{E}{RT_0}} & \frac{UA}{V \rho C_p} \\ 0 & \frac{UA}{V_c \rho_c C_{pc}} & -\frac{F_{ce0}}{V_c} - \frac{UA}{V_c \rho_c C_{pc}} \end{bmatrix} \quad (35)$$

$$\bar{\bar{B}} = \begin{bmatrix} 0 \\ 0 \\ \frac{T_{ce} - T_{c0}}{V_c} \end{bmatrix} \quad (36)$$

$$\begin{bmatrix} \Delta \dot{c}_A \\ \Delta \dot{T} \\ \Delta \dot{T}_c \end{bmatrix} = \bar{\bar{A}} \cdot \begin{bmatrix} \Delta c_A \\ \Delta T \\ \Delta T_c \end{bmatrix} + \bar{\bar{B}} \cdot \Delta F_{ce} \quad (37)$$

Una vez obtenidas las matrices y sustituidas las variables por los valores proporcionados en las Tablas 1 y 2, se ha analizado la controlabilidad del sistema lineal resultante, con el objetivo de comprobar si es posible el diseño de un controlador alrededor del punto de operación nominal del reactor. Dicho análisis se ha realizado comprobando el rango de la matriz de controlabilidad:

$$\bar{\bar{M}}_C = [\bar{\bar{B}} \quad \bar{\bar{A}}\bar{\bar{B}} \quad \bar{\bar{A}}^2\bar{\bar{B}} \quad \dots \quad \bar{\bar{A}}^{N_1-1}\bar{\bar{B}}] \quad (38)$$

donde N_1 es la dimensión del vector de estados, que tiene un orden de $N_1 = 2$ en el caso del reactor con serpentín (temperatura del reactor y concentración de reactivo a la salida). En el reactor encamisado, se añade la dinámica de la temperatura de refrigeración, resultando en un vector de estados con un orden de $N_1 = 3$.

El resultado de la ecuación (38), para cada caso estudiado, coincide con la dimensión del correspondiente vector de estados N_1 , indicando que los sistemas son completamente controlables. Cabe destacar que el estudio de la controlabilidad de los modelos no lineales desarrollados en este capítulo es mucho más complejo, por lo que el resultado de este análisis tan sólo es válido de manera local y en el entorno cercano al punto nominal. Los límites de operación de las variables del proceso se recogen en la Tabla 3.

Tabla 3. Rango de operación del reactor

Variable	Valor mínimo	Valor máximo
Concentración de reactivo a la salida (c_A)	9,5 mol/m ³	74 mol/m ³
Temperatura del reactor (T)	351,5 K (78,35 °C)	387,8 K (114,65 °C)
Temperatura de refrigeración a la salida (T_c)	345,6 K (72,45 °C)	383,15 K (110 °C)
Caudal de refrigerante (F_c)	$7 \cdot 10^{-4}$ m ³ /s	$2 \cdot 10^{-3}$ m ³ /s

3 SIMULACIÓN DEL PROCESO

3.1 Simulación de las ecuaciones mediante integración numérica

Para integrar las ecuaciones del modelo en la interfaz gráfica se han implementado los métodos de integración numérica de Euler y Runge-Kutta de orden 4, debido que se tratan de métodos de paso fijo, de sencilla implementación y que, generalmente, proporcionan resultados precisos con modelos dinámicos simples.

El algoritmo de Runge-Kutta es, generalmente, más preciso que el de Euler, aunque requiere un mayor esfuerzo computacional para su ejecución. Es por ello que se deja a elección del usuario el método utilizado por la aplicación para integrar las ecuaciones del modelo, a partir de una opción localizada dentro de la interfaz.

A continuación, se detallan ambos algoritmos.

- Euler:

Dada una ecuación diferencial expresada de la forma:

$$\frac{dy}{dt} = f(y, H(y, u(t)), u(t)) \quad (39)$$

Y dadas unas condiciones iniciales, puede obtenerse el valor en el instante siguiente:

$$y_{k+1} = y_k + \Delta t_k \cdot f(y_k, H(y_k, u_k), u_k) \quad (40)$$

- Runge-Kutta orden 4

Mediante este método se tiene:

$$y_1 = y_k \quad (41)$$

$$u_1 = u(t_k) \quad (42)$$

$$y_2 = y_k + \frac{1}{2} \Delta t_k w_1 \quad (43)$$

$$y_3 = y_k + \frac{1}{2} \Delta t_k w_2 \quad (44)$$

$$y_4 = y_k + \Delta t_k w_3 \quad (45)$$

$$u_2 = u_3 = u\left(t_k + \frac{1}{2} \Delta t_k\right) \quad (46)$$

$$u_4 = (t_k + \Delta t_k) \quad (47)$$

$$z_i = H(y_i, u_i) \quad (48)$$

$$w_i = f(y_i, z_i, u_i) \quad (49)$$

$$y_{k+1} = y_k + \Delta t_k \cdot \left(\frac{1}{6} w_1 + \frac{1}{3} w_2 + \frac{1}{3} w_3 + \frac{1}{6} w_4\right) \quad (50)$$

Debido a la necesidad de integrar las ecuaciones del modelo junto con las del controlador, y con el objeto de simular el proceso a lo largo de un tiempo previamente establecido, se ha empleado el algoritmo descrito a continuación:

- Las entradas son: el paso de integración Δt_k , el tiempo de simulación T_s , los valores iniciales del vector de estados \bar{x}_0 , de la salida y_0 , de la señal de control u_0 y de la referencia r_0 , los vectores de referencia \bar{r} y perturbación \bar{d} y sus respectivos vectores de aplicación \bar{t}_r y \bar{t}_d .
- Paso 1: Se calcula el número de ciclos de simulación, redondeando al entero más próximo (por exceso) el resultado de la siguiente expresión:

$$N_s = \frac{T_s}{\Delta t_k} \quad (51)$$

Se asigna a la variable k el valor $k = 0$ y se crea una matriz nula, H , donde se registrarán los datos de la simulación en cada ciclo.

- Paso 2: Se asigna a la variable de tiempo el valor $t = 0$. Se actualizan las variables de estado \bar{x}_k , la salida y_k , la señal de control u_k y la referencia r_k con los valores iniciales. Se inicializa la variable de perturbación con un valor de $d_k = 0$. Se actualiza la matriz de históricos.
- Paso 3: Se actualiza el índice $k : k = k + 1$.
- Paso 4: Se calcula el tiempo correspondiente al ciclo de simulación:

$$t = k \cdot \Delta t_k \quad (52)$$

- Paso 5: Se crea un índice j al que se le asigna el valor $j = 1$.
- Paso 6: Si el tiempo transcurrido es igual o superior al instante en el que se aplica el cambio de consigna, $t \geq \bar{t}_{r,j}$, se asigna a la referencia actual el valor del elemento correspondiente del vector de referencias $r_k = \bar{r}_j$.
- Paso 7: Si el índice j es igual al número de elementos del vector de referencias \bar{r} , se continúa en el siguiente paso. En caso contrario, se actualiza el índice $j = j + 1$ y se retrocede al paso 6.
- Paso 8: Se reinicia el valor de $j : j = 1$.
- Paso 9: Si el tiempo transcurrido es igual o superior al instante en el que se aplica la perturbación, $t \geq \bar{t}_{d,j}$, se asigna a la perturbación actual el valor del elemento correspondiente del vector de perturbaciones $d_k = \bar{d}_j$.
- Paso 10: Si el índice j es igual al número de elementos del vector de perturbaciones \bar{d} , se continúa en el siguiente paso. En caso contrario, se actualiza el índice $j = j + 1$ y se retrocede al paso 9.
- Paso 11: Se actualiza la señal de control. La ejecución de este paso depende del tipo de algoritmo de control utilizado.
- Paso 12: Se obtiene el vector de estados y la salida en el instante siguiente, mediante la integración numérica por Euler o Runge-Kutta de orden 4.
- Paso 13: Se actualiza la matriz de históricos con los nuevos datos obtenidos de la integración numérica y la señal de control calculada en el paso 11.
- Paso 14: Si $k = N_s$ se finaliza el algoritmo. En caso contrario, se retrocede al paso 3.
- Salidas: Matriz de históricos con los datos de la simulación.

La Tabla 4 (Algoritmo 1: Simulación del sistema controlado) recoge esta descripción de un modo simbólico.

Tabla 4. Algoritmo 1: Simulación del sistema controlado

Entrada	:	$\bar{x}_0 \in \mathbb{R}^{N_1}, \bar{r} \in \mathbb{R}^{N_2}, \bar{t}_r \in \mathbb{R}^{N_2}, \bar{d} \in \mathbb{R}^{N_3}, \bar{t}_d \in \mathbb{R}^{N_3}, \mathbf{u}_0 \in \mathbb{R}, \mathbf{y}_0 \in \mathbb{R}, \mathbf{r}_0 \in \mathbb{R}, \Delta t_k \in \mathbb{R}^+, \mathbf{T}_s \in \mathbb{R}^+$
Paso 1	:	$N_s = \text{ceil}\left(\frac{T_s}{\Delta t_k}\right), k \in \mathbb{N}^+ : k = 0, \bar{H} \in \mathcal{M}_{n_1 \times n_2}(\mathbb{R}) : \bar{H} = \bar{0}$
Paso 2	:	Para $k = 0 : \bar{x}_k = \bar{x}_0, y_k = y_0, u_k = u_0, r_k = r_0, d_k = 0, t \in \mathbb{R}^+ : t = 0$. Actualizar la matriz de históricos $\bar{H}(k+1, 1:(n_2-2)) = \bar{x}_k, \bar{H}(k+1, (n_2-1)) = y_k, \bar{H}(k+1, n_2) = u_k$
Paso 3	:	$k = k + 1$
Paso 4	:	Calcular $t = k \cdot \Delta t_k$
Paso 5	:	$j = 1$
Paso 6	:	Si $t \geq \bar{t}_{r,j}$ entonces $r_k = \bar{r}_j$
Paso 7	:	Si $j = N_2$ Continuar. En otro caso $j = j + 1$ e ir a Paso 6
Paso 8	:	$j = 1$
Paso 9	:	Si $t \geq \bar{t}_{d,j}$ entonces $d_k = \bar{d}_j$
Paso 10	:	Si $j = N_3$ Continuar. En otro caso $j = j + 1$ e ir a Paso 9
Paso 11	:	Actualizar $u_k = f(x_k)$
Paso 12	:	Obtener \bar{x}_{k+1} e y_{k+1} mediante Euler o Runge Kutta
Paso 13	:	Actualizar $\bar{H}(k+1, 1:(n_2-2)) = \bar{x}_k, \bar{H}(k+1, (n_2-1)) = y_k, \bar{H}(k+1, n_2) = u_k$
Paso 14	:	Si $k = N_s$ Fin. En otro caso ir a paso 3
Salida	:	\bar{H}_{out}

donde:

- \bar{x} es el vector de estados.
- y es la salida del proceso.
- N_1 es la longitud del vector de estados.
- u es la señal de control.
- r_0 es el valor de consigna al inicio de la simulación.
- \bar{r} es el vector de referencias (valores de consigna) a lo largo de la simulación.
- \bar{t}_r es el vector de instantes en los que se aplica cada referencia (elementos de \bar{r}).
- N_2 es la longitud del vector de referencias a lo largo de la simulación.

- \bar{d} es el vector de perturbaciones a lo largo de la simulación.
- \bar{t}_d es el vector de instantes en los que se aplica cada perturbación (elementos de \bar{d}).
- N_3 es la longitud del vector de perturbaciones a lo largo de la simulación.
- Δt_k es el paso de integración.
- T_s es el tiempo de simulación.
- \bar{H} es la matriz de históricos.
- n_1 es el número de filas de la matriz de históricos y coincide con el número de pasos de la simulación.
- n_2 es el número de columnas de la matriz de históricos. Estas columnas representan los estados, la salida y la señal de control.
- La dimensión de la matriz de históricos puede ampliarse añadiendo más columnas con el objeto de registrar un mayor número de variables.

En caso de simular el sistema en lazo abierto sin controlador, el algoritmo utilizado es el mismo, sustituyendo el vector de referencias por otro vector que contenga los valores de la variable manipulada a lo largo de la simulación y eliminando el paso 11 (actualización de la señal de control), así como la variable u .

Por otro lado, también se incluye la simulación de los sensores-transmisores de las variables del proceso, que introducen una señal de ruido gaussiano con una desviación típica previamente establecida (que puede ser definida por el usuario) y un cálculo de la señal de control en magnitudes eléctricas. El algoritmo transforma la señal de los sensores en valores correspondientes a las magnitudes físicas para realizar el cálculo de la acción de control y, posteriormente, se transforma el resultado en una señal eléctrica que acciona a la válvula de control. En el menú de configuración de la aplicación, se dispone de una opción para modificar la magnitud de las señales eléctricas de los sensores y del controlador, pudiendo elegir entre:

- Señal de 0 a 10 V.
- Señal de 4 a 20 mA.

Para el caso del control PID, la señal eléctrica de la variable controlada medida es transformada a un valor en tanto por ciento, abarcando todo el rango de medición del sensor-transmisor. Esta transformación también se aplica sobre la referencia, con el objetivo de obtener el error de seguimiento y calcular la señal de control, lo que implica una ganancia adimensional para el controlador.

Para simular tiempos muertos aparentes, se ha diseñado una función que retrasa los datos de caudal de refrigeración que se introducen en el integrador numérico. Esta misma función se utiliza para retrasar los datos del estado interno calculado y, de esta manera, obtener diferentes tiempos muertos para cada relación estado/entrada. De manera análoga, también se retrasan los datos de temperatura de refrigeración a la entrada, simulando un tiempo muerto en la dinámica de perturbación. De este modo, se añaden retardos reales que tiene el proceso.

Por otro lado, la matriz de históricos se ha ampliado para almacenar las siguientes variables:

- Estado interno del proceso (temperatura del reactor, concentración de reactivo a la salida y temperatura de refrigeración) obtenidos mediante el integrador numérico.
- Lectura de los sensores que miden las variables del estado interno del proceso.
- Datos procesados de los sensores que miden las variables del estado interno del proceso.
- Caudal de refrigeración.

- Abertura de la válvula del caudal de refrigeración.
- Temperatura de refrigeración a la entrada.
- Lectura del sensor que mide la temperatura de refrigeración a la entrada.
- Datos procesados del sensor que mide la temperatura de refrigeración a la entrada.
- Temperatura del caudal de alimentación del reactor (en el caso del control en cascada).
- Lectura del sensor que mide la temperatura del caudal de alimentación del reactor (en el caso del control en cascada).
- Datos procesados del sensor que mide la temperatura del caudal de alimentación del reactor (en el caso del control en cascada).
- Calor extraído del reactor por el sistema de refrigeración.
- Constante cinética.
- Tiempo transcurrido.
- Error de seguimiento.
- Integral del error de seguimiento.
- Valor de referencia.
- Salida del controlador.
- Referencia proporcionada por el controlador maestro, en el caso del control en cascada.

La variable manipulable se hace pasar por una función de saturación, que mantiene los valores dentro de unos límites físicos establecidos (de 0 a 100 % de apertura de la válvula de refrigeración). Lo mismo ocurre para la salida del controlador, manteniéndose siempre en el rango correspondiente, según la opción seleccionada por el usuario. Asimismo, también se incluye un algoritmo “*anti-windup*” simple en el caso del control PID, que se detalla con más profundidad en el capítulo 5.

3.2 Simulación dinámica

Con el objetivo de que el usuario pueda observar de forma progresiva cómo evoluciona el estado interno del reactor, se ha implementado en la interfaz gráfica un algoritmo que permite ejecutar los ciclos del bucle de simulación en intervalos de tiempo definidos, a un ritmo previamente fijado por el usuario. Con ello, se puede acelerar el tiempo de simulación con respecto al tiempo real de proceso.

A diferencia de la simulación estática, este algoritmo tiene asociado un temporizador que gestiona los ciclos de ejecución del bucle descrito en la Tabla 4 (Algoritmo 1: Simulación del sistema controlado) y se ejecuta de forma indefinida en el tiempo, hasta que el usuario indique su detención, pudiendo pararlo temporal o permanentemente. De esta manera, es posible simular más de un cambio en la entrada para la misma simulación, pausándola, introduciendo nuevos valores en los campos editables y retomándola desde el instante en el que se detuvo. No obstante, este modo de simulación es más lento, y no se permite indicar con un valor numérico el instante en el que se pretenden aplicar los cambios en las condiciones de entrada, si bien esto es lo que ocurre en la práctica, por lo que dota a la aplicación de un mayor realismo.

Para controlar la ejecución de los ciclos, se ha hecho uso de la clase “*timer*” de Matlab [1], creando un objeto con las siguientes propiedades:

- Modo de ejecución de velocidad fija: la función “*callback*” asociada al temporizador se ejecuta periódicamente. La primera vez se ejecuta inmediatamente después de añadirse a la cola de ejecución.
- Periodo de ejecución definido por el usuario en el intervalo $[0,01 - 1,2]$ s.
- Retraso en la primera ejecución de 0 s.
- Tareas a ejecutar infinitas: La función “*callback*” se ejecutará un número indefinido de veces hasta que sea interrumpida por la función “*stop*”.
- La función “*callback*” asociada depende del algoritmo de control a utilizar.
- Modo “*Busy*” “*Drop*”: Si debe ejecutarse la función “*callback*” y la cola de ejecución no está vacía, se omite la tarea.

El objeto “*timer*” se crea al inicializar la interfaz gráfica, definiendo sus propiedades anteriormente indicadas y asignando un tiempo de separación entre ejecuciones por defecto de 0,5 segundos. Cuando el usuario selecciona la opción “Simulación dinámica” y comienza la simulación, dependiendo de la configuración aportada, la interfaz seleccionará automáticamente la función “*callback*” que ejecutará el temporizador y guardará los estados iniciales dentro de una estructura, y_0 , y los parámetros necesarios para el control automático dentro de otra estructura, P . Los argumentos de entrada del “*callback*” se tratan de y_0 , de P , del valor de consigna, del valor de perturbación, de los objetos de línea animada (gráficas de representación de datos) y de los objetos de campo numérico de la interfaz, así como de otros parámetros constantes que contienen información necesaria para la simulación.

Una vez realizadas las preparaciones, se inicia el temporizador. En cada ciclo, se ejecuta la función de “*callback*”, la cual calcula los valores de los estados en el instante siguiente, añadiéndolos a las gráficas. Los parámetros asociados a la función y guardados en P , así como el vector de estados, son actualizados al final de cada ciclo, de forma que los argumentos de entrada en la nueva llamada a la función se corresponden con los datos de salida de la anterior.

El funcionamiento del algoritmo de simulación se representa de forma esquematizada en el siguiente diagrama de flujo (Figura 4), donde:

- “*TimerFcn*” es la propiedad del objeto “*timer*” donde se almacena en un vector de celda la referencia a la función “*callback*” y sus argumentos de entrada.
- “*Callback*” es la función ejecutada por el temporizador en cada ciclo.
- “*ref*” es el valor de consigna introducido por el usuario.
- “*dist*” es el valor de perturbación introducido por el usuario.
- “*Axes*” hace referencia a los objetos gráficos donde se representa temporalmente la evolución de los estados en la interfaz gráfica.
- “*x_max*” es el valor del horizonte de visualización de datos en las gráficas.
- “*Out*” hace referencia a los campos numéricos de la ventana principal, que son actualizados en cada ciclo.
- “*t*” es el tiempo transcurrido en minutos.

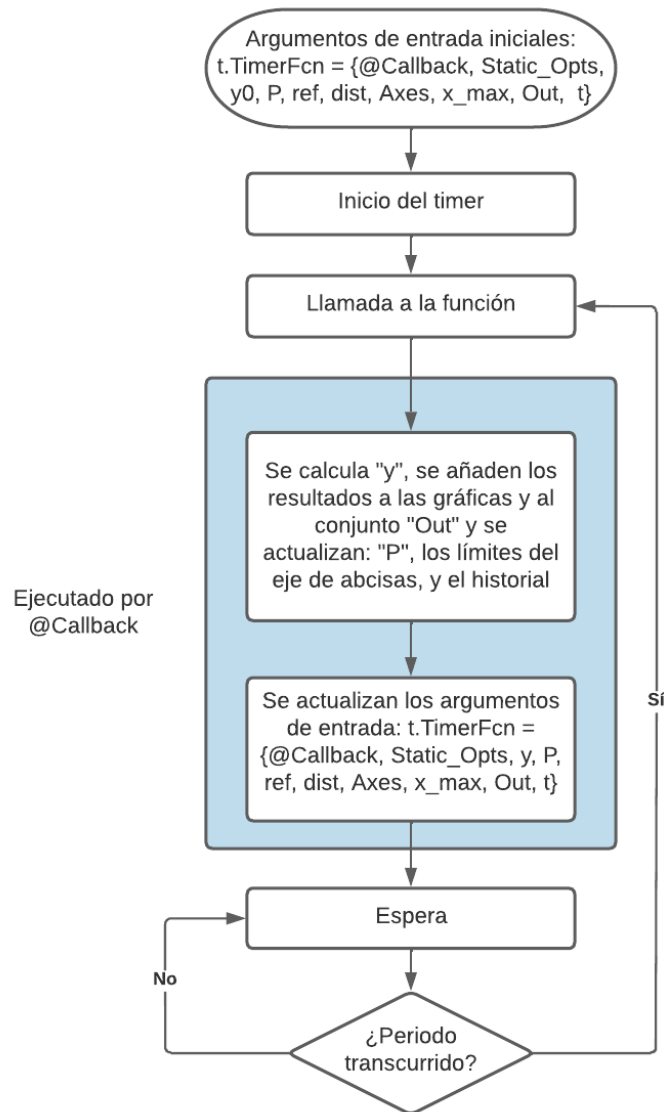


Figura 4. Diagrama de flujo del algoritmo de simulación dinámica en Matlab

Debido a que el “*function handle*” asociado a la función “*Callback*” es una propiedad del objeto “*timer*”, es posible modificarla en función del algoritmo seleccionado por el usuario. De esta forma, al comenzar la simulación dinámica, se evalúa en la configuración cuáles son los datos actuales de control, lo que permite seleccionar el “*Callback*” a ejecutar por el temporizador y establecer los valores iniciales de los parámetros de control (que son guardados dentro de la estructura *P*).

Por otro lado, también se ha hecho uso de la propiedad del temporizador “*UserData*”, que consiste en un campo genérico mediante el cual pueden añadirse nuevos datos al objeto, resultando especialmente útil para modificar el comportamiento del algoritmo que ejecuta el “*Callback*” de manera externa (por ejemplo, haciendo click sobre un determinado botón). De esta forma se han implementado las siguientes características:

- Guardar los valores de las variables de estado al inicio de la simulación (condiciones iniciales).
- Habilitar el modo de autoajuste de ejes, en caso de ser activado desde la ventana principal.
- Guardar un historial de simulación, que contiene los valores calculados de las variables del proceso y cuya longitud es configurable por el usuario.
- Realizar el guardado de datos de simulación, en caso de activar esta funcionalidad desde la ventana principal.
- Detener el guardado de datos de simulación.
- Almacenar un historial de las variables del proceso obtenidas.

Para el guardado de datos, se tienen tres entradas: Una consiste en un indicador que se activa en caso de que el usuario seleccione la opción correspondiente al guardado de los datos. Las otras dos se corresponden con el número máximo de datos que se pueden guardar y con la orden de detención del proceso de grabación.

Las salidas corresponden a las variables del proceso que son guardadas en cada ciclo de ejecución de la función “*Callback*” y un indicador que se activa en caso de haber finalizado con éxito la operación y, mediante el cual, la aplicación reconoce si existen o no datos guardados. Estas salidas se comunican como argumento de entrada a la ventana gráfica destinada a la identificación del proceso, que se detalla en profundidad en el siguiente capítulo.

Con respecto al almacenamiento del historial, los datos comienzan a registrarse desde el inicio de la simulación y, una vez alcanzado el número máximo de datos establecido, se elimina el primer valor registrado de cada variable y se almacena el siguiente, manteniendo constante, a partir de este momento, el tamaño de la base de datos.

4 IDENTIFICACIÓN DEL PROCESO

Debido a que el sistema es no lineal, se ha incluido en la interfaz gráfica una herramienta que permite obtener modelos lineales aproximados del sistema, realizando previamente una simulación de ensayo con entrada en escalón.

Estos modelos lineales son de gran utilidad para obtener un ajuste inicial del controlador en el punto de operación deseado, utilizando reglas de sintonización a partir de los parámetros identificados. Por otro lado, también resulta útil para el diseño de controladores más avanzados, como es el caso del control predictivo, en el que es necesario un modelo de predicción, o del control adaptativo “Gain Scheduling”, en el que resulta conveniente conocer distintos modelos lineales que se ajusten adecuadamente a cada punto de operación del proceso.

En este capítulo, se sientan las bases de la identificación paramétrica de sistemas dinámicos en tiempo discreto a partir de datos obtenidos mediante ensayos experimentales. Luego, se realiza una descripción sobre la herramienta de identificación implementada en la interfaz gráfica, así como los métodos empleados para la detección y prevención de errores en el proceso de estimación.

4.1 Identificación paramétrica de sistemas lineales aproximados

La identificación paramétrica abarca un amplio conjunto de métodos utilizados para la estimación de modelos matemáticos de procesos físicos a partir de datos obtenidos experimentalmente, ya sea de ensayos reales realizados en planta o a través de simulación. A diferencia de los métodos de identificación no paramétricos, este proceso da como resultado, de forma directa, los valores de los parámetros de la estructura matemática seleccionada para tal fin.

Generalmente, estos parámetros se obtienen a partir de la definición y resolución de un problema de optimización sujeto a un determinado criterio, por lo que es este carácter óptimo lo que permite la obtención de modelos más fidedignos que los proporcionados por los clásicos métodos no paramétricos.

El proceso general de identificación puede resumirse en los siguientes apartados:

1. Realización del ensayo en planta o ejecución de la simulación y registro de datos.
2. Preprocesamiento de los datos.
3. Selección del modelo matemático a identificar.
4. Selección del método de identificación.
5. Obtención de los parámetros óptimos, de acuerdo al criterio de optimización.
6. Validación del modelo.

Este ensayo experimental suele consistir en la excitación de la planta a partir de cambios de tipo escalón en la entrada del sistema, de manera que, partiendo de un punto de operación nominal, se obtenga información que permita captar la dinámica del proceso. El muestreo de cada tipo de dato necesario se realiza de forma simultánea y sincronizada, permitiendo, posteriormente, relacionar en el tiempo un dato de entrada con el correspondiente valor de salida.

El modelo obtenido mediante el proceso es normalmente lineal en tiempo discreto, de forma que se simplifica el diseño de controladores en el entorno cercano al punto de operación alrededor del cual se ha llevado a cabo el experimento.

Los tipos de modelos dinámicos más utilizado para la identificación paramétrica se detallan a continuación [6]:

- Modelo autorregresivo y exógeno (ARX):

$$y(t) = a_1y(t-1) + a_2y(t-2) + \dots + a_{n_a}y(t-n_a) + b_1u(t-1) + b_2u(t-2) + \dots + b_{n_b}u(t-n_b) + \xi(t) \quad (53)$$

donde $\xi(\cdot)$ representa un ruido blanco con media cero y varianza σ^2 . Se observa que la parte autorregresiva está formada por la secuencia de términos de salida en el pasado y la parte exógena por los términos de la secuencia de entrada. n_a y n_b son, respectivamente, el orden de la parte autorregresiva y de la parte exógena [6]. Nótese que las variables de entrada y salida son incrementales respecto a una posición de equilibrio.

La función de transferencia resultante, en el dominio de Z es la siguiente:

$$y(z) = \frac{b_1z^{-1} + b_2z^{-2} + \dots + b_{n_b}z^{-n_b}}{1 + a_1z^{-1} + a_2z^{-2} + \dots + a_{n_a}z^{-n_a}}u(z) + \frac{1}{1 + a_1z^{-1} + a_2z^{-2} + \dots + a_{n_a}z^{-n_a}}\xi(z) \quad (54)$$

$$y(z) = \frac{B(z)}{A(z)}u(z) + \frac{1}{A(z)}\xi(z) \quad (55)$$

Puede comprobarse que el caso ARX equivale a una dinámica lineal a la que se le añade un ruido blanco filtrado por su propia ecuación característica (Figura 5).

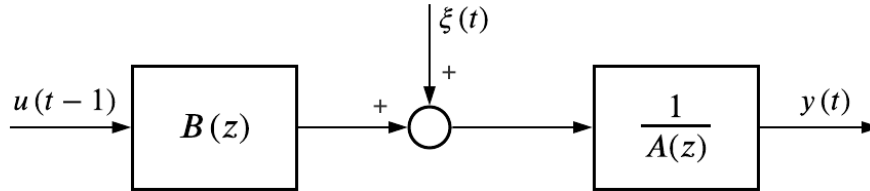


Figura 5. Diagrama de bloques de un modelo con estructura ARX

Para la identificación, los parámetros de ajuste son:

$$\theta = [a_1, a_2, \dots, a_{n_a}, b_1, b_2, \dots, b_{n_b}] \quad (56)$$

- Modelo autorregresivo y exógeno de media móvil (ARMAX):

$$y(t) = a_1y(t-1) + a_2y(t-2) + \dots + a_{n_a}y(t-n_a) + b_1u(t-1) + b_2u(t-2) + \dots + b_{n_b}u(t-n_b) + v(t) \quad (57)$$

Siendo $v(t)$ el término de media móvil asociado al ruido:

$$v(t) = \xi(t) + c_1\xi(t-1) + \dots + c_{n_c}\xi(t-n_c) \quad (58)$$

donde n_c es el orden de la parte de media móvil [6]. Se observa que, a diferencia del caso ARX, se consideran los valores de la señal de ruido blanco en el pasado, por lo que se tiene un modelo con un mayor número de parámetros, dificultando el proceso de identificación, pero obteniendo, generalmente, mejores resultados.

La función de transferencia resultante, en el dominio de Z es la siguiente:

$$y(z) = \frac{b_1 z^{-1} + b_2 z^{-2} + \dots + b_{n_b} z^{-n_b}}{1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_{n_a} z^{-n_a}} u(z) + \frac{1 + c_1 z^{-1} + c_2 z^{-2} + \dots + c_{n_c} z^{-n_c}}{1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_{n_a} z^{-n_a}} \xi(z) \quad (59)$$

$$y(z) = \frac{B(z)}{A(z)} u(z) + \frac{C(z)}{A(z)} \xi(z) \quad (60)$$

A continuación, se muestra el diagrama de bloques de la estructura ARMAX (Figura 6):

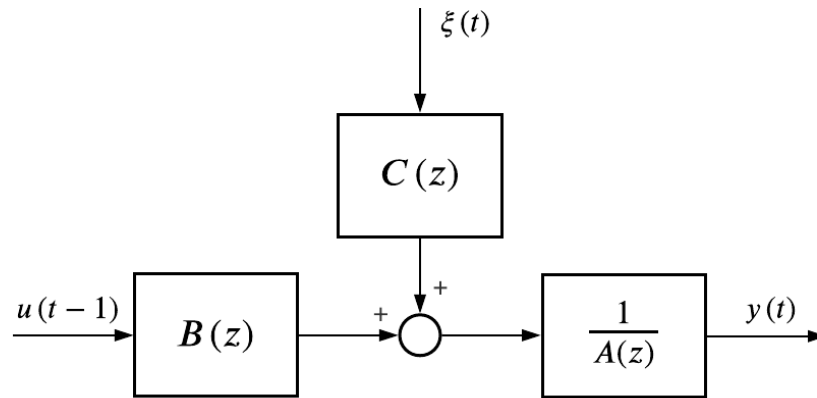


Figura 6. Diagrama de bloques de un modelo con estructura ARMAX

Para la identificación, los parámetros de ajuste son:

$$\theta = [a_1, a_2, \dots, a_{n_a}, b_1, b_2, \dots, b_{n_b}, c_1, c_2, \dots, c_{n_c}] \quad (61)$$

- Modelo Box y Jenkins:

Este tipo de modelo se trata de un caso general, que engloba a otros tipos de modelos, cuya expresión en el dominio de Z viene dado por:

$$y(z) = \frac{B(z)}{A(z)} u(z) + \frac{C(z)}{D(z)} \xi(z) \quad (62)$$

A continuación, se muestra de forma esquematizada un modelo Box y Jenkins (Figura 7):

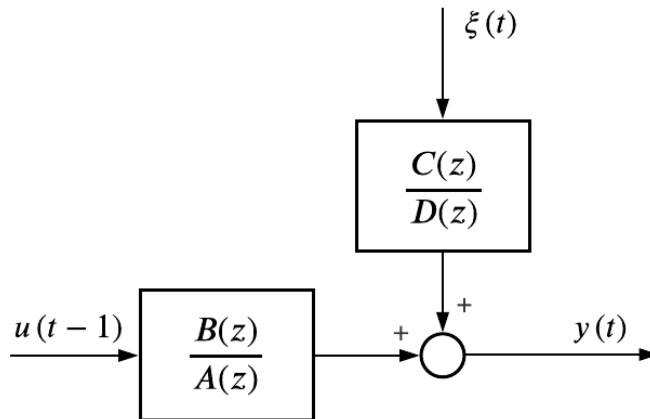


Figura 7. Diagrama de bloques de un modelo con estructura Box and Jenkins

Para la identificación, los parámetros de ajuste son:

$$\theta = [a_1, a_2, \dots, a_{n_a}, b_1, b_2, \dots, b_{n_b}, c_1, c_2, \dots, c_{n_c}, d_1, d_2, \dots, d_{n_d}] \quad (63)$$

Nótese que los órdenes de la parte autorregresiva y la parte exógena pueden seleccionarse para obtener un orden determinado en la función de transferencia de la dinámica identificada. En el control automático, suelen utilizarse para el diseño de controladores sistemas de primer y segundo orden:

- Sistemas de primer orden con tiempo muerto

En tiempo continuo y en el dominio de \mathcal{L} , la expresión general de un sistema de primer orden con tiempo muerto viene dada por la siguiente función de transferencia:

$$\frac{y(s)}{u(s)} = \frac{K_p}{\tau_p s + 1} e^{-\tau_{mp}s} \quad (64)$$

donde K_p es la ganancia estática, τ_p es la constante de tiempo del sistema y τ_{mp} el tiempo muerto.

Empleando un retenedor de orden cero (ZOH), la equivalencia en tiempo discreto puede expresarse mediante la siguiente relación de recurrencia:

$$y(k) = -a \cdot y(k-1) + b \cdot u(k-1-n_{del}) \quad (65)$$

donde n_{del} es el número de pasos que tarda un cambio en la entrada en afectar a la señal de salida.

- Sistemas de segundo orden con tiempo muerto

La función de transferencia en tiempo continuo que describe la dinámica de un sistema de segundo orden con tiempo muerto puede expresarse mediante la siguiente ecuación:

$$\frac{y(s)}{u(s)} = \frac{K_p \omega_n^2}{s^2 + 2\zeta \omega_n s + \omega_n^2} e^{-\tau_{mp}s} \quad (66)$$

donde ω_n es la frecuencia natural del proceso y ζ la constante de amortiguamiento.

Empleando un retenedor de orden cero (ZOH), la equivalencia en tiempo discreto puede expresarse mediante la siguiente relación de recurrencia:

$$y(k) = -a_1 \cdot y(k-1) - a_2 \cdot y(k-2) + b_1 \cdot u(k - n_{del} - 1) + b_2 \cdot u(k - n_{del} - 2) \quad (67)$$

Asimismo, para cualquier estructura matemática descrita en este apartado, los datos de la parte exógena pueden retrasarse n_{del} unidades para incluir en la identificación el efecto de tiempo muerto.

4.2 Método de identificación paramétrica por mínimos cuadrados

El método de identificación implementado en la interfaz gráfica es el de mínimos cuadrados, cuyo procedimiento se detalla a continuación.

Dado un conjunto de datos de entrada y salida obtenidos por simulación o mediante ensayos reales, se define como error de modelado:

$$\bar{E} = \bar{Y} - \bar{X}\bar{\theta} \quad (68)$$

donde \bar{X} es la matriz que contiene los datos de entrada, \bar{Y} los de salida y $\bar{\theta}$ es el vector paramétrico. El producto $\bar{X}\bar{\theta}$ representa, por tanto, la estimación de la salida obtenida a partir de los datos de entrada y de los parámetros a estimar. Nótese que el conocimiento de los valores pasados de las entradas y salidas no proporcionan ninguna información útil para evaluar $\xi(t)$ y, por ello, esta no se incluye en la expresión del estimador [6].

Se debe minimizar la suma de los errores al cuadrado, por lo que puede definirse el siguiente problema de optimización:

$$\min_E \bar{E}^T \bar{E} = \min_{\theta} (\bar{Y} - \bar{X}\bar{\theta})^T (\bar{Y} - \bar{X}\bar{\theta}) \quad (69)$$

Este problema tiene solución explícita dada por:

$$\bar{\theta} = (\bar{X}^T \bar{X})^{-1} \bar{X}^T \bar{Y} \quad (70)$$

Debido a que los reactores descritos en el apartado 2 se tratan de modelos de reducida complejidad, se ha optado por la selección de una estructura ARX para el algoritmo de identificación implementado en la GUI, con órdenes $n_a = 1$ y $n_b = 1$, que equivale a un sistema de primer orden (ecuación (65)). Nótese que un valor de $n_a = 1$ implica que la salida en el instante anterior, y_{k-1} , tiene repercusión en la salida actual del proceso, y_k . Por otro lado, también puede comprobarse que la estructura ARX definida mediante la ecuación (53), incluye el retraso debido al tiempo de muestreo, al no considerarse el efecto de u_k .

Con el objetivo de tener en cuenta el tiempo muerto en la estimación por mínimos cuadrados, se ha considerarlo dicho parámetro como una información conocida a priori, estableciendo un retraso de n_{del} unidades de Δt_k en los datos exógenos.

El vector paramétrico, en este caso, presenta la siguiente estructura:

$$\bar{\theta} = \begin{bmatrix} a \\ b \end{bmatrix} \quad (71)$$

donde a y b son los parámetros del modelo lineal de primer orden (ecuación (65)). Por lo que debe seleccionarse una base de datos de entrada que contenga los valores de $y(k-1)$ y $u(k-1-n_{del})$ en cada instante de tiempo y una base de datos de salida con los valores de $y(k)$. Las filas de las matrices \bar{X} e \bar{Y} son, por tanto:

$$\bar{X}_{k,*} = [-y_{k-1} \quad u_{k-1-n_{del}}] \quad (72)$$

$$\bar{Y}_{k,*} = y_k \quad (73)$$

donde \bar{u} e \bar{y} son, respectivamente, los vectores de datos de entrada y de salida incrementales respecto al punto de equilibrio inicial.

En la GUI, una vez obtenidos los datos experimentales, se aplica un filtro de media móvil sobre la base de datos, con el objetivo de reducir el impacto del ruido de medida sobre la identificación. Los valores de salida del filtro se utilizan como una nueva base de datos para el algoritmo de identificación paramétrica. En la Figura 8 puede observarse el suavizado obtenido tras aplicar el filtro a los datos medidos en un ensayo con entrada en escalón.

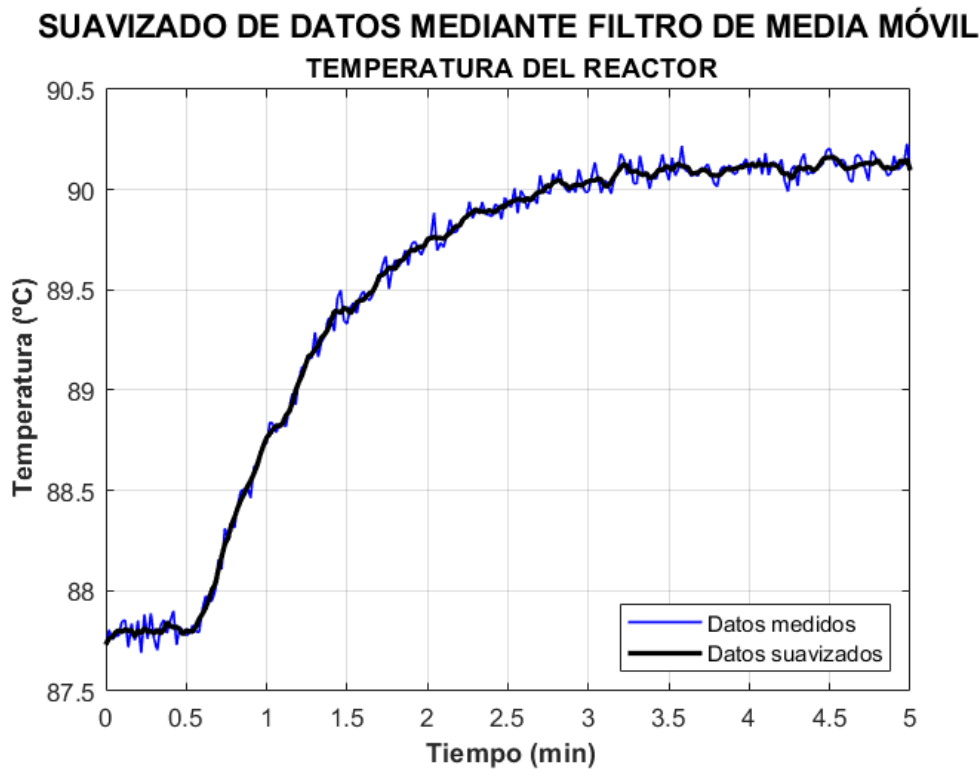


Figura 8. Suavizado de datos mediante filtro de media móvil

Una vez realizado el proceso de identificación desde la aplicación, se añade un valor aleatorio, dentro de un rango estrecho de valores, a cada parámetro obtenido, incorporando así el efecto potencial de las incertidumbres presentes en el experimento con el proceso real. Los números aleatorios son generados a partir de una distribución normal de media cero y con una desviación típica que depende del tipo de modelo seleccionado, del tipo de entrada (variable manipulada o perturbación) y del parámetro en cuestión.

En la Figura 9, se muestra una comparación entre la respuesta del proceso modelado y la del sistema lineal identificado para las mismas condiciones de entrada a la simulación. El experimento se ha realizado variando el valor de apertura de la válvula de control un 5 % respecto a su posición de equilibrio cada 10 minutos. Puede comprobarse que existen pequeñas discrepancias entre las respuestas debido a las no linealidades del proceso.

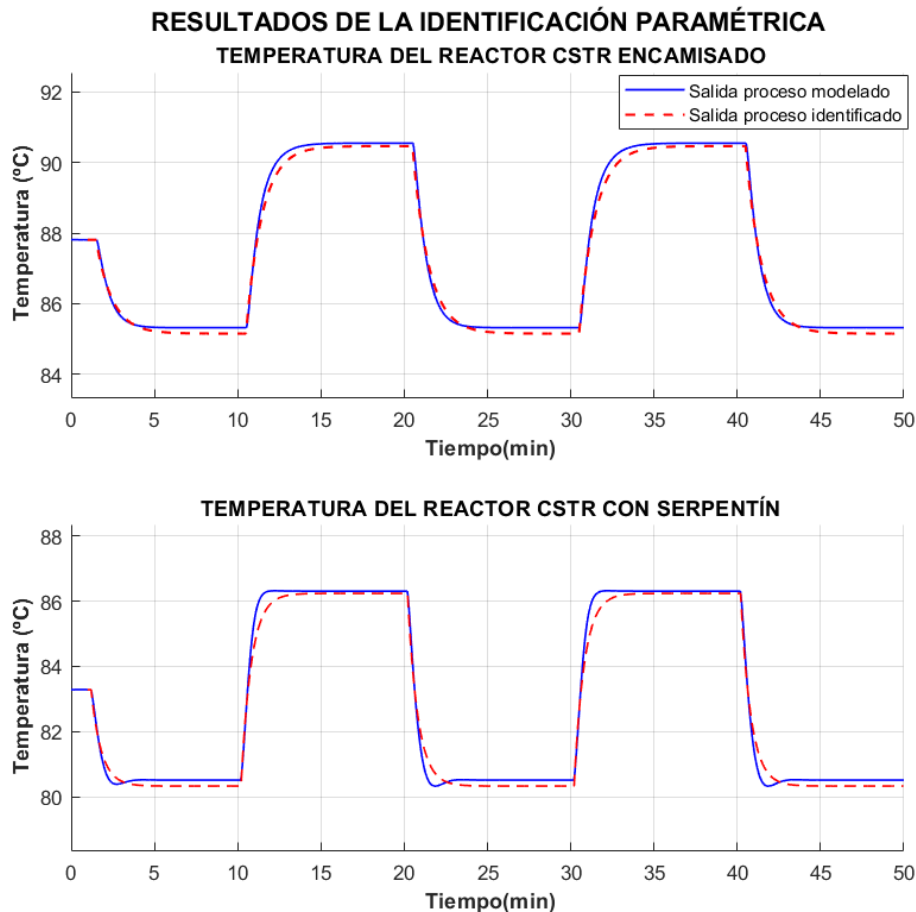


Figura 9. Comparación entre proceso modelado y proceso identificado empíricamente para distintos modelos de reactor CSTR

En el Anexo B se detallan los datos de simulación, así como los resultados numéricos obtenidos en este experimento.

4.3 Transformación del modelo de tiempo discreto a tiempo continuo

El método de identificación paramétrica descrito en el apartado anterior da como resultado un modelo lineal en tiempo discreto. Sin embargo, es importante destacar que la mayoría de métodos analíticos clásicos para el diseño de controladores, requieren de una expresión en tiempo continuo. Por ello, el algoritmo implementado realiza una transformación del sistema identificado para obtener una función de transferencia continua.

Esta transformación puede llevarse a cabo mediante varios métodos. Los más utilizados son:

- Retenedor de orden cero (“*Zero-Order Hold*”): Se asume que la entrada es constante entre periodos de muestreo.
- Retenedor de primer orden (“*First-Order Hold*”): Se asume que la entrada es lineal creciente o decreciente entre periodos de muestreo.
- Transformación bilineal o método de Tustin: Aproximación de la derivada.

El mantenedor de orden cero resulta muy útil para obtener una buena aproximación de la respuesta temporal para entradas en escalón, por lo que se ha optado por implementar este método en la interfaz gráfica. La transformación de tiempo continuo a discreto [7] viene dado por:

$$G(z) = (1 - z^{-1})\mathcal{Z}\left\{\frac{G(s)}{s}\right\} \quad (74)$$

donde para un sistema de primer orden, se tiene:

$$\mathcal{Z}\left\{\frac{k\frac{p}{s+p}}{s}\right\} = k\frac{(1 - e^{-p\Delta t_k})z}{(z-1)(z - e^{-p\Delta t_k})} \quad (75)$$

$$G(z) = k\frac{1 - e^{-p\Delta t_k}}{z - e^{-p\Delta t_k}} = k\frac{(1 - e^{-p\Delta t_k})z^{-1}}{1 - e^{-p\Delta t_k} \cdot z^{-1}} \quad (76)$$

Los modelos identificados por la aplicación vienen dados por la siguiente expresión:

$$G(z) = \frac{bz^{-1}}{1 + az^{-1}} \quad (77)$$

Con lo que puede obtenerse la correspondencia con la **constante de tiempo** de la siguiente forma:

$$\tau = \frac{1}{p} = \frac{\Delta t_k}{|-\ln(-a)|} \quad (78)$$

donde el parámetro a se obtiene de la identificación paramétrica, y tiene valor negativo si se considera una base de datos acorde a la estructura representada en las ecuaciones (72) y (73). La **ganancia estática** puede obtenerse aplicando el teorema del valor final para sistemas en tiempo discreto:

$$y(\infty) = \lim_{k \rightarrow \infty} [y(k)] = \lim_{z \rightarrow 1} [(1 - z^{-1})y(z)] \quad (79)$$

Aplicando un escalón unitario se tiene:

$$y(\infty) = \lim_{z \rightarrow 1} \left[\frac{bz^{-1}}{1 + az^{-1}} \right] = \frac{b}{1 + a} \quad (80)$$

donde b es el otro elemento del vector paramétrico obtenido en la identificación. El **tiempo muerto** debe hacerse igual al tiempo de retraso:

$$\tau_{mp} = n_{del} \cdot \Delta t_k \quad (81)$$

Nótese que el valor de n_{del} se considera como información conocida y pertenece al conjunto de argumentos de entrada del algoritmo. Esto implica que el tiempo muerto y la constante de tiempo no guardan ninguna relación desde el punto de vista de la identificación.

Finalmente, se obtiene la siguiente expresión:

$$G(s) = \frac{\frac{b}{1+a}}{\frac{\Delta t_k}{|-\ln(-a)|}s + 1} e^{-\tau_{mp}s} \quad (82)$$

4.4 Herramienta de ensayos con entrada en escalón para identificación de modelos aproximados

La interfaz gráfica dispone de una ventana dedicada a la identificación de modelos lineales a partir de ensayos con entrada en escalón, de forma que el usuario puede introducir los valores de abertura de la válvula (variable manipulada) y los instantes en los que éstos se van a aplicar para, posteriormente, simular el proceso y realizar una identificación paramétrica por mínimos cuadrados con los resultados obtenidos. En la Figura 10 se ilustra con un diagrama de bloques el sistema lineal completo:

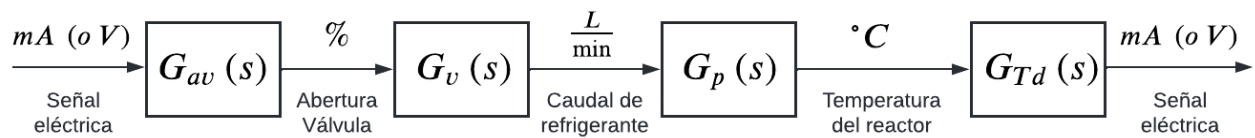


Figura 10. Diagrama de bloques del sistema lineal completo

En el proceso real, el ensayo con entrada en escalón consiste en cambiar la variable manipulada estando el controlador en manual. Este dispositivo dispondría de un botón con una escala porcentual, correspondiente al % de la abertura deseada para la válvula de control. Esta señal porcentual será convertida internamente a una señal eléctrica digital en tensión, y llegará al actuador eléctrico del elemento final de control en forma de señal eléctrica continua (mA , $o V$) mediante una convertidor digital-analógico asociado al propio controlador. El actuador de la válvula convertirá la señal de mA ($o V$) a %, y luego, esa señal porcentual vinculada a la posición del vástago de la válvula hará que cambie la posición de su obturador (es el mismo cambio por ser solidarios vástago y obturador), lo cual cambiará el caudal de fluido refrigerante, siguiendo una dinámica en la que la ganancia estática será la sensibilidad de la válvula ($l/min / \%$). A continuación, una vez cambia el caudal, cambiará el valor de la variable de interés en el proceso propiamente dicho ($^{\circ}C$), con una dinámica que será la más lenta de todas las dinámicas implicadas en el ensayo. El valor de la variable controlada solo puede conocerse midiéndola, es decir, mediante un sensor-transmisor, cuya función de transferencia es la sensibilidad de este (mA $o V/^{\circ}C$), pues se ha asumido que su dinámica es despreciable. La señal de mA ($o V$) llegará al controlador, que la transformará en una señal digital de tensión mediante un convertidor analógico-digital ($V/(mA$ $o V)$), que luego podría visualizarse en el *display* del propio controlador como una señal porcentual ($\%/V$) o ingenieril ($^{\circ}C/V$).

La Figura 10 recoge dos bloques asociados a la válvula, que son G_{av} y G_v , correspondientes al actuador y al cuerpo de la válvula de control. El sensor-transmisor es necesario, ya que es la forma de saber cuál es el valor de la variable controlada en un proceso real que tiene lugar en una planta, y forma parte de la identificación. La cadena completa válvula de control + proceso propiamente dicho + sensor-transmisor conforma la dinámica global con la que deberá sintonizarse el controlador, cuya ganancia será adimensional.

En la identificación implementada en la GUI se registran los datos del ensayo mediante el sensor-transmisor que transmite la señal en una magnitud eléctrica, posteriormente transformada a $^{\circ}C$ para la salida del proceso. Por otro lado, se registran los datos de abertura de válvula introducidos por el usuario que, en el proceso real, se obtendrían mediante el propio controlador configurado en manual, que puede tener una salida a un dispositivo de adquisición de datos, el cual también registraría la señal de temperatura. Con estos valores se lleva a cabo el proceso de identificación, dando lugar a una función de transferencia con unidades de ganancia de [$^{\circ}C/\%$], lo que permite desacoplar la identificación de las variables eléctricas. Tomando como base el diagrama de la Figura 10, se incorporan dos bloques conversores, f_{st} y f_c , de modo que hay conversiones adicionales que afectan a las señales eléctricas: a $^{\circ}C$ para la salida del proceso y de % para la abertura de la válvula. Esto implica usar convertidores analógico-digital + digital a magnitud física (en $^{\circ}C$) (f_{st} global dado en $^{\circ}C/(V$ $o mA)$) y porcentual a digital + digital-analógico (f_c global dado en $(V$ $o mA)/\%$), respectivamente. Por tanto, el producto de las funciones de transferencia es el siguiente:

$$f_c \cdot G_{av} \cdot G_v \cdot G_p \cdot G_{TD} \cdot f_{st}$$

Con ganancias cuyas unidades son:

$$\left[\frac{(V \text{ o } mA)}{\%} \right] \cdot \left[\frac{\%}{(V \text{ o } mA)} \right] \cdot \left[\frac{l/min}{\%} \right] \cdot \left[\frac{^{\circ}C}{l/min} \right] \cdot \left[\frac{(V \text{ o } mA)}{^{\circ}C} \right] \cdot \left[\frac{^{\circ}C}{(V \text{ o } mA)} \right] \rightarrow \left[\frac{^{\circ}C}{\%} \right]$$

Nótese que esta relación representa el conjunto de dinámicas y conversiones que tendrían lugar en el registro de datos del ensayo en la realidad, si bien, en lo relativo a la identificación paramétrica, tan sólo es necesario disponer de los datos procesados de las señales de la variable controlada y la variable manipulada para formar la base de datos y obtener la función de transferencia deseada.

El factor f_{st} equivale al procesamiento de los datos del sensor, y el factor f_c incluye el procesamiento de la abertura de la válvula de control desde el propio controlador puesto en manual durante el ensayo. De este modo, se obtendrían directamente las mediciones en $^{\circ}C$ y $\%$.

Así pues, según el procedimiento descrito, el sistema identificado corresponde a $G_p(s) \cdot G_v(s)$, que en el proceso real es no lineal y, cuya ganancia estática será, en este caso, negativa. Los tiempos de respuesta del transmisor y del actuador de la válvula (en este caso, considerados nulos) se incluyen en la función de transferencia identificada, pues los datos procesados de temperatura se obtienen a partir de las mediciones del transmisor, y los valores de abertura de la válvula se obtiene directamente del controlador configurado en manual.

En la GUI, la variable manipulada varía en caso de tener configurado un control en cascada con el lazo interno cerrado. Además, el usuario dispone de dos botones para acceder a la herramienta: uno para cambios en la variable manipulada y otro para cambios en la variable de perturbación. De esta forma es posible identificar un gran número de dinámicas que tienen lugar en el proceso y obtener las funciones de transferencia correspondientes.

Una vez realizada la simulación, se permite al usuario almacenar los datos obtenidos en una hoja de cálculo de extensión “.xlsx” o en un archivo de texto de extensión “.txt”. Estos resultados pueden ser utilizados fuera de la interfaz, con la finalidad de obtener un modelo lineal mediante cualquier otro método de identificación conocido por el usuario.

Asimismo, el usuario puede llamar al algoritmo de identificación paramétrica mediante dos opciones:

- Identificación única: Se obtiene una única función de transferencia a partir de los datos del ensayo.
- Identificación por puntos: Se obtiene una función de transferencia por cada escalón que el usuario ha introducido en las condiciones de ensayo.

Para el proceso de identificación por puntos, se ha diseñado un algoritmo que agrupa, en conjuntos independientes, los datos de entrada y salida correspondientes a cada cambio en la variable manipulable. Posteriormente, estos datos son procesados por el algoritmo de identificación, el cual se ejecuta tantas veces como entradas se hayan definido en el experimento.

Independientemente de la opción seleccionada, tras haber realizado la identificación, la aplicación simulará el modelo lineal con los mismos datos de entrada del ensayo. Los resultados se muestran en las gráficas de forma superpuesta (Figura 11), reflejando las discrepancias existentes con los resultados de la simulación del modelo no lineal. Las funciones de transferencia obtenidas en este proceso estarán a disposición del usuario.

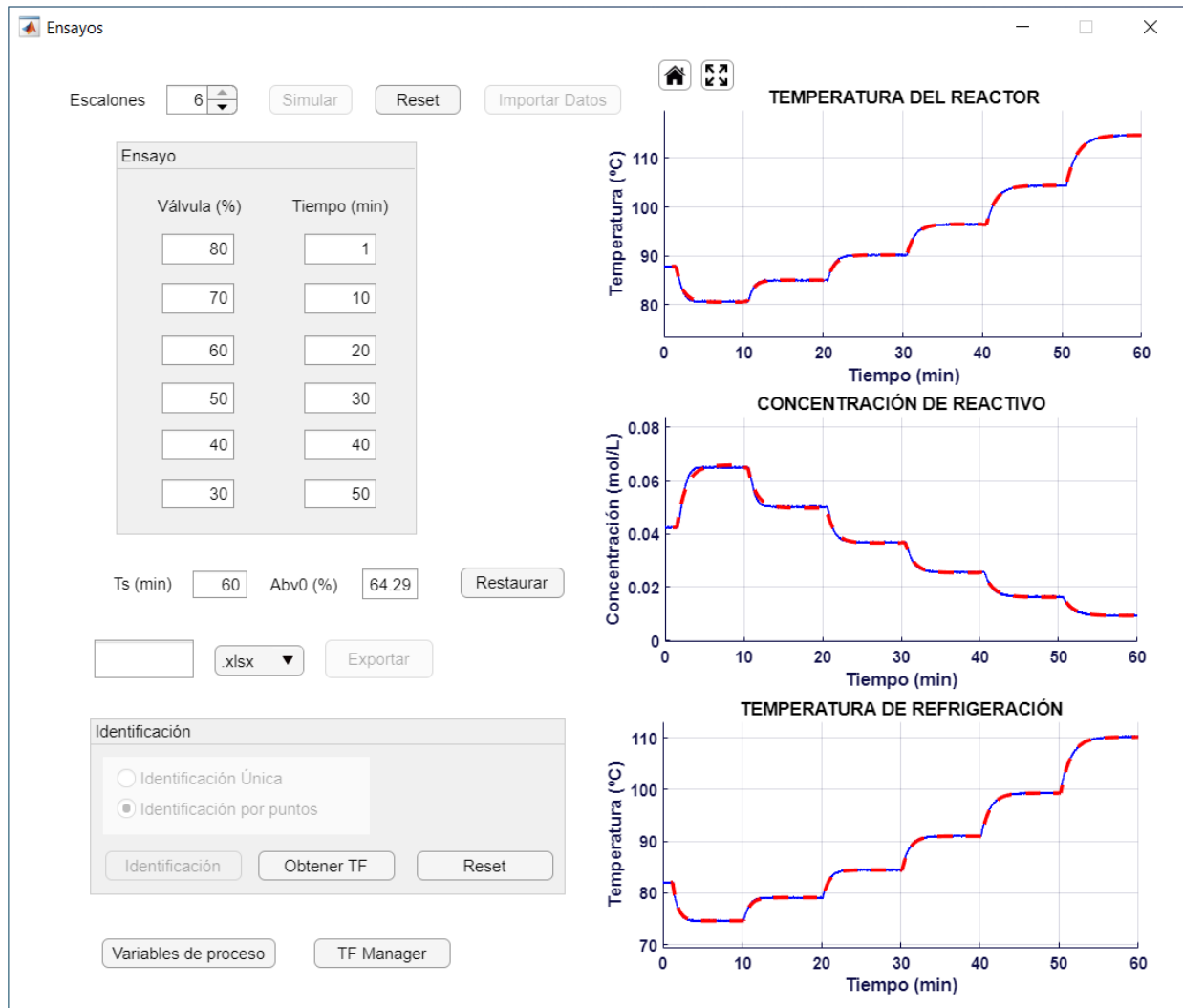


Figura 11. Realización en la GUI de una identificación paramétrica por cada punto de operación del ensayo

Esta representación gráfica de la respuesta temporal de los modelos lineales obtenidos se lleva a cabo mediante un algoritmo que ejecuta una simulación operando con variables incrementales y definiendo un vector temporal sincronizado con el ensayo realizado. Posteriormente, se suman los valores iniciales del estado interno del proceso para expresar los resultados en variables absolutas y se representan en las gráficas correspondientes con un trazado en rojo discontinuo.

En caso de haber realizado un guardado de datos con la opción de simulación dinámica desde la ventana principal, es posible utilizarlos en esta herramienta para llevar a cabo una identificación paramétrica. Asimismo, si existen archivos de simulación guardados en el equipo, también es posible importar los datos en la herramienta.

4.5 Detección y prevención de errores en la identificación

Con el fin de evitar posibles errores en la aplicación, se han implementado varias medidas mediante las cuales se impide llevar a cabo el proceso de identificación paramétrica, en caso de que los datos de entrada de simulación no sean adecuados. Esto puede ser debido a posibles incompatibilidades entre ellos o a que la información proporcionada por éstos no sea suficiente para realizar una identificación exitosa.

Las medidas adoptadas, se dividen en:

- Medidas que impiden la identificación prohibiendo o permitiendo la introducción de datos en los campos editables:
 - Límites inferior y superior en los campos editables, de forma que el usuario no puede introducir valores fuera del intervalo establecido.
 - Actualización del valor y de los límites de los campos editables de tiempo al modificar uno de ellos, de forma que se garantiza que el menor tiempo corresponde siempre a la primera fila de la tabla de datos de entrada y el resto están siempre ordenados de menor a mayor (incluyendo el tiempo de simulación).
 - Actualización de los límites de los campos editables de tiempo al seleccionar un menor número de escalones: Las filas de la tabla de datos de entrada se muestran o se ocultan en función del número de escalones seleccionados, por lo que, mediante esta medida, se garantiza que, en la última fila visible, el límite superior de tiempo es siempre el máximo establecido para la simulación (200 min).
 - Redondeo de decimales (al entero más cercano) en los campos editables de tiempo, lo que impide al usuario introducir dos escalones consecutivos en un intervalo de tiempo demasiado pequeño.
- Medidas que impiden la identificación tras evaluar los datos de entrada de la simulación
 - Comprobación de que dos campos editables de tiempo consecutivos no tengan el mismo valor. De esta forma se elimina el riesgo de incompatibilidad por dos escalones definidos de distinto valor y aplicados en el mismo instante de tiempo.
 - Comprobación de que no hay singularidad en la matriz $M = X^T X$, con lo que se garantiza que la información proporcionada por los datos de entrada puede utilizarse para la identificación paramétrica.

4.6 Modificación de funciones de transferencia asociadas al proceso

Desde la ventana de “Ensayos”, es posible acceder a una herramienta para modificar las funciones de transferencia guardadas en la GUI, que aproximan las diferentes dinámicas presentes en el proceso (Figura 12). Estas funciones están, por defecto, identificadas en torno al punto de operación nominal y se utilizan para el cálculo automático de los parámetros de control PID y “*feedforward*”, así como para la obtención del modelo de respuesta en escalón utilizado en el algoritmo DMC. El usuario puede, por tanto, guardar en la memoria de la GUI una función de transferencia identificada en una zona de operación concreta, de forma que le permita, posteriormente, sintonizar un controlador adecuado para el punto de funcionamiento deseado. Además, esta herramienta dispone de varias opciones para analizar la función de transferencia que se desea guardar:

- Respuesta temporal del sistema ante una entrada de tipo escalón unitario
- Respuesta en frecuencia y obtención de los márgenes de fase y de ganancia
- Gráfica del lugar de las raíces
- Modelo equivalente en espacio de estados y análisis de la controlabilidad

Por otro lado, se permite:

- Cargar la función de transferencia guardada en memoria
- Guardar la función de transferencia en memoria

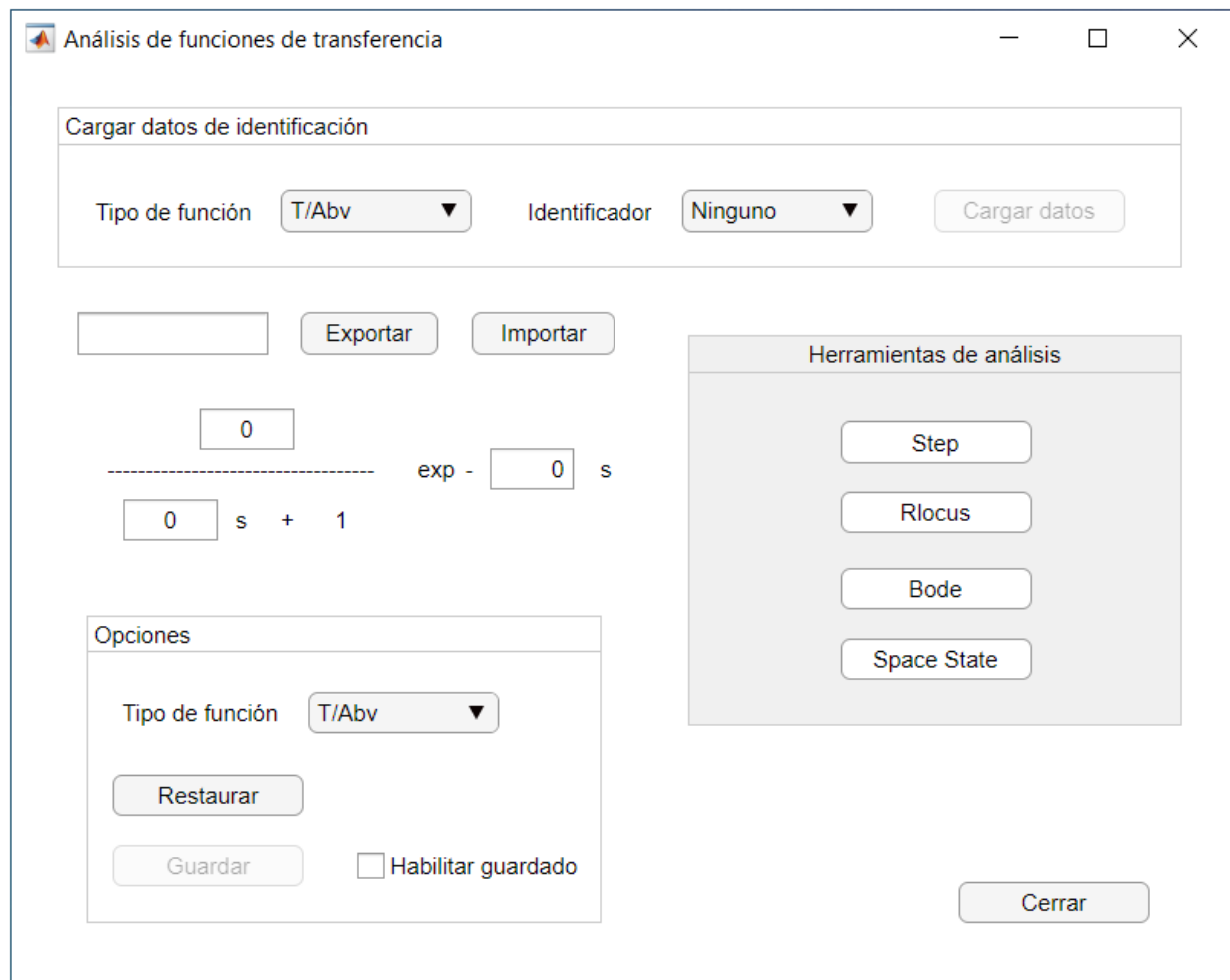


Figura 12. Herramienta de análisis y modificación de funciones de transferencia

Nótese que en la GUI se utiliza la abreviatura “Abv” para representar la abertura de la válvula de refrigeración en tanto por ciento (Figura 12), y recuérdese que la válvula de control es de tipo “*air-to open*”.

5 SISTEMA DE CONTROL POR REALIMENTACIÓN

En esta sección se realiza una breve descripción sobre los controladores que se han implementado en cualquiera de los modelos de reactor CSTR que integra la interfaz gráfica, así como sus ecuaciones y los métodos de diseño empleados en cada caso.

5.1 Control PID

5.1.1 Algoritmo de control PID en tiempo continuo

El controlador PID es uno de los algoritmos de control más utilizados comercialmente, ya que puede ofrecer un buen desempeño en un gran número de procesos industriales. Debido a su popularidad, existen muchos tipos de reglas de sintonización, tanto analíticas como empíricas, resultando ser un controlador especialmente útil en aquellos procesos donde no se tiene un modelo de la planta [8]. A continuación, se desarrolla un breve resumen sobre los fundamentos teóricos de este tipo de controlador.

La estructura PID más extendida a nivel académico es la ideal o estándar y viene dada por la expresión:

$$u(t) = K_c \left(e_r(t) + \frac{1}{T_i} \int_0^t e_r(\tau) d\tau + T_d \frac{de_r(t)}{dt} \right) \quad (83)$$

donde $u(t)$ es la señal de control y $e_r(t)$ el error de seguimiento de referencia.

Las acciones presentes en este tipo de control son tres: proporcional, integral y derivativo [9]:

- **Acción proporcional:**

Corresponde al término de la señal de control que es proporcional a la señal de error. La expresión de un controlador proporcional (P) es la siguiente:

$$u(t) = K_c e_r(t) + u_{os} \quad (84)$$

donde u_{os} es el valor de la salida del controlador cuando el error es nulo (valor de “offset”) y puede ajustarse para eliminar dicho error en un punto de consigna determinado.

La acción proporcional influye en la estabilidad del sistema y puede ajustarse para variar la agresividad y la robustez, así como para reducir el error en régimen estacionario de la respuesta temporal ante una entrada en escalón.

Por otro lado, el signo de la constante proporcional, K_c , define si el controlador es de acción directa o inversa. Se trata de acción directa cuando ante una subida en la variable controlada, el controlador responde aumentando la variable de salida del mismo, y acción inversa cuando responde disminuyéndola.

Una variante del control P se obtiene actuando sobre la variable controlada, con el objeto de eliminar el pico proporcional:

$$u(t) = -K_c y(t) \quad (85)$$

Nótese que esta última expresión es válida para la regulación del proceso, pero no para su control sobre distintos puntos de operación, ya que no admite ningún valor de consigna para el seguimiento de referencias.

- **Acción integral:**

Corresponde al término de la señal de control que es proporcional a la integral del error. Un controlador proporcional-integral (PI) para una estructura estándar (algoritmo ideal) tiene la siguiente expresión:

$$u(t) = K_c \left(e_r(t) + \frac{1}{T_i} \int_0^t e_r(\tau) d\tau \right) \quad (86)$$

Mediante la integración, se proporciona información al controlador sobre el error acumulado a lo largo del tiempo, de forma que, si el sistema se estabiliza, el error de seguimiento tiende a ser nulo para valores finitos del tiempo integral (error en régimen permanente nulo). Este parámetro también afecta a la respuesta transitoria, obteniéndose una evolución más agresiva para valores bajos de T_i o una mayor robustez para valores altos. Como contrapartida, la acción integral introduce un polo en el origen, lo que conlleva a una pérdida de estabilidad relativa. Esto puede reflejarse en una respuesta oscilatoria que, generalmente, se considera indeseable.

Cuando un controlador incorpora acción integral, existe la posibilidad de que ocurra el denominado efecto “*windup*”. Este fenómeno tiene lugar cuando los actuadores saturan debido a sus limitaciones físicas, provocando una continua acumulación del error. Si esta saturación se alcanza debido a un cambio significativo en el valor nominal de consigna del controlador o de la perturbación, es posible que la integral del error aumente o disminuya indefinidamente, de modo que deje al sistema fuera de control incluso más allá del momento en el que la excesiva perturbación o referencia cambien, tardando mucho más tiempo en conseguir mantener bajo control el sistema, dando lugar a oscilaciones o incluso pudiendo llegar a inestabilizar al sistema.

Para evitarlo, existen algunos métodos “*anti-windup*” en tiempo continuo, desarrollados para controladores analógicos, como es el caso de algoritmos incrementales o “*back calculation*”, sin embargo, debido a que, actualmente, la mayoría de controladores son digitales, suelen emplearse algoritmos en tiempo discreto. Una solución simple consiste en desactivar la acción integral cuando esta se satura, permitiendo su activación una vez que la saturación del (o de los) elemento(s) de control desaparece.

- **Acción derivativa:**

Corresponde al término de la señal de control que es proporcional a la derivada del error. Un controlador proporcional-derivativo (PD) para una estructura estándar tiene la siguiente expresión:

$$u(t) = K_c \left(e_r(t) + T_d \frac{de_r(t)}{dt} \right) \quad (87)$$

La acción derivativa proporciona al controlador información sobre el valor de la pendiente de la curva del error, de forma que se obtiene una predicción de su evolución, permitiendo mejorar la estabilidad y reducir la sobreoscilación de la respuesta temporal.

Nótese que la acción derivativa pura no es físicamente realizable, al no poder obtenerse la derivada del error en el instante actual (tangente de la curva del error en el último punto calculado), por lo que deben llevarse a cabo aproximaciones de la misma.

La principal desventaja de derivar la señal de error es que puede producirse un “pico derivativo” (“*derivative kick*”). Este fenómeno ocurre cuando se realiza un cambio brusco en el valor de consigna, produciéndose una rápida variación en la derivada del error. El resultado es un repentino aumento del esfuerzo de control que puede producir efectos indeseados en la salida de la planta.

Además, debido a que el ruido de medida induce variaciones abruptas sobre la señal de error, se producen cambios indeseados en su derivada que son amplificadas por la acción derivativa. Por ello, no debe usarse esta acción en los procesos donde el ruido es alto (dando lugar a un controlador PI).

Estas desventajas se mitigan parcialmente empleando un filtro paso bajo, que viene dado por la siguiente ecuación.

$$\alpha T_d \frac{du_D(t)}{dt} + u_D(t) = K_c T_d \frac{de_r(t)}{dt} \quad (88)$$

donde $u_D(t)$ es la contribución de la acción derivativa a la señal de control. El parámetro α se relaciona con la constante de tiempo del filtro mediante la siguiente expresión:

$$\tau_f = \alpha T_d \quad (89)$$

Realizando la transformada de Laplace a la ecuación (88) se tiene:

$$\frac{u_D(s)}{e_r(s)} = K_c \frac{T_d s}{\alpha T_d s + 1} \quad (90)$$

Para reducir el pico derivativo, la acción derivativa debe actuar sobre la variable controlada, resultando en la siguiente expresión para el controlador PD:

$$u(t) = K_c \left(e_r(t) - T_d \frac{dy(t)}{dt} \right) \quad (91)$$

La estructura de control PID implementada en la interfaz gráfica consiste en el algoritmo estándar. La expresión del controlador resultante se indica a continuación:

$$u(t) = K_c \left(e_r(t) + \frac{1}{T_i} \int_0^t e_r(\tau) d\tau \right) + u_D(t) \quad (92)$$

donde $u_D(t)$ depende de la configuración aportada por el usuario. Expresando la ecuación en el dominio de \mathcal{L} para el caso de actuación sobre el error, se tiene:

$$u(s) = K_c \left(1 + \frac{1}{T_i s} + \frac{T_d s}{\alpha T_d s + 1} \right) e_r(s) \quad (93)$$

El usuario, desde una ventana de configuración, puede:

- Incluir o eliminar la acción integral y/o la acción derivativa del algoritmo de control PID.
- Sintonizar un control proporcional puro con actuación sobre la señal controlada o señal de error.
- Configurar o eliminar el filtro de la acción derivativa.
- Seleccionar si la acción derivativa actúa sobre la variable controlada o sobre la señal de error.

5.1.2 Discretización del controlador PID

Para poder llevar a cabo una simulación por computación del proceso controlado, es necesario discretizar la ecuación del controlador PID, de forma que el valor de la señal de control se actualice en cada ciclo del bucle descrito en la Tabla 4 (Algoritmo 1: Simulación del sistema controlado).

Debido a su reducido coste computacional y a su adecuada precisión, se ha utilizado el método de Euler para aproximar la integración de la señal de error:

$$IE_{k+1} = IE_k + E_k \cdot \Delta t_k \quad (94)$$

La derivada de error puede aproximarse mediante el método de Euler hacia atrás:

$$De_{r,k} = \frac{e_{r,k} - e_{r,k-1}}{\Delta t_k} \quad (95)$$

De esta forma, la salida del controlador sin filtro en la acción derivativa en cada instante viene dada por:

$$u_k = u_0 + K_{cd} \left(e_{r,k} + \frac{1}{T_{id}} \cdot IE_k + T_{dd} \cdot DE_k \right) \quad (96)$$

Para aproximar el término derivativo, es preferible analizar las ecuaciones en el dominio de Z . La transformada Z de una secuencia temporal viene dado por:

$$\mathcal{Z}\{x(n)\} = \sum_{k=-\infty}^{\infty} -x(k)z^{-k} \quad (97)$$

donde la relación entre las variables complejas z y s , se muestra a continuación:

$$z = e^{\Delta t_k \cdot s} \quad (98)$$

Nótese que el inverso de z equivale a un retardo de Δt_k unidades de tiempo, por lo que, a partir de la ecuación (95), se deduce la expresión en el dominio de Z de la derivada de la señal de salida:

$$DE(z) = \frac{1 - z^{-1}}{\Delta t_k} e_r(z) \quad (99)$$

Existen diversos métodos para aproximar el filtrado de la acción derivativa en tiempo discreto. Debido a su simple implementación y a su reducida amplificación de ruido respecto a otros métodos, se ha optado por sustituir el operador de Laplace “ s ” por una diferencia regresiva [10]:

$$s = \frac{1 - z^{-1}}{\Delta t_k} \quad (100)$$

Aplicando la ecuación (100) en el término derivativo del controlador PID en tiempo continuo, dado por la ecuación (90) se tiene:

$$\frac{u_D(z)}{e_r(z)} = K_c \frac{T_{dd}N_f - T_{dd}N_f z^{-1}}{N_f \Delta t_k + T_{dd} - T_{dd} z^{-1}} \quad (101)$$

donde T_{dd} es el tiempo derivativo y N_f es un parámetro de diseño del filtro, que se corresponde con el inverso de α , y cuya relación con la constante de tiempo del mismo viene dada por:

$$\tau_f = \frac{T_{dd}}{N_f} \quad (102)$$

Expresando la ecuación (101) mediante una relación de recurrencia se tiene:

$$u_{Dk} = \frac{T_{dd}}{T_{dd} + N_f \Delta t_k} u_{Dk-1} + \frac{K_c T_{dd} N_f}{T_{dd} + N_f \Delta t_k} (e_{r,k} - e_{r,k-1}) \quad (103)$$

La Figura 13 muestra el desempeño de la acción derivativa para tres casos con diferente de implementación. Los resultados se han obtenido realizando varias simulaciones del reactor CSTR encamisado para las mismas condiciones del ensayo, realizando un cambio en la referencia en el instante $t = 0,5 \text{ min}$, con un tiempo derivativo de $T_{dd} = 0,1$ y sin simular ruido de medida. Puede comprobarse que derivando la señal de salida en lugar del error y añadiendo un filtro se obtiene una mayor suavidad en la señal de control y se evita el efecto del pico derivativo.

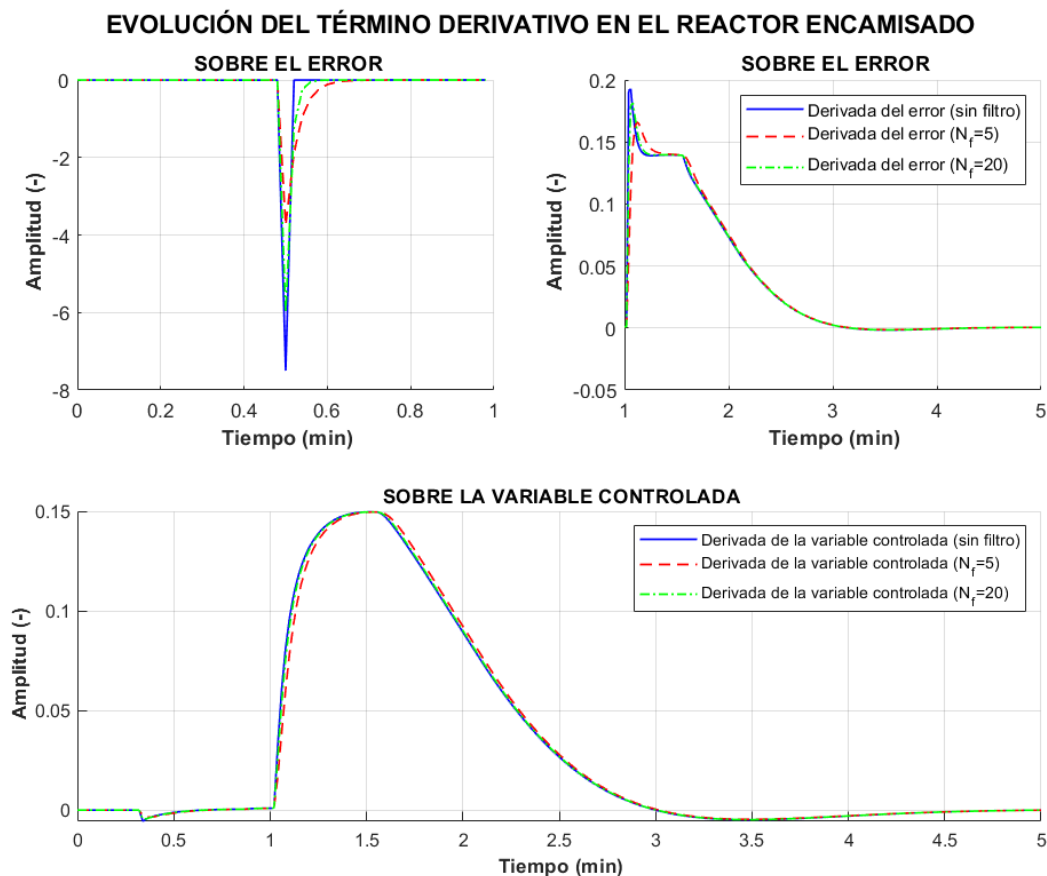


Figura 13. Comparación de desempeño entre distintos tipos de acción derivativa

Con la adición de este filtro, la salida del controlador en cada instante se obtiene con la siguiente expresión:

$$u_k = u_0 + K_{cd} \left(E_k + \frac{1}{T_{id}} \cdot IE_k \right) + u_{Dk} \quad (104)$$

donde para pasar de variables incrementales a variables absolutas, se ha añadido el término u_0 , que corresponde al valor nominal de la abertura de la válvula de control. Para evitar el efecto “windup”, se ha optado por desactivar la acción integral en caso de saturación de la válvula de control dispuesta en la línea de refrigerante.

5.1.3 Implementación en la GUI del algoritmo de control PID

Debido a que el controlador simulado en la GUI es digital, se han incluido módulos de conversión, con el objetivo de procesar los datos de los sensores y obtener una ganancia estática adimensional. La Figura 14 muestra en un diagrama de bloques la estructura de control, donde f_c es la conversión de la señal de control calculada en señal eléctrica (de % a V o mA) y f_{pct} es la sensibilidad del bloque conversor de la salida del transmisor (en mA o V) a un valor porcentual. Para simplificar la ilustración, se ha considerado en la figura un modelo de reactor lineal.

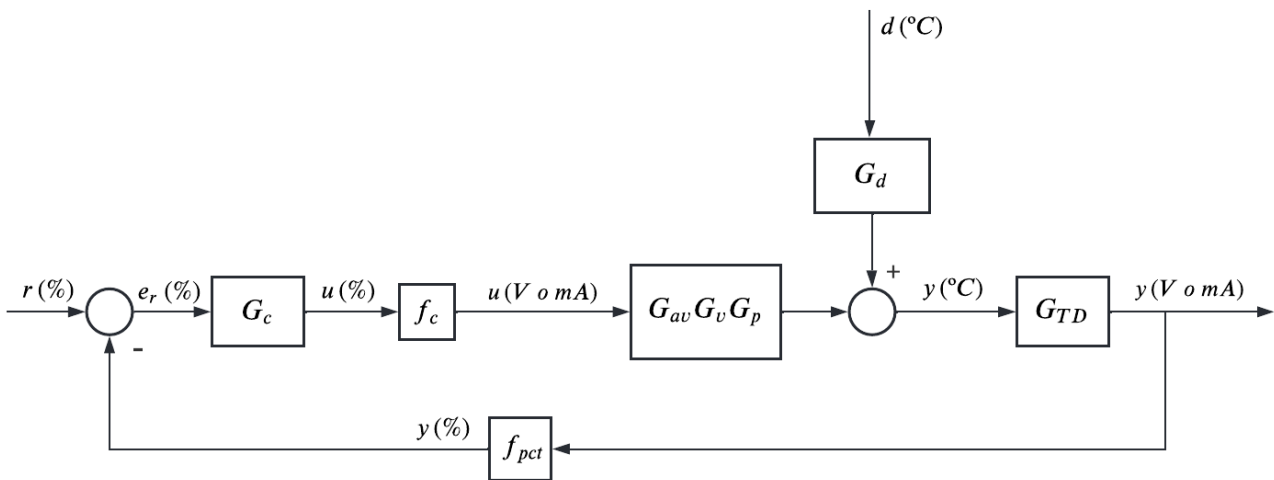


Figura 14. Diagrama de bloques de la implementación en la GUI implementado en la GUI

Nótese que $f_c \cdot G_{av}(s) = 1$. La referencia es introducida por el usuario en valores de $^{\circ}C$ que, posteriormente, es transformada en un valor porcentual por el propio controlador para obtener la señal de error. Esta estructura implica un procesamiento de los datos del sensor-transmisor y una conversión posterior de la señal de control a magnitud eléctrica, proporcionando una mayor flexibilidad al sistema de control. De esta manera, en el proceso real, sería posible emplear cualquier tipo de transmisor, incluyendo sensores con adaptadores para salida digital mediante protocolos de comunicación industriales, así como diferentes tipos de actuadores.

El modelo aproximado del proceso que proporciona la herramienta de identificación paramétrica incluye la dinámica de la válvula de control, $G(s) = G_v(s) \cdot G_p(s)$. Esta función de transferencia está aislada de los componentes eléctricos que componen el sistema, por lo que, para sintonizar un controlador PID mediante los métodos descritos en el apartado 5.1.4, debe incluirse también el producto $G_{TD}(s) \cdot f_{pct}$. Teniendo en cuenta que el tiempo de respuesta del transmisor (en este caso, considerado nulo) se incluye en la función de transferencia identificada, la incorporación de este nuevo producto al modelo identificado es equivalente a realizar una transformación de $(^{\circ}C/\%)$ a $(\%/ \%)$ en función del rango de medición del sensor mediante una sencilla multiplicación cruzada.

5.1.4 Diseño de un controlador P/PI/PID mediante reglas de sintonización

Se ha diseñado una funcionalidad en la GUI para proporcionar al usuario, de forma automática, los valores del controlador P, PI o PID, en función de la opción de agresividad seleccionada, pudiendo elegir entre una respuesta agresiva, moderada o conservadora. Para ello, se ha hecho uso de reglas de sintonización basadas en ITAE [11] (para la acción proporcional) y en IMC [9] (para el control PI y PID), dadas por las siguientes expresiones:

- Control P

$$K_c = \frac{1}{2 \cdot K_p} \left(\frac{\tau_p}{\tau_{mp}} \right)^{1,08} \quad (105)$$

- Controlador PI

$$K_c = \frac{\tau_p}{K_p \cdot (\tau_{lc} + \tau_{mp})} \quad (106)$$

$$T_i = \tau_p \quad (107)$$

- Controlador PID

$$K_c = \frac{1}{K_p} \frac{(\tau_p + 0,5 \cdot \tau_{mp})}{(\tau_{lc} + 0,5 \cdot \tau_{mp})} \quad (108)$$

$$T_i = \tau_p + 0,5 \cdot \tau_{mp} \quad (109)$$

$$\frac{\tau_p \cdot \tau_{mp}}{2\tau_p + \tau_{mp}} \quad (110)$$

donde K_p y τ_p son, respectivamente, la ganancia estática y constante de tiempo del sistema en lazo abierto, τ_{lc} la constante de tiempo deseada en lazo cerrado y τ_{mp} el retardo del proceso. Una regla comúnmente utilizada consiste en seleccionar un valor para el parámetro de ajuste de $\tau_{lc} = 3\tau$ para obtener un controlador robusto y $\tau_{lc} = \tau_p$ para una sintonización agresiva [9]. Aunque en la literatura también existen otras reglas, en este caso, se ha optado por determinar de forma experimental tres posibilidades para τ_{lc} :

- Agresivo: $\tau_{lc} = 0,5 \cdot \tau_p$
- Moderado: $\tau_{lc} = \tau_p$
- Conservador: $\tau_{lc} = 2 \cdot \tau_p$

Estos valores se han seleccionado tras analizar el comportamiento de los reactores con serpentín y con camisa de refrigeración para distintos valores de la constante de tiempo en lazo cerrado deseada. El modelo lineal utilizado para calcular los parámetros de control puede modificarse desde la ventana dedicada a la identificación. La regla utilizada para el cálculo de τ_{lc} no es modificable, pero se da la posibilidad al usuario de introducir manualmente los parámetros de control, pudiendo utilizarse de manera externa cualquier regla de sintonización.

Para proporcionar al usuario una solución, se ha diseñado un controlador PI a partir de un modelo lineal de primer orden aproximado, obtenido mediante el método de identificación descrito en el capítulo 4, tras haber realizado un ensayo en escalón abarcando todo el rango de operación del reactor. Los parámetros diseñados son mostrados al usuario por defecto en la ventana de configuración del controlador.

En este caso, se han realizado varias simulaciones para distintos valores de τ_{lc} próximos a τ_p , con el objetivo de obtener un controlador a partir de un criterio de compromiso entre velocidad y suavidad en la respuesta del proceso. El retraso introducido por el mantenedor de orden cero no se ha tenido en cuenta, ya que se considera despreciable en comparación con el tiempo de respuesta del proceso.

Con la solución diseñada, tanto en el caso del reactor con serpentín como en el reactor encamisado, se obtienen unos márgenes de ganancia y fase de $M_G > 3,16$ (10 dB) y $M_f > 60^\circ$ para el sistema controlado en lazo abierto, dotando al control de una gran robustez.

En la Figura 15, se muestra la respuesta temporal del reactor encamisado con un controlador PI ajustado mediante una sintonización IMC, para distintos valores de τ_{lc} .

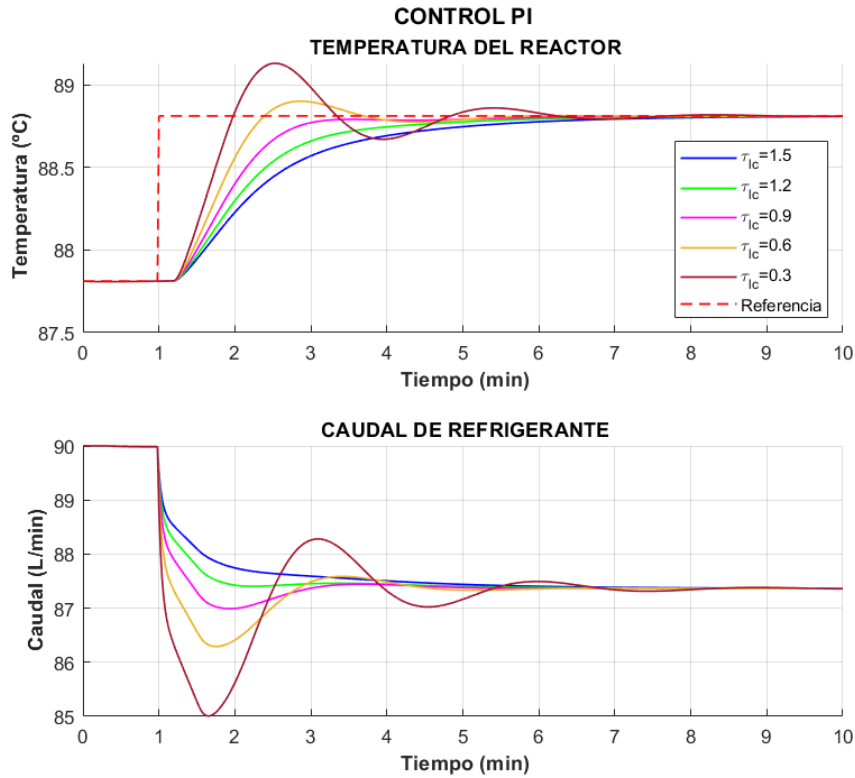


Figura 15. Comparación de desempeño entre distintos controladores PI obtenidos mediante sintonización IMC

Otro método alternativo de sintonización PI es el de SIMC [12], en el que la ganancia proporcional y el tiempo integral se obtienen mediante las ecuaciones (106) y (111), respectivamente.

$$T_i = \min\{\tau_p, 4(\tau_{lc} + \tau_{mp})\} \quad (111)$$

donde generalmente, $\tau_{lc} \geq \tau_{mp}$ y $\tau_{lc} > 0,1 \tau_p$. Éste método se ha comparado con la regla IMC para los tres niveles de agresividad definidos en la GUI, a partir de modelos lineales aproximados obtenidos por identificación paramétrica. Los resultados calculados mediante ambos métodos resultan ser idénticos, por lo que se ha optado por incluir en la GUI únicamente la regla IMC.

5.1.5 Acción anticipativa

El valor de la temperatura de entrada de caudal de refrigeración puede variar, especialmente si el fluido procede de etapas anteriores en el proceso de producción. Estas variaciones son, a nivel de control, perturbaciones que afectarán a la salida del sistema y que, dependiendo del controlador implementado, podrán o no ser rechazadas por el lazo de realimentación.

Para reducir los efectos sobre la señal de salida de estas perturbaciones, se ha implementado en la interfaz gráfica la opción de añadir acción anticipativa (“*feedforward*”) a un control PI.

El esquema general de un controlador en lazo cerrado con acción anticipativa se representa mediante la Figura 16:

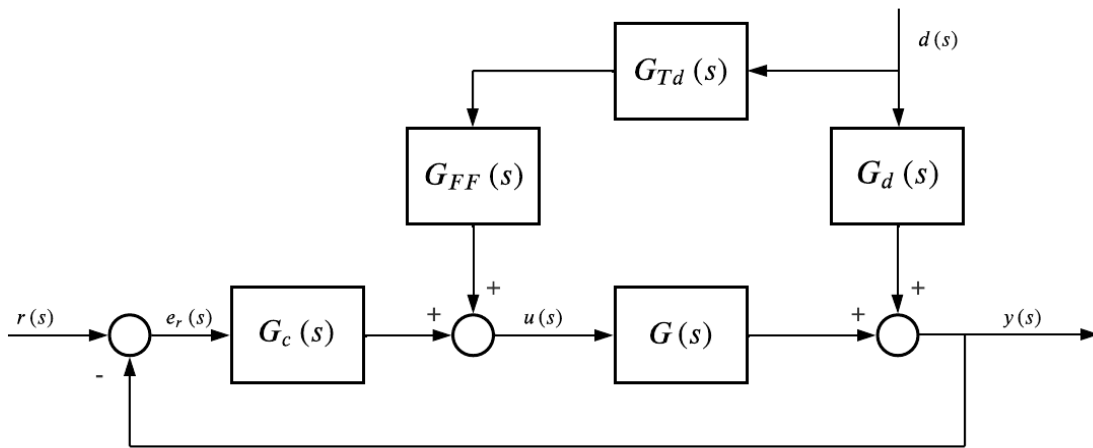


Figura 16. Diagrama de bloques del control en lazo cerrado con acción anticipativa

La ley de control viene dada en tiempo continuo por:

$$u(s) = G_{TD}(s) \cdot G_{FF}(s) \cdot d(s) + G_c(s) \cdot e_r(s) \quad (112)$$

donde $G_{TD}(s)$ es la función de transferencia de un instrumento de medida de la perturbación $d(s)$ y $G_{FF}(s)$ la función de transferencia del controlador “*feedforward*”. La relación $G(s)$ incluye la dinámica del proceso, la de la válvula de control y la de su actuador.

La salida viene dada por:

$$y(s) = G(s) \cdot u(s) + G_d(s) \cdot d(s) \quad (113)$$

donde $G(s) = G_{av}(s) \cdot G_v(s) \cdot G_p(s)$. Considerando que el sistema está en equilibrio ($e_r(s) = 0$), la salida deberá ser nula para cualquier valor que tome la señal de perturbación, por lo que, sustituyendo, se tiene:

$$G_{FF}(s) = -\frac{G_d(s)}{G(s) \cdot G_{TD}(s)} \quad (114)$$

Para una implementación más sencilla, es posible considerar únicamente la ganancia estática de la función de transferencia resultante.

$$G_{FF}(s) = -\frac{K_d}{K \cdot K_{TD}} = -\frac{K_d}{K_{av} \cdot K_v \cdot K_p \cdot K_{TD}} \quad (115)$$

Para el diseño implementado en la interfaz gráfica se ha usado el algoritmo de identificación paramétrica descrito en el apartado 4, con el objetivo de identificar ambas dinámicas y se ha discretizado la función de transferencia $G_{FF}(s)$ resultante, mediante el método de mantenedor de orden cero (“*Zero-Order Hold*”), teniendo en cuenta la compensación de tiempo muerto. Además, se ha calculado la ganancia estática del controlador “*feedforward*”, de forma que el usuario puede elegir si implementarlo de forma estática o dinámica.

En la Figura 17 se muestra una comparativa entre las respuestas temporales obtenidas para los casos de control PI con y sin acción anticipativa, aplicados a los dos modelos de reactor CSTR en el punto de operación nominal. Los resultados se han obtenido aplicando un aumento de 10 °C en la temperatura T_{ce} en el instante $t = 2 \text{ min}$.

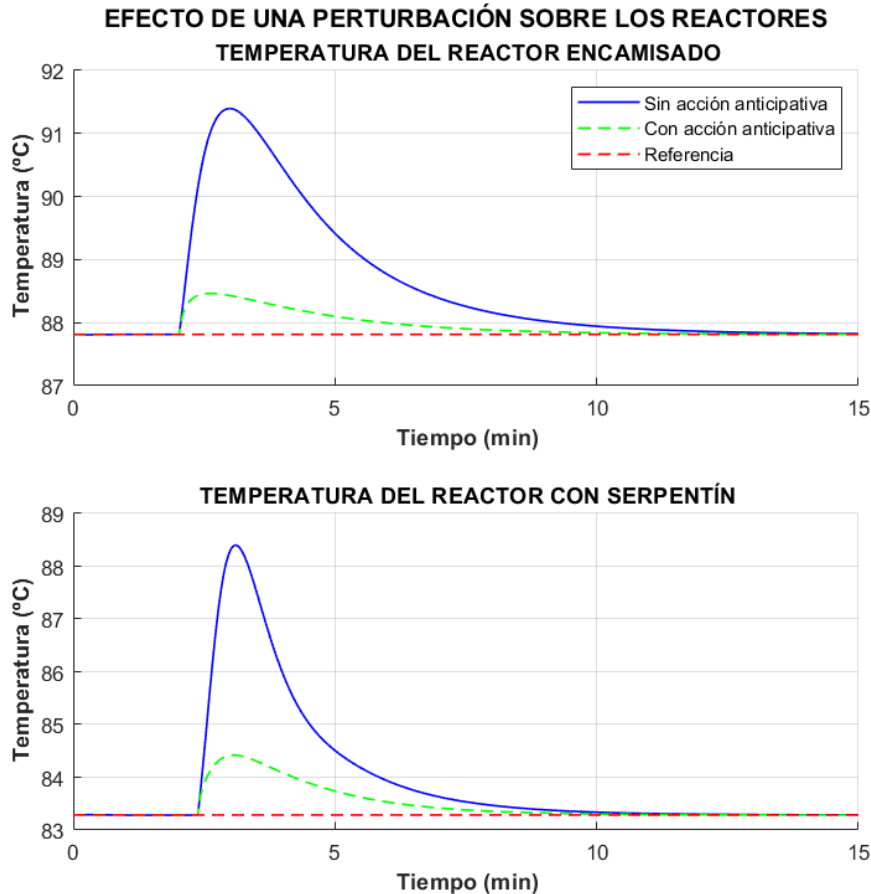


Figura 17. Comparación de las respuestas temporales ante una perturbación para los diferentes modelos controlados por un PI con y sin acción anticipativa

Nótese que, al tratarse de modelos diferentes, aunque los valores iniciales de entrada (F_{c0} y T_{ce0}) sean iguales (véase Tablas 1 y 2), cada simulación comienza en un punto de equilibrio diferente.

Desde la GUI, antes de realizar la sintonización de la acción anticipativa pueden llevarse a cabo las siguientes acciones:

- Modificar la función de transferencia que relaciona la entrada del proceso con la temperatura del reactor desde la ventana de “Ensayos”.
- Modificar la función de transferencia que relaciona la entrada de perturbación con la temperatura del reactor desde ventana de “Ensayos”.

El usuario, a través de una ventana dedicada, dispone de diversas opciones para sintonizar la acción anticipativa:

- Indicar qué función de transferencia aproxima la relación entre la temperatura del reactor y la entrada de perturbación.
- Indicar qué función de transferencia aproxima la relación entre la temperatura del reactor y la variable manipulada.
- Cargar desde la memoria de la GUI o importar desde un archivo de texto la función de transferencia que aproxima la relación entre la temperatura del reactor y la entrada de perturbación.
- Cargar desde la memoria de la GUI o importar desde un archivo de texto la función de transferencia que aproxima la relación entre la temperatura del reactor y la variable manipulada.
- Establecer la dinámica de la acción anticipativa.

- Deducir automáticamente la dinámica de la acción anticipativa.
- Evaluar el controlador para distintos puntos de operación.
- Seleccionar si se desea que el algoritmo a aplicar sea dinámico o estático.

La Figura 18 muestra en un diagrama de bloques la estructura de control con acción anticipativa implementada en la GUI. Esta estructura implica el procesamiento de los datos de los transmisores, proporcionando una mayor flexibilidad al sistema de control.

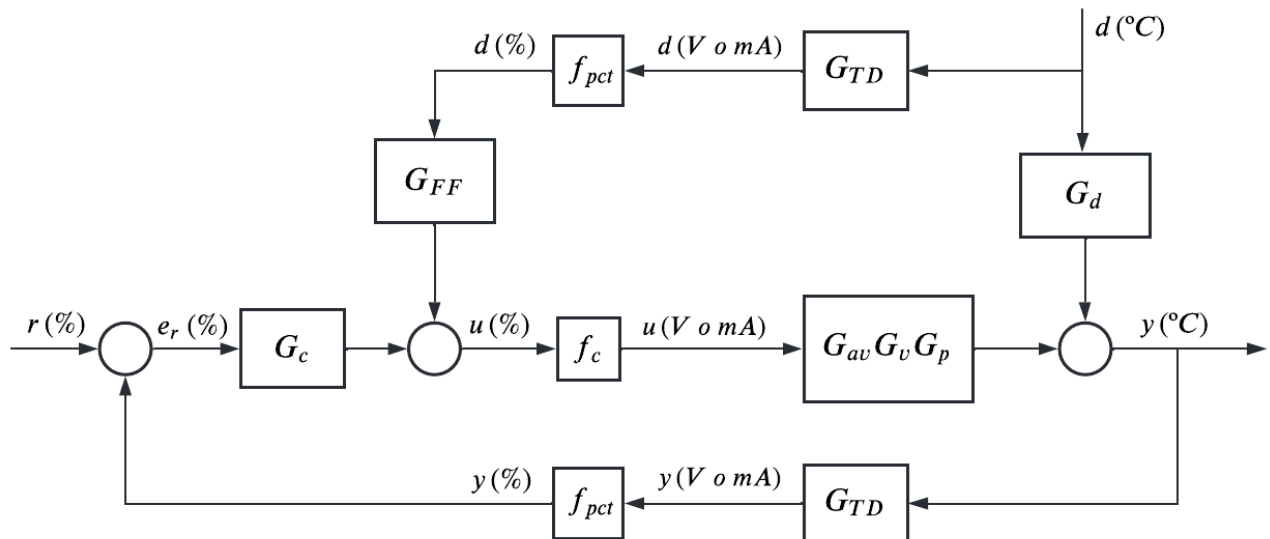


Figura 18. Implementación en la GUI del control en lazo cerrado con acción anticipativa

Con el objetivo de obtener un controlador de ganancia adimensional, la acción anticipativa considerada tiene también una ganancia adimensional. Para ello, la expresión del controlador queda:

$$G_{FF}(s) = -\frac{G_d(s)}{f_c \cdot G_{av}(s) \cdot G_v(s) \cdot G_p(s) \cdot G_{TD}(s) \cdot f_{pct}} = -\frac{G_d(s)}{G_T(s) \cdot G_{TD}(s) \cdot f_{pct}} \quad (116)$$

donde $G_T(s) = G_v(s) \cdot G_p(s)$, ya que se tiene $f_c \cdot G_{av}(s) = 1$.

Asimismo, puede utilizarse únicamente la ganancia estática del controlador obtenido como alternativa a la acción anticipativa dinámica.

5.1.6 Estrategia de control en cascada

Es muy común en los procesos químicos el uso de una estrategia de control en cascada, con el objetivo de rechazar perturbaciones que actúan sobre el lazo interno, evitando que éstas lleguen a afectar al controlador principal [13]. Este esquema de control se compone de un controlador primario o maestro que se comunica con un controlador secundario o esclavo, enviándole los valores de consigna necesarios para realizar el seguimiento de la referencia principal.

Para añadir más posibilidades de control dentro de la interfaz gráfica, se ha implementado una opción que permite utilizar esta estructura en el reactor encamisado. En este caso, el controlador maestro (PI o DMC) recibe la información del error de seguimiento de la variable controlada (temperatura del tanque T) y calcula el valor de la temperatura de refrigeración T_c necesaria para alcanzar la referencia. El controlador esclavo (P o PI) recibe esta última señal y se encarga de manipular la válvula de control, de manera que se consiga el caudal de

refrigeración F_c requerido para proporcionar la temperatura de refrigeración deseada a la salida, que se está midiendo. La Figura 19 muestra de forma esquematizada el funcionamiento de este algoritmo de control. Nótese que el sensor-transmisor de la salida del proceso se ha ubicado en la línea de realimentación al controlador maestro para obtener una relación de aspecto más conveniente para su visualización en formato A4. En cambio, el sensor-transmisor asociado al lazo interno debe estar necesariamente en la línea de realimentación al controlador esclavo.

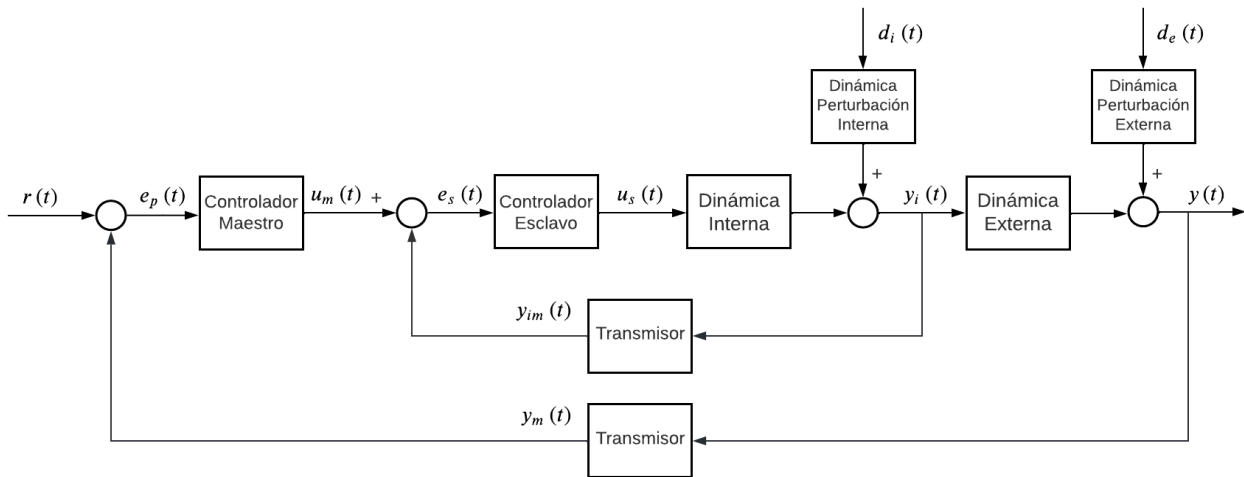


Figura 19. Diagrama de bloques del algoritmo de control en cascada

Las perturbaciones rechazadas por el lazo interno consisten en variaciones de temperatura del refrigerante a la entrada de la camisa, que se traducen en cambios en su temperatura de salida.

Con el objetivo de proporcionar al usuario una solución por defecto, en primer lugar, se ha ajustado manualmente un controlador proporcional para el lazo interno. Para ello, se ha partido de un valor bajo para K_c y, posteriormente, se ha aumentado progresivamente, observando en cada iteración la respuesta temporal correspondiente. El proceso se ha detenido al obtener un error pequeño en régimen permanente y al comprobar que no se producen oscilaciones en la respuesta transitoria. Seguidamente, se ha llevado a cabo una identificación paramétrica del proceso cerrando el lazo interno y con el controlador externo en modo manual. El modelo obtenido se ha utilizado para sintonizar un PI maestro mediante las reglas IMC expuestas en el apartado 5.1.4.

Por otro lado, para facilitar el ajuste en caso de incluir acción integral al lazo secundario, se han establecido tres niveles de agresividad, de forma que pueden obtenerse automáticamente los parámetros de un controlador PI en función de la opción seleccionada por el usuario:

- Agresivo: $\tau_{lc} = 0,05 \cdot \tau_p$
- Moderado: $\tau_{lc} = 0,1 \cdot \tau_p$
- Conservador: $\tau_{lc} = \tau_p$

Estos valores se han obtenido de manera experimental, siguiendo un criterio de alta ganancia y de crecimiento monótono (sin oscilaciones) de la respuesta temporal.

En la Figura 20, se muestra una comparativa entre los resultados obtenidos a partir de simulaciones del reactor CSTR encamisado con diferentes tipos de control y mismas condiciones de ensayo, aplicando un aumento de 10°C en la temperatura T_{ce} en el instante $t = 2 \text{ min}$. Puede comprobarse la mejora introducida por la estructura en cascada, que da lugar a un rechazo de la perturbación más eficaz y eficiente.

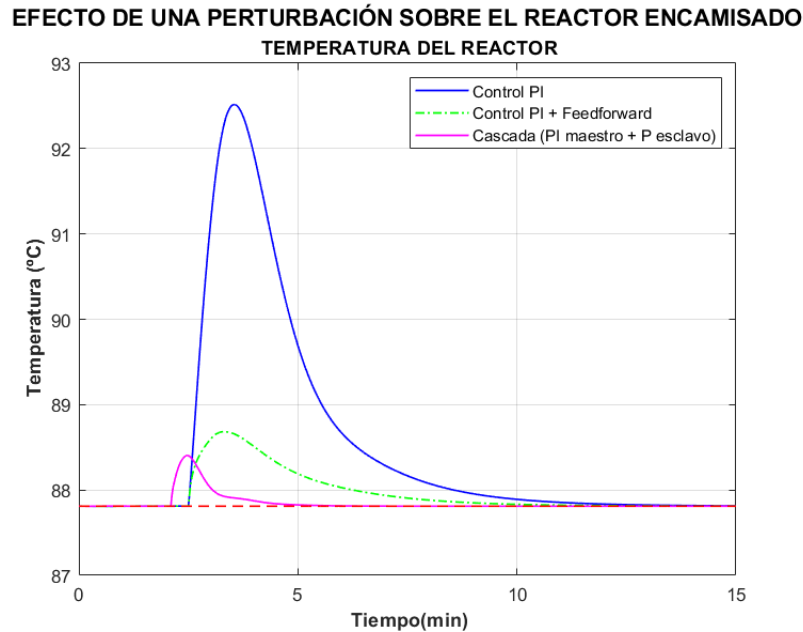


Figura 20. Rechazo de una perturbación (temperatura del refrigerante a la entrada de la camisa del reactor) utilizando diferentes variaciones del algoritmo PI

Además, en el caso de seleccionar en la GUI una estructura de control en cascada, se incluye la posibilidad de introducir un valor para la temperatura de entrada al reactor (T_e) diferente al nominal, de manera que se simulan perturbaciones que afectan a la dinámica externa. Para reducir el impacto de esta variable sobre la respuesta del proceso, se ha incorporado una acción anticipativa, que puede activarse para mejorar el rechazo de la perturbación externa. La sintonización de este controlador viene dada por la ecuación (116), donde debe sustituirse $G_T(s)$ por el modelo resultante de multiplicar la función de transferencia de la dinámica interna controlada por la función de transferencia de la dinámica externa en lazo abierto. En la Figura 21 se muestra una comparación de los resultados obtenidos en un experimento donde se introduce una perturbación externa de 10 °C respecto al valor nominal en el instante $t = 2 \text{ min}$. Puede observarse que, de por sí, el control en cascada rechaza eficazmente la perturbación, y se obtiene una mejora mediante la acción anticipativa o sustituyendo el controlador maestro por un DMC con modelo de perturbación.

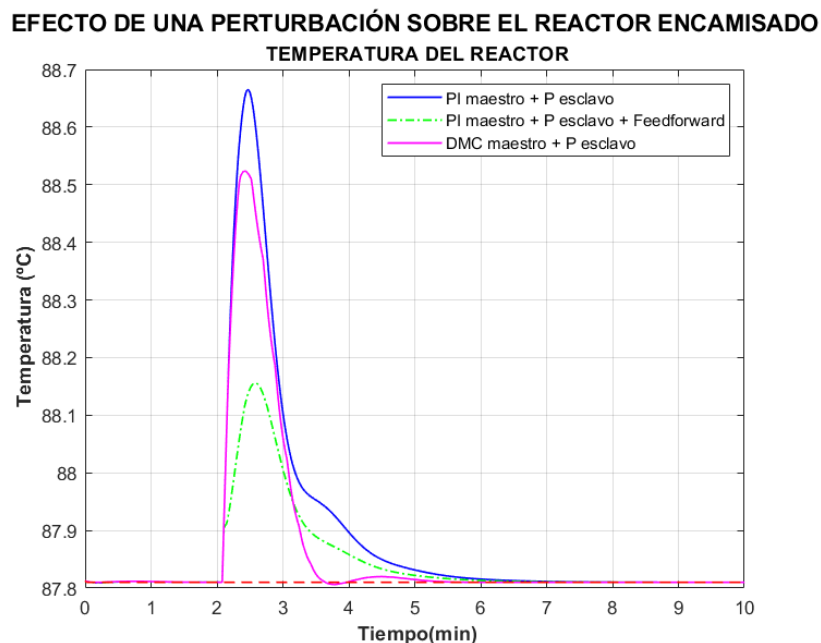


Figura 21. Rechazo de una perturbación externa (temperatura a la entrada del reactor) para distintas estrategias de control

5.2 Control por realimentación del vector de estados

Una alternativa al control PID ampliamente utilizada consiste en la realimentación del vector de estados (“*state feedback*”). Los controladores que implementan este método tienen la ventaja de poder ser diseñados mediante un enfoque de asignación de polos o un enfoque óptimo. Además, son capaces de tratar el problema del control multivariable de forma sencilla y tiene aplicabilidad a una gran variedad de sistemas, entre los que se incluyen lineales, no lineales, variantes e invariantes en el tiempo.

En la aplicación, se han implementado las ecuaciones del controlador “*state feedback*”, tanto para el caso proporcional como para el caso con acción integral simple, proporcionando herramientas para su diseño y simulación. De esta manera, el usuario podrá familiarizarse con estructuras de control moderno, así como con métodos de diseño alternativos a los usados en el control PID.

A continuación, se desarrollan de manera breve los fundamentos teóricos de este tipo de controladores.

5.2.1 Control proporcional con pre-compensador en la entrada

Las ecuaciones (117) y (118) corresponden a la representación en el espacio de estados de un sistema lineal en tiempo discreto y describen, respectivamente, la evolución las variables del estado interno y la salida obtenida en el mismo instante de tiempo.

$$\bar{x}(k+1) = \bar{A}\bar{x}(k) + \bar{B}u(k) \quad (117)$$

$$y(k+1) = \bar{C}\bar{y}(k) + \bar{D}u(k) \quad (118)$$

donde los autovalores de la parte regresiva de la evolución del estado corresponden a los polos del sistema en lazo abierto. Aplicando la ley de control dada en la ecuación (119), se obtiene el sistema autónomo descrito por la ecuación (120).

$$u(k) = -\bar{K}_c\bar{x}(k) \quad (119)$$

$$\bar{x}(k+1) = (\bar{A} - \bar{B}\bar{K}_c)\bar{x}(k) \quad (120)$$

Nótese que el término $(\bar{A} - \bar{B} \cdot \bar{K}_c)$ equivale a la matriz de estados de un nuevo sistema autónomo. \bar{K}_c se le denomina “vector de ganancia de realimentación” y corresponde al parámetro de diseño del controlador, de manera que variando sus elementos es posible ajustar los autovalores de la nueva matriz de estado y, por tanto, los polos del sistema resultante. Los elementos de \bar{K}_c pueden ser ajustados mediante asignación de polos, utilizando la fórmula de Ackermann [8]:

$$\bar{K}_c = [0 \quad 0 \quad \dots \quad 0 \quad 1] \cdot \bar{M}_c \cdot \phi(\bar{A}) \quad (121)$$

donde \bar{M}_c es la matriz de controlabilidad y $\phi(\bar{A})$ es el polinomio característico deseado evaluado en \bar{A} . Asimismo, el diseño del vector de ganancia también puede abordarse mediante un enfoque óptimo con técnicas como LQR o LQG.

En caso de no añadir acción integral, resulta conveniente incorporar un pre-compensador para ajustar la ganancia del sistema y reducir el error de seguimiento. Por tanto, la nueva ley de control viene dada por:

$$u(k) = -\bar{K}_c x(k) + PreC \cdot r(k) \quad (122)$$

donde r es el valor de consigna. Sustituyendo en las ecuaciones de espacio de estados y expresándolo en el dominio de Z se tiene:

$$X(z) = [z\bar{I} - (\bar{A} - \bar{B} \cdot \bar{K}_c)]^{-1} \cdot \bar{B} \cdot PreC \cdot R(Z) \quad (123)$$

$$Y(Z) = \bar{C} \cdot [z\bar{I} - (\bar{A} - \bar{B} \cdot \bar{K}_c)]^{-1} \cdot \bar{B} \cdot PreC \cdot R(Z) \quad (124)$$

Aplicando el teorema del valor final para una entrada en escalón puede calcularse un pre-compensador para cualquier ganancia del sistema en régimen permanente. Para el caso de seguimiento de referencia, se requiere una ganancia unitaria:

$$\bar{C} \cdot [\bar{I} - (\bar{A} - \bar{B} \cdot \bar{K}_c)]^{-1} \cdot \bar{B} \cdot PreC = \bar{I} \quad (125)$$

$$M = \bar{C} \cdot [\bar{I} - (\bar{A} - \bar{B} \cdot \bar{K}_c)]^{-1} \cdot \bar{B} \quad (126)$$

$$PreC = M^{-1} \quad (127)$$

El pre-compensador multiplica al valor de la referencia, por lo que éste, por sí mismo, no tiene ninguna influencia en el rechazo de perturbaciones. Por otro lado, el valor del pre-compensador varía en función del vector de ganancia de realimentación introducido por el usuario en la interfaz gráfica y se calcula, en cada caso, al aplicar la configuración del controlador.

5.2.2 Realimentación del estado con acción integral

Para añadir acción integral al controlador proporcional en tiempo discreto, debe partirse de una aproximación de la integral del error de seguimiento. Mediante el método de Euler se tiene:

$$IE(k+1) = IE(k) + e_r(k)\Delta t_k \quad (128)$$

$$e_r(k) = r(k) - y(k) \quad (129)$$

$$y(k) = \bar{C}x(k) \quad (130)$$

Por lo que puede ampliarse el estado, resultando en:

$$\begin{bmatrix} x(k+1) \\ x_I(k+1) \end{bmatrix} = \begin{bmatrix} \bar{A} & 0 \\ -C\Delta t_k & \bar{I} \end{bmatrix} \begin{bmatrix} x(k) \\ x_I(k) \end{bmatrix} + \begin{bmatrix} \bar{B} \\ 0 \end{bmatrix} u(k) + \begin{bmatrix} 0 \\ \bar{I}\Delta t_k \end{bmatrix} r(k) \quad (131)$$

La nueva ley de control se muestra a continuación:

$$u(k) = [-\bar{K}_c \quad -\bar{K}_I] \cdot \begin{bmatrix} x(k) \\ x_I(k) \end{bmatrix} \quad (132)$$

Al igual que para el caso proporcional, la matriz de ganancia de realimentación puede obtenerse mediante los métodos de asignación de polos o LQR (entre otros).

5.2.3 Diseño de un controlador óptimo por realimentación de estado mediante LQR

Para proporcionar al usuario una solución, se ha ajustado un controlador proporcional y un controlador proporcional-integral por realimentación de estado mediante el método “*Linear Quadratic Regulator*” (LQR). Este método consiste en un enfoque de optimización que trata de minimizar una función objetivo, de acuerdo a unos parámetros de diseño que ponderan la desviación del estado respecto a la posición de equilibrio, \bar{Q} y la energía de control \bar{R} . Esta función de coste viene dada por:

$$J = \sum_{k=0}^{\infty} \bar{x}(k)^T \bar{Q} \bar{x}(k) + \bar{u}(k)^T \bar{R} \bar{u}(k) \quad (133)$$

donde el coste óptimo de la suma de infinitos términos tiene la forma:

$$J^*(x) = \bar{x}^T \bar{P} \bar{x} \quad (134)$$

siendo \bar{P} una matriz de coste simétrica y semidefinida positiva. La ley de control óptima, viene dada por [14]:

$$\bar{u}^* = -(\bar{R} + \bar{B}^T \bar{P} \bar{B})^{-1} \bar{B}^T \bar{P} \bar{A} \bar{x} = -\bar{K}_c \bar{x} \quad (135)$$

La matriz \bar{P} viene dada por la ecuación recursiva de Riccati, que puede resolverse de forma iterativa asignando un valor inicial: $\bar{P}_0 = \bar{Q}$ y resolviendo hasta converger al valor final [14]:

$$\bar{P}_{j+1} = \bar{Q} + \bar{A}^T \bar{P}_j \bar{A} - \bar{A}^T \bar{P}_j \bar{B} (\bar{R} + \bar{B}^T \bar{P}_j \bar{B})^{-1} \bar{B}^T \bar{P}_j \bar{A} \quad (136)$$

Nótese que al realizar el diseño sobre el modelo lineal y, teniendo en cuenta las discrepancias con el proceso real, este algoritmo no tendrá en la práctica el desempeño de un controlador óptimo. Sin embargo, su comportamiento será similar al óptimo en el caso de rechazo de perturbaciones (regulación automática), puesto que el error de modelado es más pequeño en el entorno cercano al punto de linealización.

Las matrices \bar{Q} y \bar{R} se han diseñado para normalizar las variables del problema de optimización. Posteriormente, se ha añadido un factor de ponderación μ que multiplica a los elementos de la matriz \bar{R} con el objetivo de ajustar el controlador:

$$\bar{R}' = \mu \cdot \bar{R} \quad (137)$$

Para el caso del controlador con acción integral, se ha añadido otro factor γ que multiplica al elemento de la diagonal de \bar{Q} correspondiente a la acción integral. Nótese que se obtienen controladores más agresivos aumentando el valor de γ y reduciendo μ , ya que γ pondera el coste de la desviación del estado respecto a la posición de equilibrio y μ pondera el coste de la energía de control.

En la Figura 22, se muestra una comparación de la respuesta temporal del proceso, aplicando un control por realimentación del estado proporcional con pre-compensador y diseñado mediante el método LQR para distintos valores de μ .

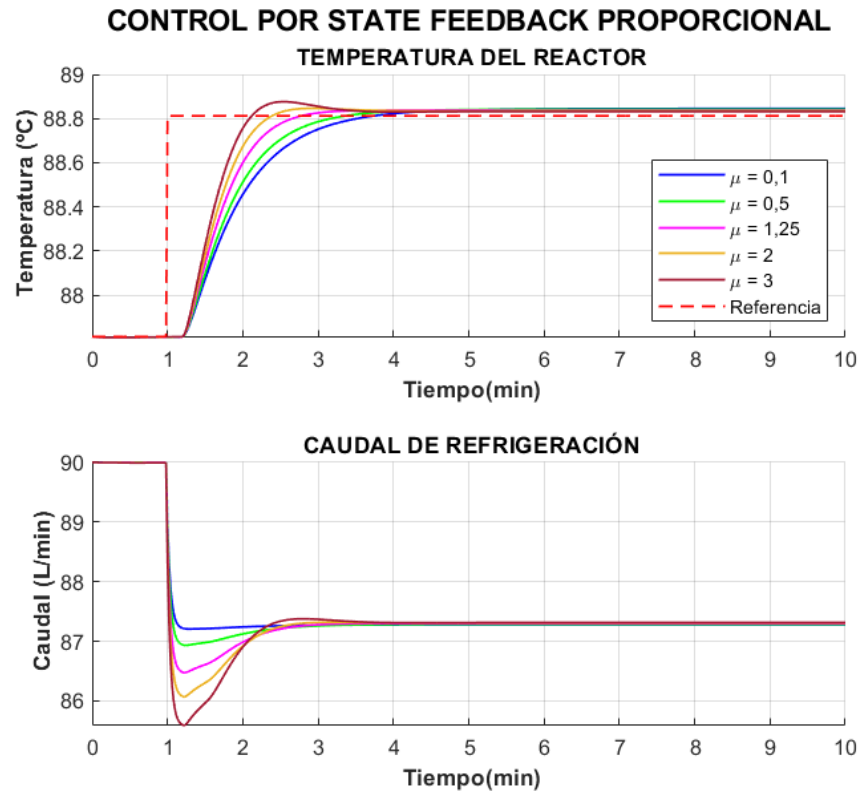


Figura 22. Desempeño de un controlador “*state feedback*” proporcional para distintos valores de μ

La Figura 23 muestra una comparación de la respuesta temporal del proceso, aplicando un control por realimentación del estado con acción integral, para un valor del factor de ponderación de la energía de control $\mu = 2$ y distintos valores de γ .

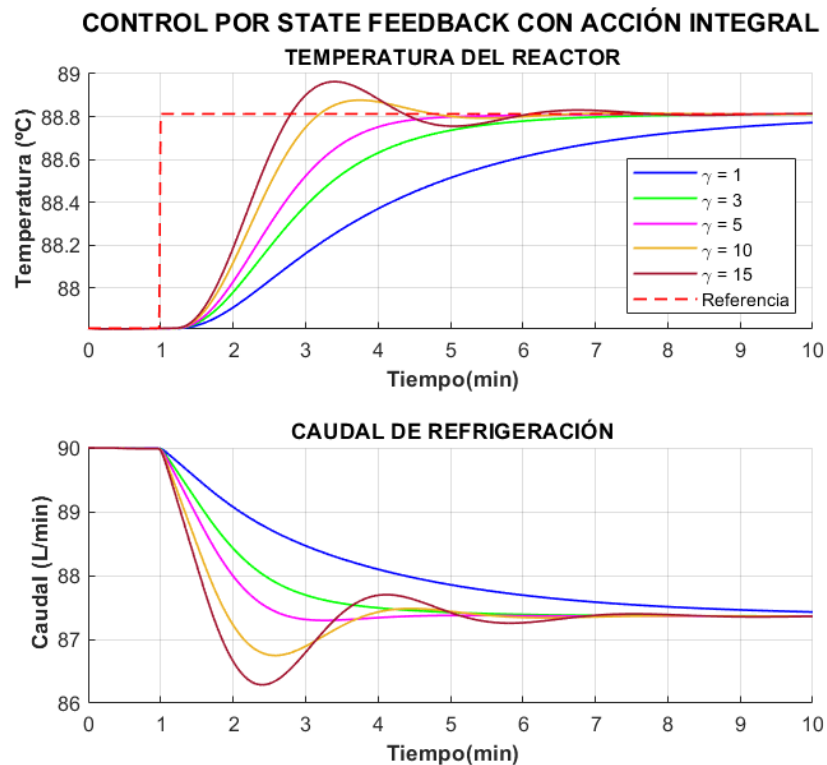


Figura 23. Desempeño de un controlador “*state feedback*” con acción integral para distintos valores de γ

En la Figura 24 se muestra el mismo análisis para el caso del rechazo de una perturbación de tipo escalón unitario aplicado en el instante $t = 1 \text{ min}$. Puede comprobarse que con un valor de $\gamma = 5$ se obtiene una respuesta rápida sin sobreoscilación y una evolución suave de la abertura de la válvula de control.

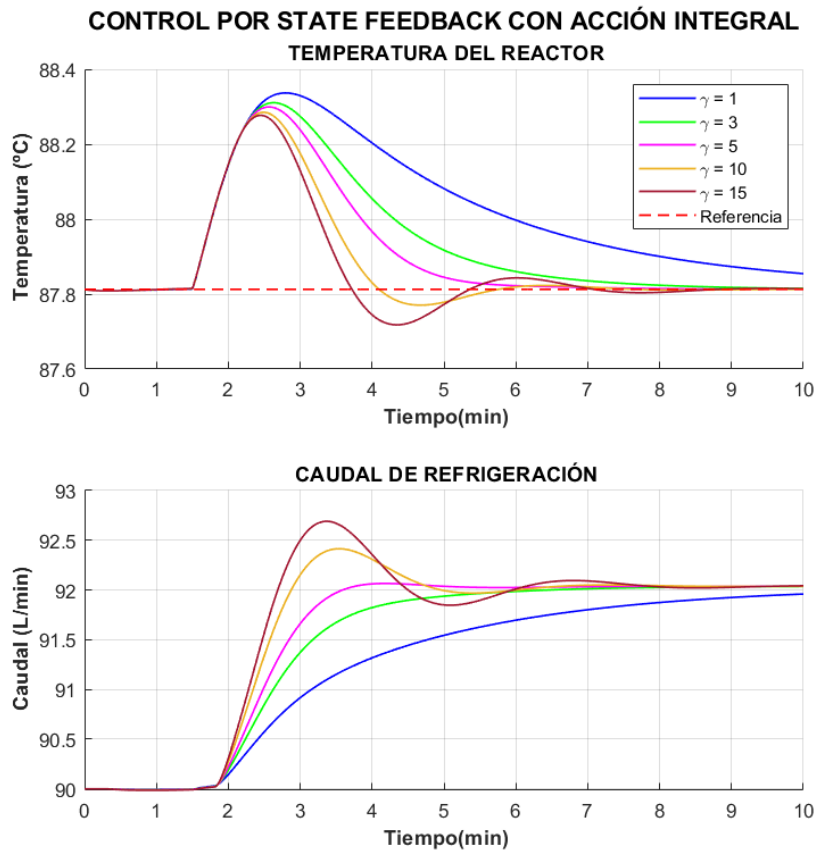


Figura 24. Desempeño en el rechazo de perturbaciones de un controlador “state feedback” con acción integral para distintos valores de γ

5.2.4 Herramienta de sintonización de controladores de realimentación de estado

Para facilitar al usuario la sintonización de controladores de realimentación del estado, se ha incluido en la GUI una herramienta que permite calcular la matriz de ganancia de realimentación mediante los métodos de asignación de polos y LQR (Figura 25).

Los parámetros de ajuste disponibles para el método de asignación de polos son los polos deseados del sistema en lazo cerrado sin parte imaginaria. El usuario puede introducir directamente los valores o seleccionar una de las tres opciones sugeridas, que se detallan a continuación:

- Respuesta agresiva: Los polos corresponden a los del sistema en lazo abierto guardado en la memoria de la aplicación, multiplicado por un factor de 1,2 para el caso proporcional y 1,1 para el caso con acción integral.
- Respuesta moderada: Los polos corresponden a los del sistema en lazo abierto guardado en la memoria de la aplicación, multiplicado por un factor de 1 para el caso proporcional y 0,9 para el caso con acción integral.
- Respuesta conservadora: Los polos corresponden a los del sistema en lazo abierto guardado en la memoria de la aplicación, multiplicado por un factor de 0,85 para el caso proporcional y 0,7 para el caso con acción integral.

Asignación de polos:

p1

p2

p3

p4

LQR:

$$Q = \begin{bmatrix} 1/0.01^2 & 0 & 0 & 0 \\ 0 & 1/1^2 & 0 & 0 \\ 0 & 0 & 1/1^2 & 0 \\ 0 & 0 & 0 & \gamma \cdot 1/1^2 \end{bmatrix}$$

$R = \mu \cdot 1/1^2$

μ

γ

Autoajuste Asignación

Agresivo

Moderado

Conservador

Vector de estados:

$$x = \begin{bmatrix} cA \\ T \\ Tc \\ x_i \end{bmatrix} \begin{bmatrix} mol/l \\ ^\circ C \\ ^\circ C \\ ^\circ C \end{bmatrix}$$

Autoajuste LQR

Agresivo

Moderado

Conservador

Método de diseño

Asignación

LQR

Figura 25. Herramienta para la sintonización de controladores de realimentación de estado

Estos valores se han seleccionado tras analizar el comportamiento del proceso para distintos valores de diseño, de forma que se ofrece al usuario tres soluciones de diferentes prestaciones, donde la diferencia radica en el tiempo de establecimiento y en el transitorio de la respuesta del proceso. El proceso de diseño realizado consiste en partir de los polos del proceso en lazo abierto y, posteriormente, multiplicarlos por un factor cercano a 1 para obtener diferentes respuestas. En caso de incluir acción integral, el factor que multiplica a los polos del sistema aproximado en lazo abierto es más pequeño con el objetivo de compensar la pérdida de estabilidad relativa.

Por otro lado, los parámetros de ajuste disponibles para el método LQR son:

- μ : Factor de ponderación de la energía de control.
- γ : Factor de ponderación de la desviación de la integral del error respecto a la posición de equilibrio. No estará disponible en caso de configurar un controlador proporcional.

Las matrices \bar{R} y \bar{Q} se han diseñado con el objetivo de normalizar las variables del problema de optimización. Para ello, se ha dividido cada elemento de la diagonal de \bar{Q} y \bar{R} por un factor normalizador, resultando en una penalización similar para todas las variables de decisión. De esta forma, los parámetros de ajuste μ y λ proporcionan una sintonización más simple e intuitiva del controlador. Esta normalización puede verse en la representación de las matrices \bar{Q} y \bar{R} en la Figura 25.

Al igual que en el método de asignación de polos, el usuario puede introducir los valores de los parámetros o seleccionar una de las tres opciones predefinidas:

- Respuesta agresiva: Corresponde a un valor de $\mu = 0,25$ para el caso proporcional y unos valores de $\mu = 2$ y $\gamma = 10$ para el caso con acción integral.
- Respuesta moderada: Corresponde a un valor de $\mu = 1$ para el caso proporcional y unos valores de $\mu = 2$ y $\gamma = 5$ para el caso con acción integral.
- Respuesta conservadora: Corresponde a un valor de $\mu = 5$ para el caso proporcional y unos valores de $\mu = 2$ y $\gamma = 2$ para el caso con acción integral.

Estos valores se han seleccionado tras analizar la respuesta temporal del proceso para distintos valores de los parámetros de ajuste, tanto para el reactor encamisado como para el reactor enfriado por serpentín. El proceso de diseño realizado consiste en la obtención de una solución inicial para el control proporcional, $\mu = 1$, para, posteriormente, aumentar y disminuir el valor del parámetro, obteniendo diferentes tiempos de establecimiento. En el caso del control con acción integral, se ha obtenido experimentalmente un valor de μ tal que no penalice de manera excesiva la energía de control para distintos valores de γ . Este último parámetro se ha utilizado para obtener distintas respuestas temporales.

5.3 Control predictivo DMC

5.3.1 Algoritmo de control DMC para sistemas lineales

El control predictivo basado en modelo (MPC) abarca diferentes métodos de control que implementan una aproximación práctica del controlador óptimo, utilizando para ello un modelo matemático del proceso, una acción de control que minimiza una función objetivo y una estrategia de horizonte deslizante.

Entre las numerosas ventajas que proporcionan los algoritmos de control MPC, son particularmente interesantes la efectividad para tratar los sistemas multivariables y el problema del tiempo muerto, la capacidad de introducir acción anticipativa para las perturbaciones medibles y la posibilidad de controlar un gran número de procesos, tanto sencillos como de gran complejidad. Además, estos métodos resultan especialmente útiles cuando se conocen las referencias que van a aplicarse en el futuro [15].

Los controladores MPC presentan la siguiente estrategia (Figura 26):

- Se utiliza un modelo matemático para predecir los valores futuros de la salida del proceso \hat{y} , a lo largo de un horizonte de predicción p . Esta información se calcula en cada instante en función de la información conocida del pasado y la secuencia de control calculada, que se aplicará en el futuro.
- La acción de control se calcula optimizando una función objetivo. Para ello, se considera un horizonte de control m tal que $m \leq p$.
- Se utiliza una estrategia de horizonte deslizante: una vez calculada la secuencia óptima de acciones de control, se aplica la correspondiente al instante actual, eliminando el resto de elementos. El proceso se repite en cada instante de tiempo.

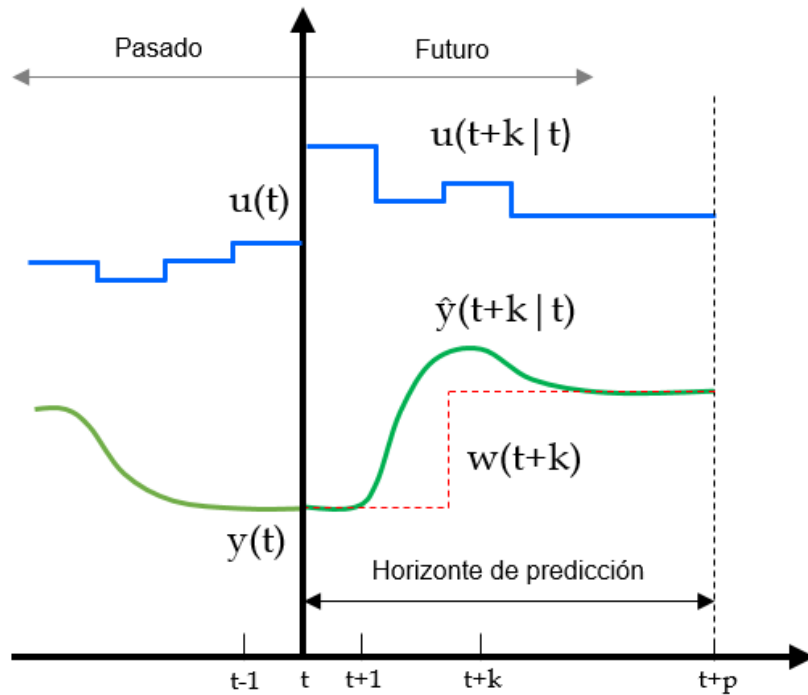


Figura 26. Diagrama de funcionamiento de un controlador predictivo basado en modelo

En este caso, para proporcionar al usuario herramientas de simulación y diseño para el estudio de nociones básicas sobre métodos de control MPC, se ha implementado en la interfaz gráfica un algoritmo DMC con estructura para sistemas SISO, cuyos principios se detallan a continuación.

Dado el modelo matemático en escalón de un sistema:

$$y(t) = \sum_{i=1}^{\infty} \bar{g}_i \Delta u(t-i) \quad (138)$$

donde \bar{g} es un vector que contiene los coeficientes del modelo en escalón. Nótese que el modelo tiene en cuenta el cambio en la entrada de un instante al siguiente y no el valor de la entrada en cada instante.

La predicción de las salidas futuras viene dada por:

$$\hat{y}(t+k|t) = \sum_{i=1}^k \bar{g}_i \Delta u(t+k-i) + f(t+k) \quad (139)$$

El término $f(t+k)$ viene dado por:

$$\begin{aligned} f(t+k) &= y_m(t) + \sum_{i=1}^{\infty} (\bar{g}_{k+i} - \bar{g}_i) \Delta u(t-i) + \sum_{i=1}^{\infty} (\bar{g}d_{k+i} - \bar{g}d_i) \Delta d(t-i) + \\ &+ \sum_{i=1}^k \bar{g}d_i \Delta d(t+k-i) \end{aligned} \quad (140)$$

donde \bar{g} corresponde a los coeficientes de la respuesta en escalón debido a la variable manipulada mientras que $\bar{g}d$ es debido a la perturbación.

A esta última expresión se le denomina “respuesta libre” (o no forzada) y es la señal equivalente a la salida del sistema, teniendo en cuenta todas las acciones de control tomadas en instantes anteriores a k y suponiendo incrementos nulos en la señal de control en los instantes posteriores.

Si se trata de un proceso asintóticamente estable, es posible aproximar la suma de infinitos términos de las ecuaciones (138) y (140) estableciendo un horizonte de tiempo, N , lo suficientemente grande como para captar la dinámica del sistema. En la práctica, es preferible no seleccionar un valor excesivamente grande, ya que repercute en la carga computacional del algoritmo. Los horizontes asociados a los términos del proceso y a los términos de la perturbación pueden ser diferentes, al tratarse de dinámicas distintas (N y N_d respectivamente).

El otro término de la expresión de la predicción de salida en la ecuación (139), $\sum_{i=1}^k g_i \Delta u(t+k-i)$, corresponde a la respuesta forzada, que equivale a la respuesta del sistema suponiendo nulas todas las acciones de control en el pasado (instantes anteriores a k) y aplicando en los instantes presente y futuros una señal de control igual a la secuencia de control que se calculará, en este caso, resolviendo un problema de optimización.

Expresando la predicción de salida en forma matricial, para varios instantes futuros, se tiene:

$$\hat{y} = \bar{G}\bar{u} + \bar{f} \quad (141)$$

La secuencia de acciones de control \bar{u} se obtiene minimizando una función objetivo que depende del error de seguimiento de referencias y del esfuerzo de control:

$$J = \sum_{j=1}^p [\hat{y}(t+j|t) - \bar{w}(t+j)]^2 + \sum_{j=1}^m \lambda [\Delta u(t+j-1)]^2 \quad (142)$$

donde p se denomina “horizonte de predicción”, m “horizonte de control”, λ es un parámetro de diseño utilizado para ponderar los esfuerzos de control y \bar{w} es el vector de las referencias futuras.

Nótese que, para un horizonte de predicción infinito, la secuencia de control calculada en cada instante corresponde a la señal de control óptima. Teniendo en cuenta la simplificación utilizada de los horizontes de tiempo finitos, un adecuado valor para p debe contemplar todo el transitorio de la respuesta en lazo cerrado, por lo que puede tratarse del número de muestras equivalentes al tiempo de establecimiento deseado.

El horizonte de predicción es aquel en el que se evalúa la función de coste definida en la ecuación (142), por lo que la variación de este parámetro tendrá repercusión en la respuesta estimada, así como en el número de elementos de \bar{w} considerados, influyendo en la anticipación al seguimiento de referencias y en el transitorio de la respuesta del sistema.

Por otro lado, el horizonte de control consiste en el número de incrementos de la señal de control considerados para controlar al proceso evaluado en el horizonte de predicción. Generalmente, un valor pequeño para m ayuda a obtener un controlador internamente estable y, para plantas con retardo, es fundamental establecer un valor tal que $m < p$. Un valor para el horizonte de control menor que el de predicción implica que la señal de control óptima será constante al final del horizonte de control.

El problema de optimización tiene solución explícita dada por:

$$u = (\bar{G}^T \bar{G} + \lambda I)^{-1} \bar{G}^T (\bar{w} - \bar{f}) \quad (143)$$

Con la solución calculada, se utiliza una estrategia de horizonte deslizante, aplicando únicamente el primer valor de la secuencia y repitiendo el cálculo en el instante siguiente.

5.3.2 Algoritmos para la implementación en la GUI de un control DMC para sistemas no lineales

Para integrar el control DMC en la interfaz gráfica, de forma que se permita al usuario introducir los parámetros de ajuste (p , m y λ), así como las condiciones de entrada de la simulación (valor de consigna, instante de aplicación de la consigna y valor e instante de la perturbación), se ha implementado el siguiente algoritmo para preparar los datos necesarios para la simulación:

- Las entradas son: \bar{r} , \bar{t}_r , T_s , Δt_k , las ganancias K_p y K_d , las constantes de tiempo τ_p y τ_d y los tiempos muertos τ_{mp} y τ_{md} de los sistemas de primer orden aproximados (proceso y perturbación), el tiempo de muestreo del controlador T_{DMC} , el número de elementos de la respuesta temporal del proceso N y de la dinámica de perturbación N_d , el valor del horizonte de predicción p y horizonte de control m y el parámetro de ponderación de los esfuerzos de control λ .
- Paso 1: A partir de los datos de entrada, se obtiene un vector, \overline{ref} , que contiene todos los valores de consigna aplicados en cada instante de tiempo de muestreo del controlador DMC, a lo largo de toda la simulación, de forma que pueda ser interpretado por el algoritmo de control.
- Paso 2: Se crea un índice, i_{DMC} , y se le asigna el valor $i_{DMC} = 0$. Este índice se utilizará para contar el número de veces que se ejecuta el algoritmo de control.
- Paso 3: Se construye, a partir de los datos de entrada del algoritmo, la estructura de la función de transferencia $G(s)$ que aproxima la dinámica del proceso.
- Paso 4: Se discretiza la función de transferencia mediante el método “Zero-Order Hold”, con un tiempo de muestreo igual al tiempo de muestreo del controlador DMC.
- Paso 5: Se realiza una simulación del sistema lineal en tiempo discreto y se obtienen suficientes elementos de la respuesta temporal para una entrada de tipo escalón unitario ($N + p_{m\acute{a}x}$ elementos), de forma que se capte la dinámica del proceso. Se crea un nuevo vector \bar{g} al que se le asignan los resultados obtenidos.
- Paso 6: Se repiten los pasos 3, 4 y 5 para $G_d(s)$, que aproxima la dinámica de perturbación, y se asignan los resultados a \bar{g}_d .
- Paso 7: Se obtiene la matriz dinámica \bar{G} del algoritmo de control DMC a partir de los resultados obtenidos en el paso 5.
- Paso 8: Se calcula la parte “offline” de la secuencia de control óptima del algoritmo de control DMC, obteniendo la matriz \bar{M} .
- Paso 9: Se extrae la primera fila de la matriz calculada en el paso anterior, que corresponde a la parte “offline” del primer valor de la secuencia de control y se le asigna al vector \bar{K}_{DMC} .
- Paso 10: Se crea un vector nulo $\bar{\Delta U}$, que se utilizará en el algoritmo de control DMC para almacenar las acciones de control aplicadas en cada instante de tiempo.
- Las salidas son: El índice i_{DMC} y los vectores \overline{ref} , \bar{g} , \bar{g}_d , \bar{K}_{DMC} y $\bar{\Delta U}$.

La Tabla 5 (Algoritmo 2: Preparación del control DMC) muestra de forma detallada el desarrollo matemático de este algoritmo.

Tabla 5. Algoritmo 2: Preparación del control DMC

Entrada	: $\bar{r} \in \mathbb{R}^{N_2}, \bar{t}_r \in \mathbb{R}^{N_2}, K_d \in \mathbb{R}, K_p \in \mathbb{R}, \tau_d \in \mathbb{R}^+, \tau_p \in \mathbb{R}^+, \tau_{mp} \in \mathbb{R}^+, \tau_{md} \in \mathbb{R}^+, T_s \in \mathbb{R}^+, \Delta t_k \in \mathbb{R}^+, T_{DMC} \in \mathbb{R}^+, \lambda \in \mathbb{R}^+, N \in \mathbb{N}, N_d \in \mathbb{N}, p \in \mathbb{N}, m \in \mathbb{N}$
Paso 1	: $\overline{ref} \in \mathbb{R}^n$. Obtener \overline{ref} a partir de $\bar{r}, \bar{t}_r, T_s, \Delta t_k$ y T_{DMC}
Paso 2	: $i_{DMC} = 0$
Paso 3	: $G(s) = \frac{K_p}{\tau_p \cdot s + 1} e^{-\tau_{mp} \cdot s}$, donde $s \in \mathbb{C}$
Paso 4	: $G(z) = (1 - z^{-1}) Z \left\{ \frac{G(s)}{s} \right\}$ con tiempo de muestreo: $t_m = T_{DMC}$, donde $z \in \mathbb{C}$
Paso 5	: Obtener, al menos, $N + p_{m\acute{a}x}$ elementos de la respuesta temporal de $G(z)$ para una entrada de tipo escalón unitario. Los resultados se asignan al vector $\bar{g} \in \mathbb{R}^{N+p_{m\acute{a}x}}$
Paso 6	Repetir los pasos 3, 4 y 5 para $G_d(s)$ y asignar los resultados a $\bar{g}_d \in \mathbb{R}^{N_d+p_{m\acute{a}x}}$
Paso 7	: $\bar{G} \in \mathcal{M}_{p \times m}$, tal que $\bar{G} = \begin{bmatrix} g_1 & 0 & \dots & 0 \\ g_2 & g_1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_p & g_{p-1} & \dots & g_{p-m+1} \end{bmatrix}$
Paso 8	: Calcular $\bar{M} = (\bar{G}^T \bar{G} + \lambda I)^{-1} \bar{G}^T$
Paso 9	: $\bar{K}_{DMC} \in \mathbb{R}^p, \bar{K}_{DMC} = [\bar{M}_{1,1} \quad \bar{M}_{1,2} \quad \dots \quad \bar{M}_{1,p}]$
Paso 10	: $\bar{\Delta U} \in \mathbb{R}^{N+1}, \bar{\Delta U} = \bar{0}$
Salida	: $i_{DMC}, \overline{ref}, \bar{g}, \bar{g}_d, \bar{K}_{DMC}, \bar{\Delta U}$

Un criterio ampliamente extendido para seleccionar el tiempo de muestreo del controlador consiste en tomar un valor entre el 10% y el 25% del mínimo tiempo de respuesta deseado para el sistema en lazo cerrado [1]. En este caso, se ha seleccionado un tiempo de muestreo de 0,18 min. Para simplificar el proceso de diseño por parte del usuario, este valor se mantiene fijo en la GUI.

Una vez realizada la preparación del algoritmo DMC, cuando el usuario lo indique, se procederá a la simulación del sistema con los datos aportados. El cálculo de la señal de control en cada ciclo (Paso 11 del Algoritmo 1), viene representado por el siguiente algoritmo:

- Las entradas son: La variable controlada en el instante inicial y_0 y en el instante actual y_k , los vectores $\overline{ref}, \bar{g}, \bar{g}_d$ y $\bar{\Delta U}$, el último valor de la señal de control u_{k-1} , los horizontes de predicción p y control m , el número de elementos de la respuesta en escalón del sistema lineal aproximado del proceso N y de la dinámica de perturbación N_d , el tiempo equivalente al ciclo actual de simulación t , el tiempo de muestreo del controlador DMC, T_{DMC} , y el índice, i_{DMC} .
- Paso 1: Se obtiene el resto de la división entre t y T_{DMC} . Si es nulo, se continúa en el siguiente paso, en caso contrario, se finaliza el algoritmo.
- Paso 2: Se aumenta en una unidad el valor del índice i_{DMC} .
- Paso 3: Se obtiene la salida medida incremental, restando a la salida actual y_k el valor de la salida en el instante inicial y_0 .

- Paso 4: Se construye el vector de referencias futuras \bar{w} , a partir del vector \overline{ref} , del horizonte de predicción p y del índice i_{DMC} .
- Paso 5: Se calcula la respuesta libre mediante la ecuación (140).
- Paso 6: Se desplazan los elementos del vector $\overline{\Delta U}$ una posición a la izquierda.
- Paso 7: Se asigna al último elemento del vector $\overline{\Delta U}$ la siguiente acción de control, calculada mediante la ecuación (143).
- Paso 8: Se actualiza la señal de control sumándole el último elemento del vector $\overline{\Delta U}$.
- Las salidas son: La actualización del vector $\overline{\Delta U}$, de la señal de control u_k y del índice i_{DMC} .

La Tabla 6 (Algoritmo 3: Cálculo de la acción de control óptima mediante DMC) muestra de forma detallada el desarrollo matemático de este algoritmo.

Tabla 6. Algoritmo 3: Cálculo de la acción de control óptima mediante DMC

Entrada	: $\overline{ref} \in \mathbb{R}^n, \bar{K}_{DMC} \in \mathbb{R}^p, \bar{g} \in \mathbb{R}^{N+p_{m\acute{a}x}}, \bar{g}_d \in \mathbb{R}^{N_d+p_{m\acute{a}x}}, \overline{\Delta U} \in \mathbb{R}^{N+1}, u_{k-1} \in \mathbb{R}, y_0 \in \mathbb{R}, y_k \in \mathbb{R}, t \in \mathbb{R}^+, T_{DMC} \in \mathbb{R}^+, N \in \mathbb{N}, N_d \in \mathbb{N}, p \in \mathbb{N}, m \in \mathbb{N}, i_{DMC} \in \mathbb{N}$
Paso 1	: Si $(t \bmod T_{DMC}) = 0$ ir a Paso 2. En otro caso Fin
Paso 2	: $i_{DMC} = i_{DMC} + 1$
Paso 3	: $y_m = y_k - y_0$
Paso 4	: $\bar{w} \in \mathbb{R}^p, \bar{w} = [ref_{i_{DMC}+1} \quad \dots \quad ref_{i_{DMC}+p}]^T$
Paso 5	: $\bar{f} = [f(t+1) \quad \dots \quad f(t+p)]^T$
Paso 6	: $\overline{\Delta U}_i = \overline{\Delta U}_{i+1}, \forall i < N+1, i \in \mathbb{N}^+$
Paso 7	: $\overline{\Delta U}_{N+1} = K_{DMC} \cdot (w - f)$
Paso 8	: $u_k = u_{k-1} + \overline{\Delta U}_{N+1}$
Salida	: $\overline{\Delta U}, u_k, i_{DMC}$

Nótese que, para el correcto funcionamiento del algoritmo, T_{DMC} debe ser múltiplo de Δt_k , con el objetivo de obtener periódicamente un resto nulo en la división del paso 1.

En el caso de que el usuario realice una simulación con la opción “Simulación Dinámica” seleccionada, los elementos del vector de referencias futuras del algoritmo DMC serán todos del mismo valor que la consigna indicada por el usuario.

En la Figura 27 se muestran las respuestas temporales obtenidas tras realizar varias simulaciones del reactor encamisado controlado por un algoritmo DMC, con horizontes de predicción y control $p = 25$ y $m = 10$ y distintos valores del factor de ponderación de los esfuerzos de control λ . Para valores mayores de λ , se observa una menor agresividad en la respuesta transitoria a cambio de una reducción en el esfuerzo de control. Por otro lado, puede comprobarse la anticipación del controlador predictivo al seguimiento de referencias, obtenida mediante la resolución del problema de optimización utilizando un vector de referencias futuras. Nótese que el tiempo de muestreo considerado tiene efecto sobre la evolución del caudal de refrigeración, que está gobernado por la válvula de control, de tal modo que se aprecia un cierto rizado en las curvas.

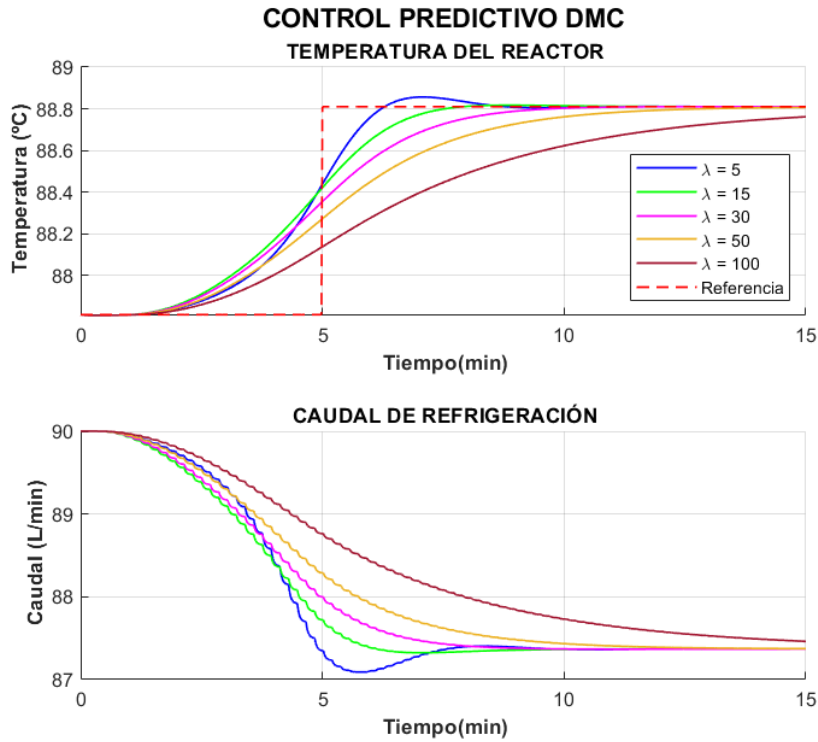


Figura 27. Desempeño de un controlador DMC para distintos valores de λ

La Figura 28 muestra una comparación entre las respuestas temporales obtenidas mediante distintas variaciones del algoritmo DMC aplicado al reactor encamisado, para una perturbación de 10°C sobre el valor nominal de la temperatura de refrigeración a la entrada en el instante $t = 2 \text{ min}$. Puede comprobarse que la adición de un modelo de perturbación mejora el comportamiento, y se obtiene un rechazo más eficaz y eficiente de la misma al utilizar el controlador en el lazo principal en una estructura de control en cascada.

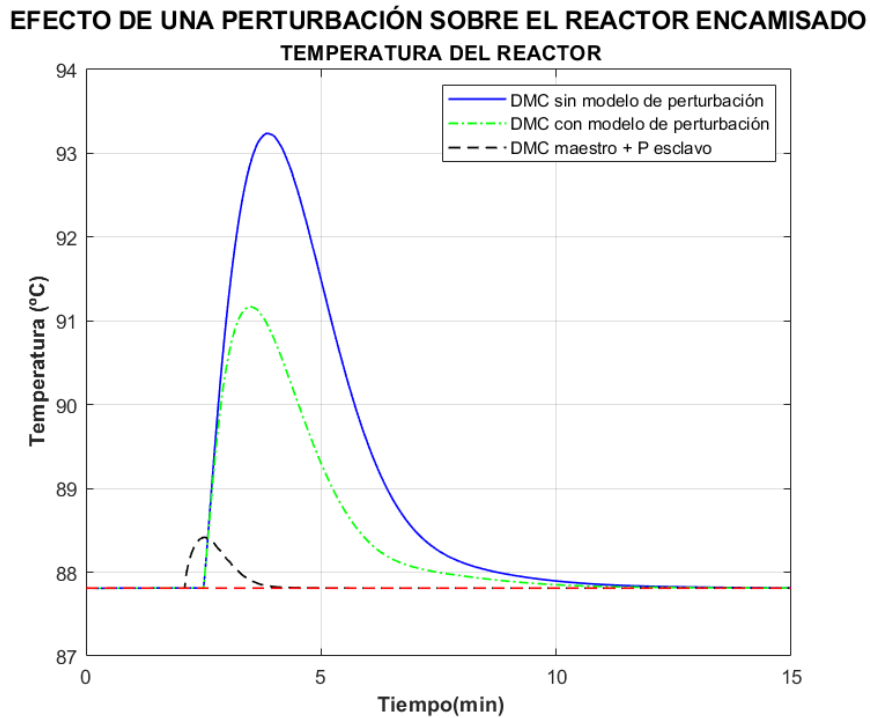


Figura 28. Comparación del rechazo de una perturbación para distintas variaciones del algoritmo DMC

En la Figura 29 se muestran los resultados de un experimento realizado sobre el reactor encamisado, en el que se introduce, en primer lugar, una perturbación de 10 °C respecto al valor nominal de T_{ce} en el instante $t = 2 \text{ min}$ y, posteriormente se aplica una referencia de 95 °C en $t = 10 \text{ min}$. Los controladores se han sintonizado para proporcionar una respuesta agresiva y obtener así un buen rechazo de la perturbación y un rápido seguimiento de la referencia. Puede observarse cómo en el control DMC la señal de control está optimizada para obtener un seguimiento de las referencias futuras, a la vez que se tienen en cuenta los esfuerzos de control, de acuerdo a la función objetivo considerada.

RESPUESTA DEL REACTOR ENCAMISADO MEDIANTE CONTROL PID Y DMC

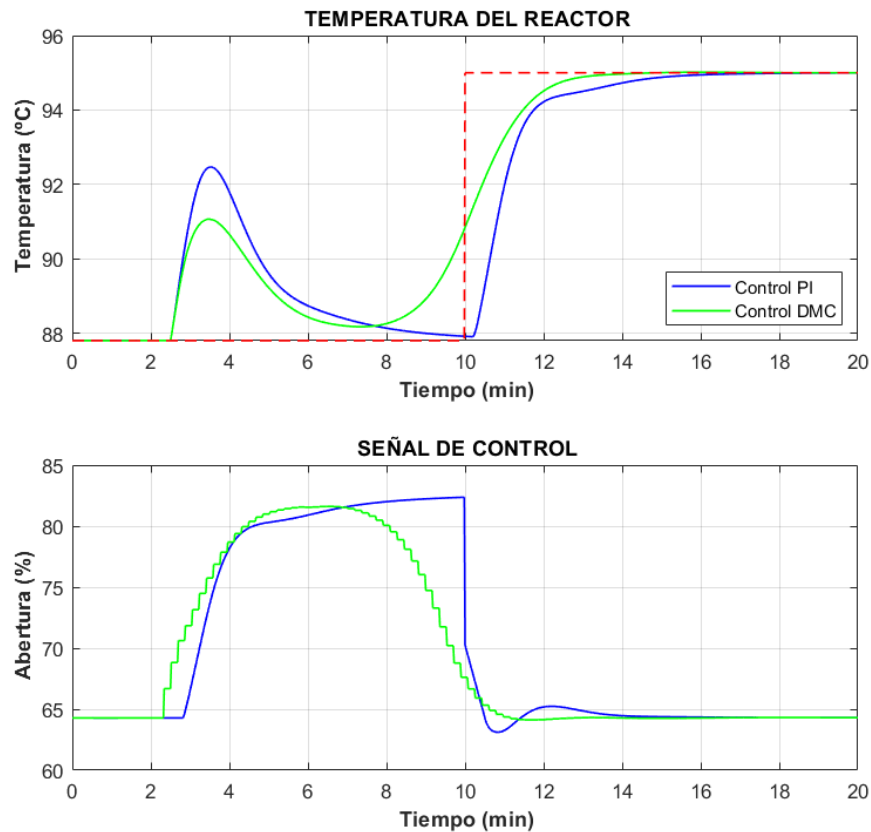


Figura 29. Comparación de la respuesta temporal obtenida mediante control PID y DMC

6 CONTROL ADAPTATIVO Y AUTOAJUSTABLE

Los controladores convencionales suelen diseñarse partiendo de un sistema lineal aproximado del proceso, con el objetivo de obtener un controlador que funcione en el entorno cercano a un punto de operación determinado. Esta aproximación introducirá una mayor desviación respecto al comportamiento deseado cuanto más se difiera de la dinámica real.

La agresividad, la robustez y, en general, la respuesta transitoria del sistema son aspectos que pueden verse afectados al cambiar de punto de operación, debido a las no linealidades presentes en la mayoría de procesos químicos, haciendo que el desempeño del control o regulación (rechazo de perturbaciones) deje de cumplir con las especificaciones para las cuáles se ha llevado a cabo el diseño. Esto también sucede cuando, debido a cambios físicos en la planta como el ensuciamiento o el desgaste de elementos, se producen variaciones en los parámetros del proceso, como puede ser el caso del coeficiente global de transferencia de calor. El control adaptativo y autoajustable permite dar solución a este problema, consiguiéndose el comportamiento deseado en un mayor rango de operación.

Por ello, se incluye una herramienta integrada en la GUI que consiste en un entorno de simulación con controladores adaptativos y autoajustables. Mediante todas las herramientas presentes en la aplicación y detalladas en capítulos anteriores, se facilita, en gran medida, el diseño de este tipo de controladores, permitiendo al usuario sintonizarlos sin necesidad de realizar cálculos.

En este capítulo se proporciona una breve descripción de los fundamentos teóricos de los algoritmos de control añadidos en esta herramienta, así como los criterios de diseño establecidos. En primer lugar, se trata la estrategia de control “*Gain Scheduling*” aplicado a un PI y a un DMC y, posteriormente, se describe el funcionamiento del control autoajustable mediante el método de mínimos cuadrados recursivos (RLS) y sintonización IMC.

6.1 Estrategia de control adaptativo mediante ajustable por tabla

Esta técnica, también conocida por su terminología inglesa “*Gain Scheduling*”, consiste en un control en el que los parámetros de un determinado controlador varían en función del punto de operación del proceso, de acuerdo a una tabla previamente establecida. Para ello, es necesario seleccionar la variable auxiliar que será utilizada como fuente de información para determinar este cambio en las condiciones y, posteriormente, sintonizar el controlador para cada caso considerado, de forma que se compensen las no linealidades del proceso.

La ventaja del ajuste por tabla, frente a otros métodos de control adaptativo, reside en que no es necesario hacer una estimación de parámetros, reduciendo el esfuerzo de control y permitiendo una rápida variación de los parámetros del controlador ante los cambios en el proceso [16]. Sin embargo, la conmutación puede implicar transitorios en la respuesta temporal del sistema, provocando efectos indeseados. Para evitarlo, puede utilizarse un filtro paso bajo con un tiempo de respuesta ajustado en función de cada proceso.

La Figura 30 muestra de forma esquematizada el funcionamiento de este tipo de algoritmo de control [16].

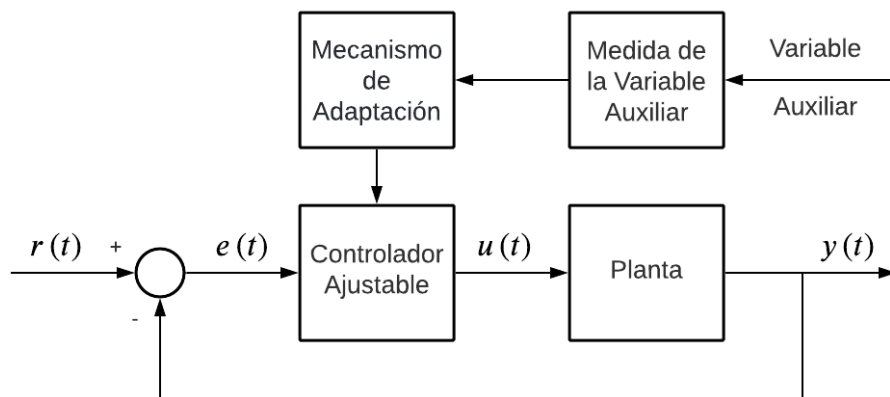


Figura 30. Diagrama de bloques de un control adaptativo mediante ajuste por tabla

La variable auxiliar, dependiendo del sistema, puede estar presente tanto en el medio ambiente (variables externas), como en el propio proceso. En procesos químicos, resulta conveniente utilizar variables internas (por ejemplo, la temperatura o la concentración) como entradas al mecanismo de adaptación.

En este caso, se ha empleado como variable auxiliar (Figura 30) la variable controlada y se ha obtenido un modelo lineal aproximado para diferentes puntos de operación mediante el algoritmo de identificación descrito en el capítulo 4. Además, para evitar el efecto indeseado de las conmutaciones en los parámetros de control debido a cambios pequeños en la variable auxiliar, se han establecido unos valores de tolerancia y se ha diseñado y ajustado de manera experimental un filtro paso bajo digital a la salida del algoritmo de selección de parámetros.

Tras analizar la respuesta del proceso en todo el rango de operación, se han observado tres zonas diferenciadas donde la dinámica es similar. La aproximación de tres modelos lineales para cada una de estas zonas da como resultado una buena representación del proceso real y éstos pueden ser utilizados para el diseño de tres controladores del mismo tipo, pero con diferentes parámetros. Las zonas establecidas son:

- Zona 1: de 105 a 114 °C
- Zona 2: de 85 a 105 °C
- Zona 3: de 78,5 a 85 °C

La Figura 31 representa de forma esquematizada el algoritmo de conmutación con tolerancias. Notar que estos valores no se aplican directamente, puesto que son introducidos a la entrada del filtro digital y pueden ser interpretados como los parámetros deseados para cada punto de operación.

De acuerdo a la Figura 31 se tiene:

- Controlador 1: Opera en zona 1
- Controlador 2: Opera en zona 2
- Controlador 3: Opera en zona 3

La conmutación entre controladores se produce de acuerdo a la siguiente lógica:

- Para conmutar a un controlador con un menor índice asignado, el valor de la variable auxiliar debe ser mayor o igual que el nivel superior de tolerancia (VLH o VHH)
- Para conmutar a un controlador con un mayor índice asignado, el valor de la variable auxiliar debe ser menor o igual que el nivel inferior de tolerancia (VLL o VHL)

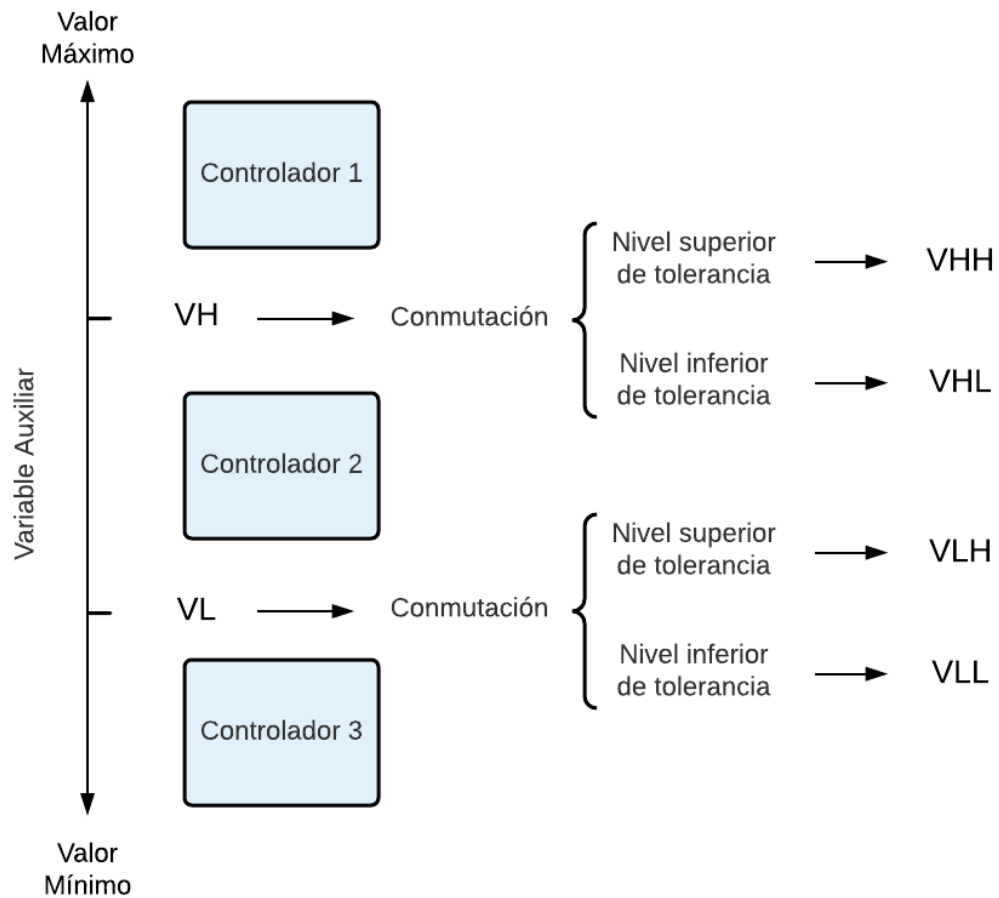


Figura 31. Diagrama de conmutación de la estrategia de control adaptativo de ajuste por tabla

Los valores de tolerancias deben ser lo suficientemente pequeños para que las zonas estén bien diferenciadas, pero lo suficientemente grandes para evitar conmutaciones indeseadas. Por ello, se ha considerado un único valor de 3°C. Los valores numéricos de conmutación establecidos para la interfaz gráfica se representan en la Tabla 7 en función del caso de control.

Tabla 7. Valores de conmutación del control adaptativo mediante ajuste por tabla

Control de temperatura	
VHH	108 °C
VH	105 °C
VHL	102 °C
VLH	88 °C
VL	85 °C
VLL	82 °C

Esta estrategia se ha aplicado al caso del reactor encamisado controlado mediante un PI y un DMC. Para el caso del control PI, puede seleccionarse en la ventana gráfica los valores de K_c y T_i para cada zona establecida. En el caso del DMC, el parámetro de adaptación se trata de la matriz dinámica y del factor de ponderación λ , el cual puede ser ajustado por el usuario para cada zona de operación. Los horizontes de predicción y de control seleccionados se mantendrán constantes a lo largo de la simulación, independientemente del punto de funcionamiento del reactor.

Para proporcionar al usuario una solución por defecto, se ha llevado a cabo una identificación paramétrica para cada una de las tres zonas establecidas. Con los parámetros de las funciones de transferencia obtenidas, se ha llevado a cabo un preajuste de los controladores PI mediante las reglas IMC descritas en el apartado 5.1.4. Posteriormente, se han ajustado los valores para obtener un equilibrio entre velocidad y suavidad en la respuesta transitoria. Por otro lado, se han incluido en la configuración de la aplicación los tres modelos lineales identificados, con el objeto de ser utilizados por el algoritmo DMC. Los valores del horizonte de predicción, horizonte de control y del factor de ponderación de los esfuerzos de control λ se han ajustado experimentalmente hasta obtener un buen desempeño.

La Figura 32 muestra una comparación entre los resultados obtenidos con controladores PI y DMC convencionales y los obtenidos con la estrategia de ajuste por tabla. El ensayo se ha realizado variando el valor de la referencia, de forma que, partiendo de la operación nominal, se conduce al proceso a la zona establecida como “baja temperatura” (zona 3). Posteriormente, se aplica una perturbación de -5°C en la temperatura de refrigeración de entrada respecto a su valor nominal en el instante $t = 60 \text{ min}$. Los controladores no adaptativos se han diseñado considerando un modelo lineal identificado en la zona de “alta temperatura” (zona 1). Puede comprobarse que se obtiene un mejor seguimiento de la referencia con la estrategia “Gain Scheduling”, al variar los parámetros para ajustarse al punto de operación comandado.

6.2 Controlador autoajustable basado en mínimos cuadrados recursivos

Los controladores autoajustables, también conocidos como *Self-Tuning Regulators* (STR), pueden ser sintonizados automáticamente en línea, a partir de datos obtenidos del proceso real y combinando técnicas de identificación con métodos analíticos de cálculo de controladores. Para ello, es necesario, en primer lugar, diseñar un controlador con parámetros conocidos que establezca el sistema para, posteriormente, sustituir estos valores por aquellos proporcionados por el algoritmo de autoajuste. Este método es especialmente útil en sistemas con dinámica no lineal, ya que introduce una estrategia de adaptación, actualizando los parámetros del controlador en función del punto de operación del proceso. También es muy conveniente para corregir las desviaciones debidas a cambios en los parámetros del modelo (desgaste, obstrucción, cambio en el nivel de altura de la mezcla, etc.). La estructura de control se ilustra de forma esquematizada en la Figura 33 [9].

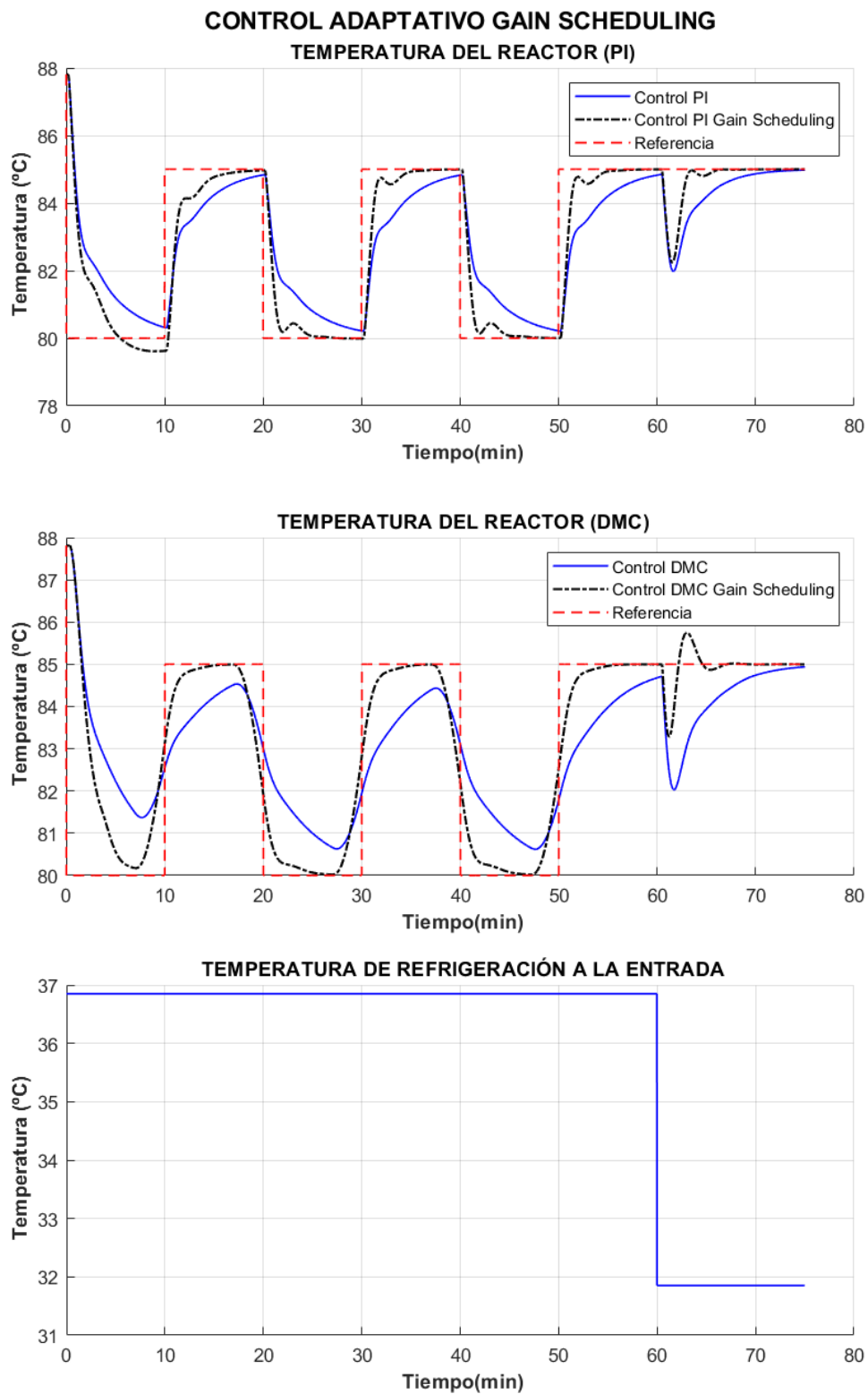


Figura 32. Comparación entre las respuestas temporales obtenidas para el caso del reactor CSTR encamisado, aplicando control PI y DMC con y sin estrategia de ajuste por tabla.

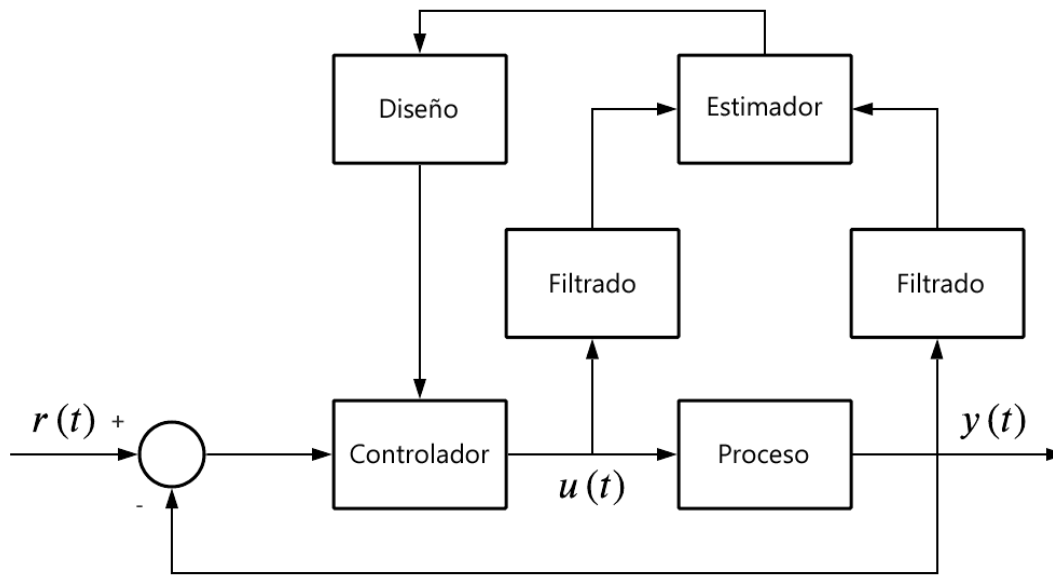


Figura 33. Diagrama de bloques de un controlador autoajustable

Con relación a la etapa de estimación, muchos algoritmos resultan inadecuados para su implementación en línea, como es el caso del método de mínimos cuadrados descrito en el apartado 4.2. Esto es debido a que dicho estimador necesita una base de datos del proceso obtenida de manera “*off-line*” para calcular un único vector paramétrico. Por ello, es preferible utilizar el método de mínimos cuadrados recursivos (RLS), mediante el cual se actualizan los parámetros en cada periodo de muestreo con el objetivo de minimizar el error de estimación. Este método puede describirse mediante las siguientes ecuaciones [9]:

$$\bar{\theta}(t) = \bar{\theta}(t-1) + \bar{\Gamma}(t)\bar{\varphi}(t)\epsilon(t) \quad (144)$$

$$\epsilon(t) = y(t) - \bar{\varphi}(t)^T \bar{\theta}(t-1) \quad (145)$$

$$\bar{\Gamma}(t) = \frac{1}{\lambda_o} \left(\bar{\Gamma}(t-1) - \frac{\bar{\Gamma}(t-1)\bar{\varphi}(t)\bar{\varphi}^T(t)\bar{\Gamma}(t-1)}{\lambda_o + \bar{\varphi}(t)^T \bar{\Gamma}(t-1)\bar{\varphi}(t)} \right) \quad (146)$$

donde $\bar{\theta}$ es el vector paramétrico que, al igual que en el algoritmo de mínimos cuadrados, contiene los parámetros a identificar, $\bar{\varphi}$ es el vector de regresión, que contiene los datos de entrada al algoritmo, $\bar{\Gamma}$ es la matriz de covarianza del error de estimación, y λ_o el factor de olvido, que modifica la actualización de la matriz de covarianza con el objetivo de dar un mayor peso a los datos recientes respecto a los antiguos.

Este método de estimación paramétrica no requiere de una gran base de datos, porque sólo es necesario conocer los valores en el instante actual e instante previo de las variables de estimación. Además, a diferencia del método de mínimos cuadrados, no se necesita invertir matrices, lo que reduce el esfuerzo computacional del algoritmo.

En caso de no disponer de información a priori de los parámetros, puede considerarse $\bar{\Gamma}_0 = \infty \bar{I}$ (idealmente). En el extremo opuesto, si el conocimiento a priori del vector paramétrico es perfecto, debe tomarse $\bar{\Gamma}_0 = \bar{0}$ (idealmente). Respecto al factor de olvido, éste puede variar entre 0 y 1 y su elección depende del comportamiento deseado: un valor cercano a 1 corresponde a una mayor robustez mientras que un valor cercano a 0 corresponde a una mayor velocidad de adaptación [9].

La estructura del regresor y del vector paramétrico dependen del tipo de modelo que se desea identificar. En este caso, al igual que en el algoritmo implementado en la herramienta de identificación y descrito en el capítulo 4, se aproxima a un sistema de primer orden, con lo que se tiene:

$$\bar{\varphi}_k = \begin{bmatrix} u_{k-1-n_{del}} \\ -y_{k-1} \end{bmatrix} \quad (147)$$

$$\bar{\theta} = \begin{bmatrix} a \\ b \end{bmatrix} \quad (148)$$

donde pueden deducirse los parámetros del sistema equivalente en tiempo continuo a partir del método de transformación descrito en el apartado 4.3.

Este procedimiento tiene la desventaja de que, al mantener un regresor nulo durante cierto tiempo, la matriz de covarianza crece exponencialmente. Para solucionar este problema, el algoritmo realiza exclusivamente la estimación paramétrica cuando hay suficiente excitación en la señal de control [9]. Para ello dispone de un sistema de detección de la excitación, que consiste en un filtrado de tipo paso alto de la referencia:

$$y_{hp}(k) = c_{hp}y_{hp}(k-1) + k_{hp}c_{hp}(r(k) - r(k-1)) \quad (149)$$

donde y_{hp} es la salida del filtro, y la señal de salida del proceso, c_{hp} es la constante de ajuste del filtro y k_{hp} es una ganancia para escalar la señal de salida.

El algoritmo de estimación puede diseñarse para ponerse en marcha cuando la señal de salida del filtro paso alto sea distinta de cero y detenerse en caso contrario. Sin embargo, es posible que pueda producirse un cambio en la referencia lo suficientemente pequeño como para no aportar excitación suficiente. Para evitar este problema, se han establecido unos determinados valores no nulos de salida del filtro para los cuales se activará o desactivará el algoritmo de estimación paramétrica. Estos valores se han determinado de manera experimental mediante la evaluación por simulación de la detección de excitación.

Por otro lado, la variación de los parámetros de control se llevará a cabo una vez que la salida del filtro paso alto tenga un valor cercano a cero y el proceso de estimación recursiva se haya detenido. Esta medida se realiza para evitar que esta variación se produzca durante la respuesta transitoria del proceso. Además, con esto también se evita la aplicación del ajuste automático del controlador a partir de los primeros datos de identificación, los cuales tienen asociados una mayor incertidumbre en el error de estimación.

En la Figura 34 se muestran los datos de una simulación del proceso controlado mediante un PI sin autoajuste, donde se aplica el algoritmo de identificación por mínimos cuadrados recursivos. La información a priori considerada es un vector paramétrico nulo (parámetros desconocidos) y una matriz de covarianza arbitrariamente grande. Los resultados se comparan con los parámetros obtenidos a partir de una identificación paramétrica fuera de línea, mediante el método de mínimos cuadrados, alrededor del mismo punto de operación. En la gráfica de la respuesta estimada, puede comprobarse la similitud entre la estimación (producto del regresor por el vector paramétrico calculado) y la respuesta del proceso simulado. Asimismo, también se representa la salida del filtro paso alto, que realiza la función de detección de la excitación, accionando e inhabilitando el algoritmo de identificación recursivo.

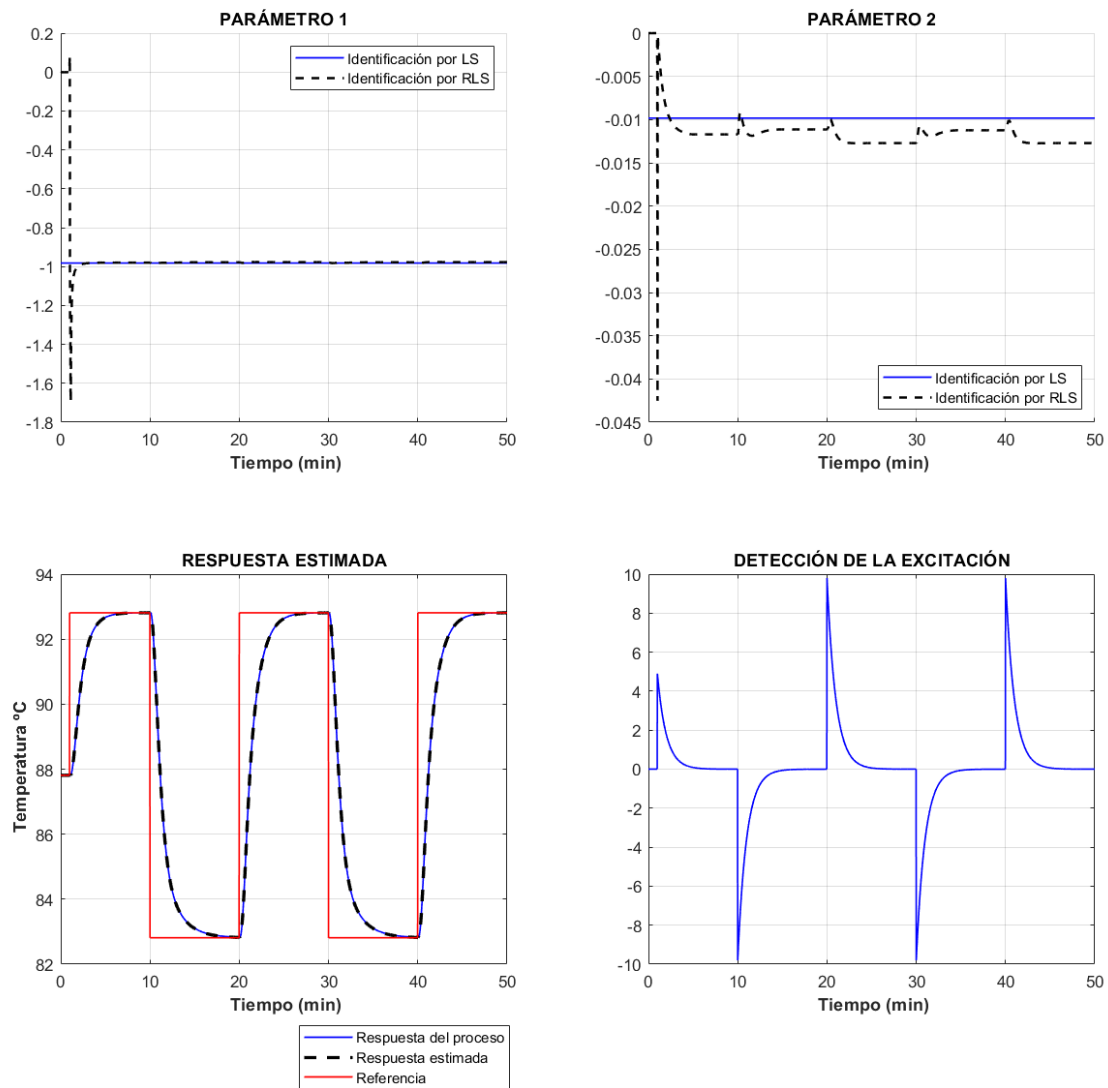


Figura 34. Resultados de una identificación paramétrica mediante mínimos cuadrados recursivos

En el Anexo B se detallan los datos de simulación, así como los resultados numéricos obtenidos en el experimento de identificación paramétrica recursiva.

Una vez realizado el proceso de identificación en un instante de tiempo determinado, deben utilizarse los parámetros obtenidos para realizar una sintonización automática del controlador. En este caso se ha considerado una estructura de control PI estándar y se ha empleado el método de sintonización IMC descrito en el apartado 5.1.4, en función de un valor de τ_{lc} proporcionado por el usuario. Los parámetros del controlador calculados no son actualizados de forma directa, ya que se hacen pasar previamente por un filtro paso bajo para evitar efectos indeseados en la respuesta transitoria del proceso. El usuario deberá aportar también los valores iniciales de la constante proporcional y tiempo integral con los que se pondrá en marcha la simulación.

A continuación, se detalla el algoritmo implementado en la interfaz gráfica para obtener en cada ciclo de simulación la actualización de los parámetros del controlador:

- Preparación fuera de línea:
 - Definir una matriz de covarianza inicial $\bar{\Gamma}$, con valores arbitrariamente altos y un vector paramétrico $\bar{\theta}$ inicial, preferiblemente obtenido a partir de una identificación con datos experimentales sobre el punto de operación nominal.

- Diseñar los parámetros del controlador PI y asignar a los parámetros deseados los valores: $K_{c,des} = K_c$ y $T_{i,des} = T_i$.
- Diseñar las constantes de los filtros paso bajo de la señal de entrada b_f , de la señal de salida a_f , de los parámetros del controlador p_{lp} , y las constantes del filtro paso alto c_{hp} y k_{hp} .
- Las entradas son: Los valores en el ciclo anterior de las variables $\bar{\Gamma}, \theta, u, u_{lp}, y, y_{hp}, y_{lp}$. Los valores en el ciclo actual de las variables r, u, y . Los parámetros $y_0, K_{c,des}, T_{i,des}, \tau_{lc}, l_1, l_2, \lambda_0$, el valor del indicador RLS_f y las constantes de los filtros $a_f, b_f, c_{hp}, k_{hp}, p_{lp}$.
- Paso 1: Se actualiza el filtro paso bajo de la señal de control y de la señal de salida del sistema.
- Paso 2: Se actualiza el regresor de acuerdo a la ecuación (147).
- Paso 3: Se actualiza el filtro paso alto de la referencia, de acuerdo a la ecuación (149).
- Paso 4: Si $RLS_f = 0$ y $|y_{hp_k}| \geq l_1$, entonces $RLS_f = 1$.
- Paso 5: Si $RLS_f = 1$ y $|y_{hp_k}| \leq l_2$, entonces $RLS_f = 0$.
- Paso 6: Si $RLS_f = 1$, entonces se actualiza la matriz $\bar{\Gamma}$, de acuerdo a la ecuación (146) y se continúa en el paso 7. En caso contrario, continuar en el paso 11.
- Paso 7: Se actualiza el error de modelado mediante la ecuación (145).
- Paso 8: Se actualiza el vector paramétrico mediante la ecuación (144).
- Paso 9: Se obtienen los parámetros del modelo equivalente en tiempo continuo.
- Paso 10: Se realiza una sintonización IMC, de acuerdo a las ecuaciones (106) y (107), actualizando el valor de los parámetros deseados para el controlador.
- Paso 11: Si $RLS_f = 0$, se actualiza el filtro paso bajo aplicado a los parámetros del controlador.
- Las salidas son: Las variables actualizadas $\bar{\Gamma}, \bar{\theta}, \bar{\varphi}, K_{c,des}, T_{i,des}, K_c, T_i, u_{lp}, y_{hp}, y_{lp}$ y RLS_f .

El desarrollo matemático de este algoritmo se detalla en la Tabla 8 (Algoritmo 4: Actualización de los parámetros de un PI autoajutable mediante RLS y sintonización IMC).

La Figura 35 muestra los resultados obtenidos tras realizar una simulación del reactor CSTR controlado mediante un PI autoajutable por RLS y sintonización por IMC. Para las condiciones iniciales, se han diseñado unos parámetros de control que proporcionan una respuesta temporal lenta en la salida del proceso y un valor para τ_{lc} de 1 minuto, con el objeto de aumentar la agresividad. La respuesta del proceso se hace variar entre 80 y 90 °C para activar el algoritmo de identificación recursiva, de forma que pueda calcularse la actualización de los parámetros de control. En el instante $t = 60 \text{ min}$, se aplica una perturbación de $-5 \text{ }^\circ\text{C}$ respecto al valor nominal de la temperatura de entrada de refrigeración. Puede comprobarse que, a medida que avanza la simulación, los valores de K_c y T_i evolucionan para ajustarse a la especificación de diseño, τ_{lc} .

Tabla 8. Algoritmo 4: Actualización de los parámetros de un PI autoajustable mediante RLS y sintonización IMC

Entrada	: $\bar{\Gamma}_{k-1} \in \mathcal{M}_{n \times m}(\mathbb{R}), \bar{\theta}_{k-1} \in \mathbb{R}^2, \mathbf{r}_k \in \mathbb{R}, \mathbf{r}_{k-1} \in \mathbb{R}, \mathbf{u}_k \in \mathbb{R}, \mathbf{u}_{k-1} \in \mathbb{R}, \mathbf{u}_{lp_{k-1}} \in \mathbb{R}, \mathbf{y}_0 \in \mathbb{R}, \mathbf{y}_k \in \mathbb{R}, \mathbf{y}_{k-1} \in \mathbb{R}, \mathbf{y}_{hp_{k-1}} \in \mathbb{R}, \mathbf{y}_{lp_{k-1}} \in \mathbb{R}, \mathbf{K}_{c,des} \in \mathbb{R}, \tau_{lc} \in \mathbb{R}, \mathbf{a}_f \in \mathbb{R}^+, \mathbf{b}_f \in \mathbb{R}^+, \mathbf{c}_{hp} \in \mathbb{R}^+, \mathbf{k}_{hp} \in \mathbb{R}^+, \mathbf{p}_{lp} \in \mathbb{R}^+, \mathbf{l}_1 \in \mathbb{R}^+, \mathbf{l}_2 \in \mathbb{R}^+, \mathbf{T}_{i,des} \in \mathbb{R}^+, \lambda_o \in \mathbb{R}^+ : \lambda_o \in \mathbb{Z}^+ : \lambda_o \in [0, 1], \mathbf{RLS}_f \in \mathbb{Z}^+ : \mathbf{RLS}_f \in [0, 1]$
Paso 1	: Filtrado de la señal de entrada: $u_{lp_k} = b_f u_{lp_{k-1}} + (1 - b_f) \cdot (u_k - u_0)$. Filtrado de la señal de salida: $y_{lp_k} = a_f y_{lp_{k-1}} + (1 - a_f) \cdot (y_k - y_0)$.
Paso 2	: Actualización del regresor: $\varphi_1 = -y_{lp_{k-1}}, \varphi_2 = u_{lp_{k-1-n_{del}}}$
Paso 3	: Filtro paso alto: $y_{hp_k} = c_{hp} y_{hp_{k-1}} + c_{hp} k_{hp} (r_k - r_{k-1})$
Paso 4	: Si $\mathbf{RLS}_f = 0$ y $ y_{hp_k} \geq l_1$, entonces $\mathbf{RLS}_f = 1$.
Paso 5	: Si $\mathbf{RLS}_f = 1$ y $ y_{hp_k} \leq l_2$, entonces $\mathbf{RLS}_f = 0$
Paso 6	: Si $\mathbf{RLS}_f = 1$, entonces $\bar{\Gamma}_k = \left(\frac{1}{\lambda_o}\right) \left(\bar{\Gamma}_{k-1} - \frac{\bar{\Gamma}_{k-1} \bar{\varphi} \bar{\varphi}^T \bar{\Gamma}_{k-1}}{\lambda_o + \bar{\varphi}^T \bar{\Gamma}_{k-1} \bar{\varphi}}\right)$ y continuar en paso 7. En caso contrario, continuar en paso 11
Paso 7	: $\epsilon_k = y_{lp_k} - \bar{\varphi}^T \bar{\theta}_{k-1}$
Paso 8	: $\bar{\theta}_k = \bar{\theta}_{k-1} + \bar{\Gamma}_k \bar{\varphi} \epsilon_k$
Paso 9	: $K_c = \frac{\theta_2}{1 + \theta_1}, \tau = \frac{\Delta t_k}{ -\ln(-\theta_1) }, \tau_{mp} = n_{del} \cdot \Delta t_k$
Paso 10	: $K_{c,des} = \frac{\tau_p}{K_p(\tau_{lc} + \tau_{mp})}, T_{i,des} = \tau_p$
Paso 11	: Si $\mathbf{RLS}_f = 0, K_c = p_{lp} K_c + (1 - p_{lp}) K_{c,des}, T_i = p_{lp} T_i + (1 - p_{lp}) T_{i,des}$
Salida	: $\bar{\Gamma}_k, \bar{\theta}_k, \bar{\varphi}, \mathbf{K}_{c,des}, \mathbf{T}_{i,des}, \mathbf{k}_c, \mathbf{T}_i, \mathbf{u}_{lp_k}, \mathbf{y}_{hp_k}, \mathbf{y}_{lp_k}, \mathbf{RLS}_f$

En la aplicación, se ha configurado un vector paramétrico inicial, el cual se ha obtenido a partir de una identificación fuera de línea con el algoritmo de mínimos cuadrados, realizando un experimento alrededor del punto de operación nominal. La matriz de covarianza inicial, así como las constantes de los filtros también han sido diseñadas experimentalmente y no son modificables por el usuario.

Por otro lado, para proporcionar una solución por defecto, se han diseñado todos los parámetros de ajuste. En primer lugar, se ha sintonizado un controlador PI mediante las reglas basadas en IMC descritas en el apartado 5.1.4, partiendo del modelo obtenido con una identificación paramétrica por mínimos cuadrados. La constante del sistema en lazo cerrado puede variar en función de la agresividad deseada. En este caso, se ha considerado un valor de 1 *min*, que da lugar a una respuesta moderada-agresiva. Respecto al factor de olvido, se ha priorizado la robustez frente a la velocidad de estimación, por lo que se ha considerado un valor de 0,995.

En la aplicación, el usuario puede configurar los parámetros del controlador PI inicial, el tiempo deseado para el sistema en lazo cerrado y el factor de olvido.

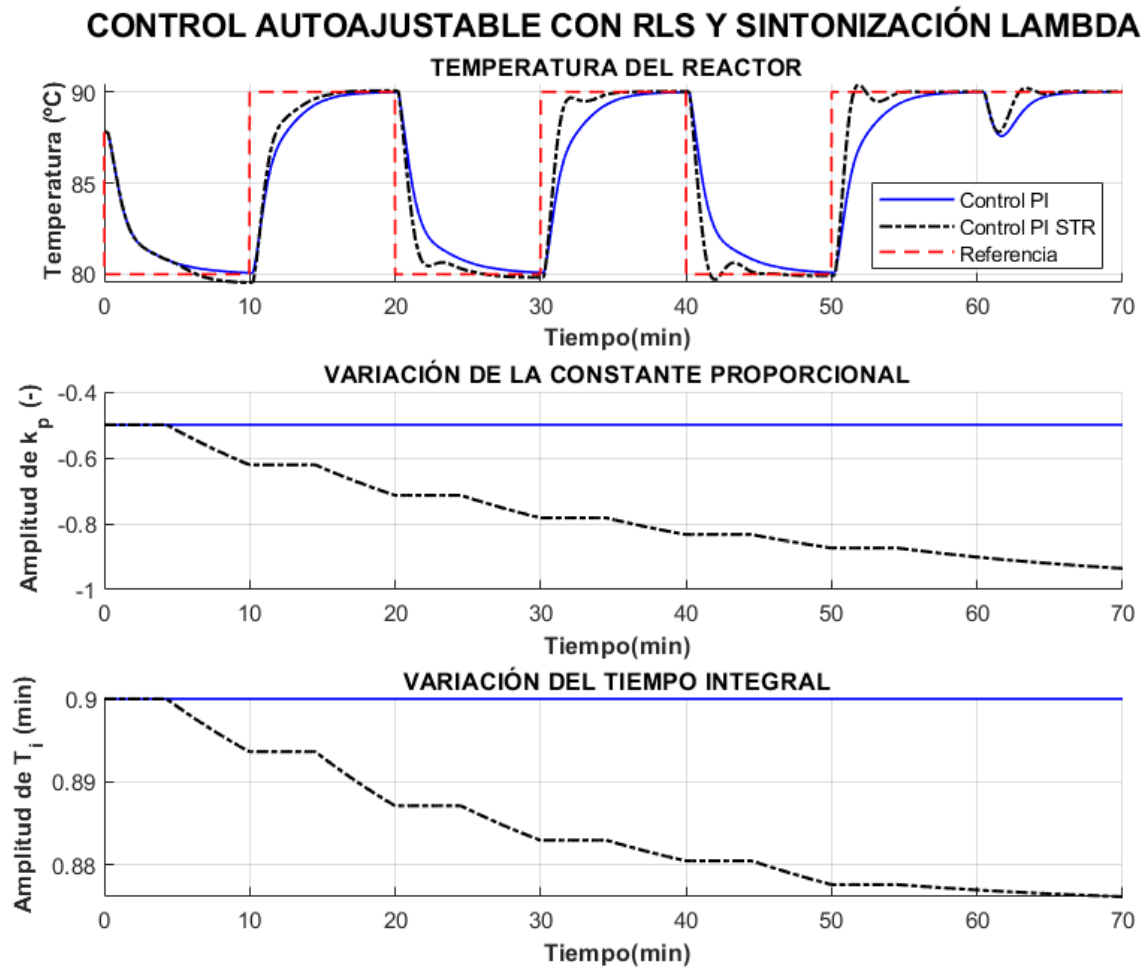


Figura 35. Respuesta temporal y evolución de los parámetros k_c y T_i para un control PI autoajustable, aplicado al caso del reactor CSTR encamisado

7 MEJORAS FUTURAS

En este capítulo se discuten los aspectos que han quedado fuera del alcance de este trabajo, y que contribuirían a mejorar la aplicación diseñada, proporcionando nuevas herramientas de análisis y diferentes posibilidades en la configuración de la simulación y del lazo de control. Estas mejoras podrán ser incluidas en el desarrollo de una nueva versión de la interfaz gráfica, manteniendo, además, las funcionalidades ya existentes y expuestas en los capítulos anteriores.

7.1 Opciones de simulación

En lo referente a las opciones de simulación, la adición de modelos de reactor de mayor complejidad permitiría la inclusión de herramientas dedicadas a un campo más avanzado de la teoría del control automático: el control multivariable. Diversos algoritmos de control podrían ser programados para facilitar al usuario el diseño de los parámetros y la observación y comparación de las respuestas obtenidas en cada caso.

Otra posible mejora consiste en el escalado de las dimensiones de los reactores a valores industriales, de forma que se obtengan casos más realistas donde, además, el tiempo muerto pueda ser mayor que los valores obtenidos en este TFM. Para ello, deben modificarse los parámetros de los modelos matemáticos implementados en la aplicación, así como las unidades y límites en los campos editables de las ventanas gráficas afectados por dicha modificación. Además, debe ajustarse de nuevo cada controlador para proporcionar un valor por defecto al usuario.

Para facilitar al usuario la comparación entre las diferentes respuestas temporales obtenidas con los distintos controladores, se propone introducir una opción que permita la representación, en una misma gráfica, de varias simulaciones consecutivas, mostrando, a su vez, una leyenda donde se indique a qué simulación corresponde cada curva.

Respecto a la simulación dinámica, se podría incluir un panel donde se programará con antelación un conjunto de datos de entrada, definidos en distintos instantes de tiempo. De esta forma, los cambios prefijados se aplicarían automáticamente a medida que avanza la simulación, proporcionando una mayor precisión y comodidad en el ensayo. El algoritmo de control DMC, en este caso, se ejecutaría recibiendo como argumento de entrada el vector de referencias futuras, lo que permitiría una anticipación óptima de los cambios en los valores de consigna.

7.2 Identificación

Para mejorar la herramienta de identificación, se propone añadir la posibilidad de seleccionar entre las distintas estructuras de modelos descritas en el apartado 4.1, así como implementar una opción que permita seleccionar dos conjuntos de datos: uno para identificación y otro para validación. Debido a esta nueva variedad de opciones, resultaría conveniente realizar un análisis estadístico basado en los residuos, mostrando al usuario en una gráfica la autocorrelación de los residuos y permitiéndole comparar los resultados con los de otro experimento diferente.

Por otro lado, el algoritmo de identificación puede mejorarse considerando el número de pasos de retraso como otro elemento del vector paramétrico, lo que conlleva a modificar el problema de optimización por uno de mayor complejidad.

7.3 Opciones de control

En relación a las opciones de control, se propone la adición de variaciones en el control PID, como es el caso de la estructura en paralelo u otras variantes comerciales del algoritmo, para lo cual, debería llevarse a cabo un diseño de los parámetros con la finalidad de proporcionar al usuario unos valores por defecto. Asimismo, podrían facilitarse diferentes métodos de sintonización y permitir la modificación de las reglas para los niveles de agresividad establecidos.

En caso de escalar las dimensiones del reactor y obtener un modelo con un retardo de tiempo dominante ($\tau_{mp} \gg \tau_p$), resultaría conveniente añadir una estructura de predictor de Smith, con parámetros ajustables por el usuario.

Por otro lado, también resultaría conveniente la posibilidad de extender la herramienta de control adaptativo y autoajutable al caso del reactor con serpiente.

7.4 Visualización

Para mejorar la flexibilidad de las ventanas gráficas, se propone habilitar la opción de modificar el tamaño de las mismas, de forma que puedan expandirse o contraerse para obtener cualquier relación de aspecto, así como poder adaptarse a distintas resoluciones de pantalla. Para ello, debería diseñarse un algoritmo que calcule el tamaño inicial de los elementos y figuras presentes en cada ventana, el tamaño de fuente inicial y establecer sus posiciones relativas iniciales. Asimismo, al ejecutar la aplicación, debería determinarse la resolución del monitor y del valor del zoom seleccionado en la configuración de pantalla, de forma que se establezca una posición, tamaño y relación de aspecto inicial para todas las ventanas que componen la GUI.

Por otro lado, sería necesario diseñar otro algoritmo que modifique el tamaño de los elementos, posiciones relativas y tamaño de fuente en función de la medida en píxeles del largo y ancho de cada ventana en cada instante de tiempo. Para no comprometer la claridad visual con los cambios realizados en las dimensiones, sería conveniente diseñar distintas distribuciones para los elementos de cada ventana, de manera que se adapten a distintas relaciones de aspecto, siendo de especial interés las siguientes:

- Relación 1:1
- Relación 3:2
- Relación 4:3
- Relación 16:9

Actualmente, la GUI es compatible con las siguientes resoluciones:

- 1920x1080 (con 100% y 125% de zoom)
- 1400x1050 (con 100% y 125% de zoom)
- 1680x1050 (con 100% y 125% de zoom)
- 1280x1024
- 1280x960
- 1600x900
- 1440x900

REFERENCIAS

- [1] Mathworks, «es.mathworks.com,» [En línea]. Available: <https://es.mathworks.com>. [Último acceso: 2022].
- [2] E. Peña, A. Pérez, A. J. Miranda y J. Sánchez, «Modelado de un reactor químico tipo CSTR y evaluación del control predictivo aplicando Matlab-Simulink,» *Revista INGENIERÍA UC*, vol. 15, nº 3, pp. 97-112, 2008.
- [3] G. L. Foutch y A. H. Johannes, «Reactors in Process Engineering,» de *Encyclopedia of Physical Science and Technology*, Academic Press, 2003, pp. 23-43.
- [4] K.-Y. Show y D.-J. Lee, «Bioreactor and Bioprocess Design,» de *Biohydrogen*, Elsevier, 2013, pp. 317-337.
- [5] D. E. Seborg, T. F. Edgar, D. A. Mellichamp y F. J. Doyle III, *Process Dynamics and Control*, 4th Edition, John Wiley & Sons Ltd., 2016.
- [6] S. Bittanti, *Model Identification and Data Analysis*, John Wiley & Sons, 2019.
- [7] Y. Okuyama, *Discrete Control Systems*, Springer Science+Business Media, 2014.
- [8] K. Ogata, *Ingeniería de Control Moderna*, 5ª Edición, Pearson Education, 2010.
- [9] K. Johan Åström y T. Hägglund, *Control PID Avanzado*, Pearson Education, 2009.
- [10] K. J. Åström y B. Wittenmark, *Computer-Controlled Systems: Theory and Design*, Prentice Hall, 1997.
- [11] C. A. Smith y A. B. Corripio, *Principles and Practice of Automatic Process Control*, John Wiley and Sons, 1997.
- [12] S. Skogestad, «Tuning for Smooth PID Control with Acceptable Disturbance Rejection,» *Industrial & Engineering Chemistry Research*, vol. 45, pp. 7817-7822, 2006.
- [13] A. O. Ahmed, G. A. Gasmelseed, A. B. Karama y A. E. Musa, «Cascade Control of a Continuous Stirred Tank Reactor (CSTR),» *Journal of Applied and Industrial Sciences*, 2013.
- [14] A. J. Shaiju y I. R. Petersen, «Formulas for Discrete Time LQR, LQG, LEQG and Minimax LQG Optimal Control Problems,» *IFAC-PapersOnLine*, vol. 41, pp. 8773-8778, 2008.
- [15] E. Fernández Camacho y C. Bordons, *Model Predictive Control*, Springer Science+Business Media, 2007.
- [16] F. Rodríguez Rubio y M. J. López Sánchez, *Control Adaptativo y Robusto*, Editorial Universidad de Sevilla, 1996.

ANEXO A. MANUAL DE LA INTERFAZ GRÁFICA DE USUARIO

1. Instalación

En primer lugar, ejecute el instalador haciendo doble click sobre él. Se mostrará la siguiente ventana (Figura A1):

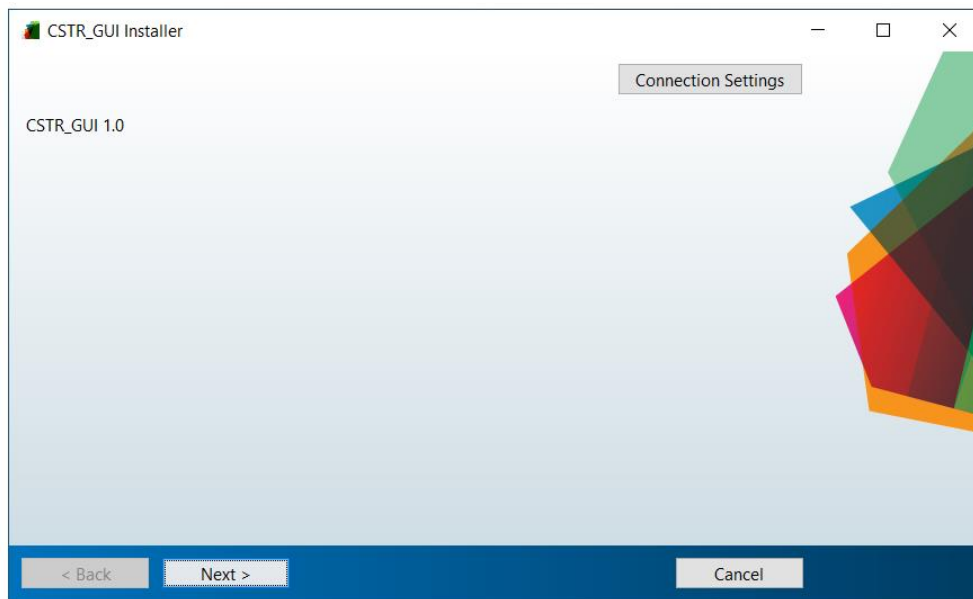


Figura A1. Instalación del software. Inicio

En el apartado de “*Connection Settings*” pueden introducirse los datos de conexión necesarios en caso de estar utilizando un servidor proxy. Si su conexión a internet no requiere de este tipo de servidores deberá dejar en blanco los campos editables.

Haga click en el botón “*Next*” para continuar con el siguiente paso. Se mostrará una nueva ventana donde deberá indicar la ruta de instalación de la interfaz gráfica, así como seleccionar si se desea o no añadir un acceso directo en el escritorio (Figura A2). Con el botón “*Restore Default Folder*”, se selecciona de nuevo la ruta de instalación por defecto.

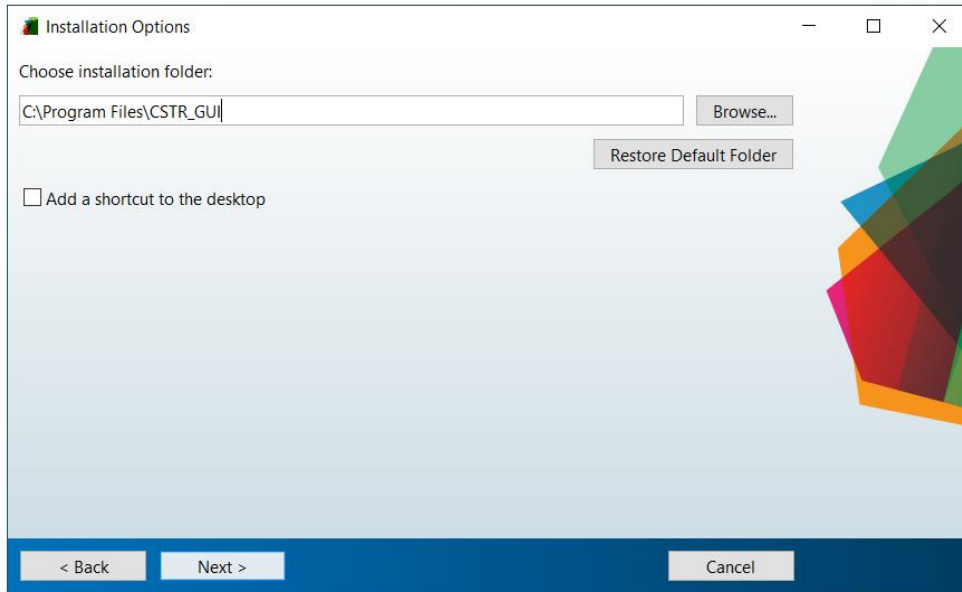


Figura A2. Instalación del software. Selección de ruta de instalación

Una vez indicada la ruta, vuelva a hacer click en el botón “*Next*”. El paquete de instalación incluye el software “*Matlab Runtime*”. Si no lo tiene instalado en su equipo se mostrará una ventana donde deberá indicar la ruta de instalación del mismo (Figura A3). Posteriormente, deberá aceptar los términos y condiciones de uso.

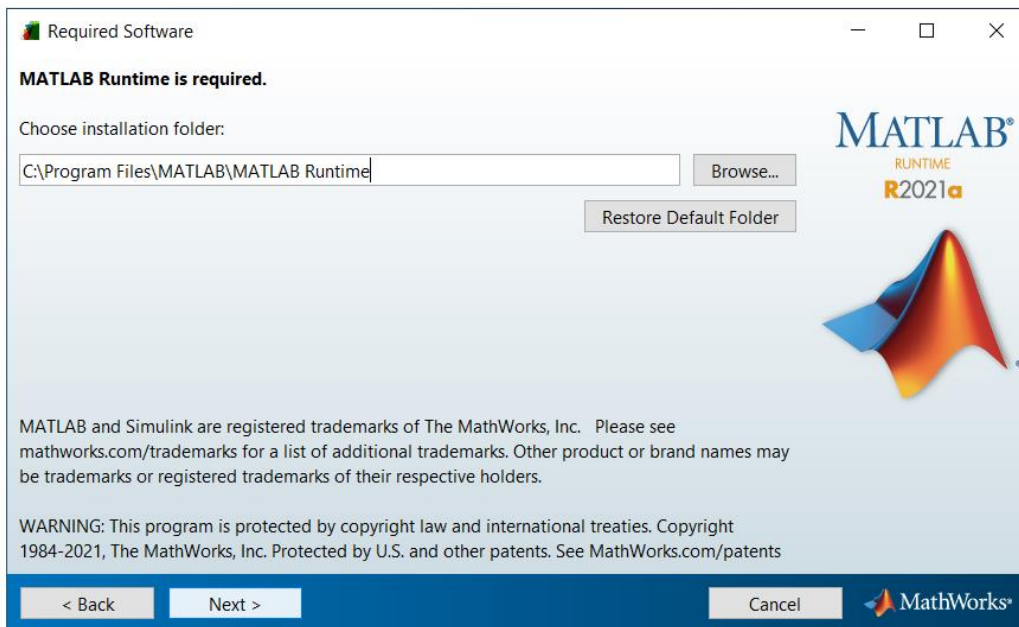


Figura A3. Instalación del software Matlab Runtime

Una vez configurada la instalación, se mostrará una ventana de confirmación, donde se dispone de un resumen de las opciones seleccionadas, así como información sobre el tamaño de la descarga (Figura A4). Haciendo click en el botón “*Install*” comenzará la instalación.

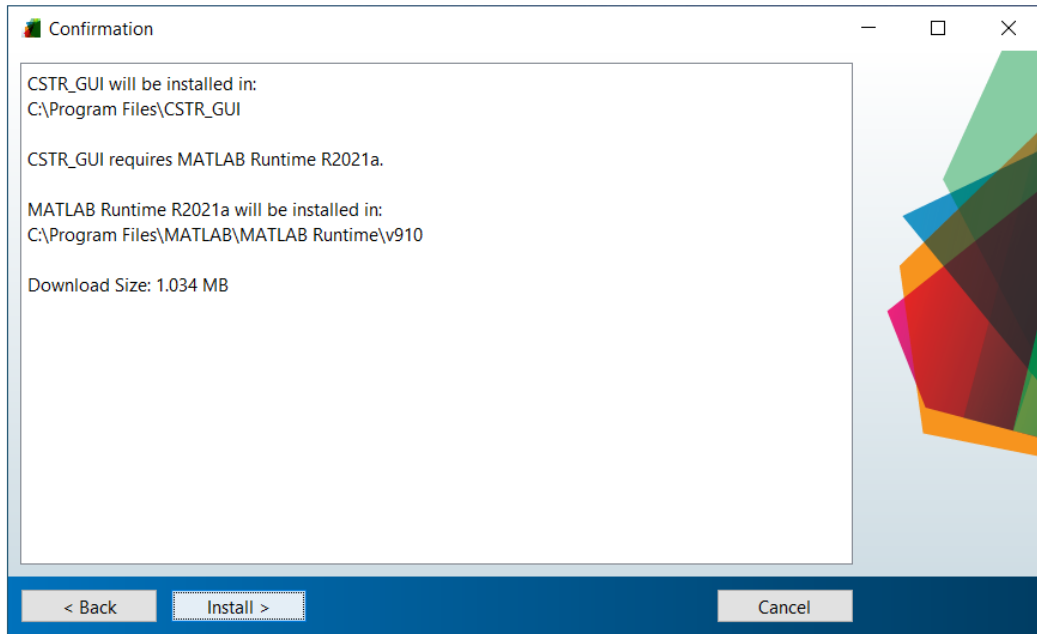


Figura A4. Instalación del software. Resumen de la instalación

2. Inicio

Al iniciar la aplicación, se mostrará una ventana donde deberá seleccionar el caso de estudio que desea simular (Figura A5):

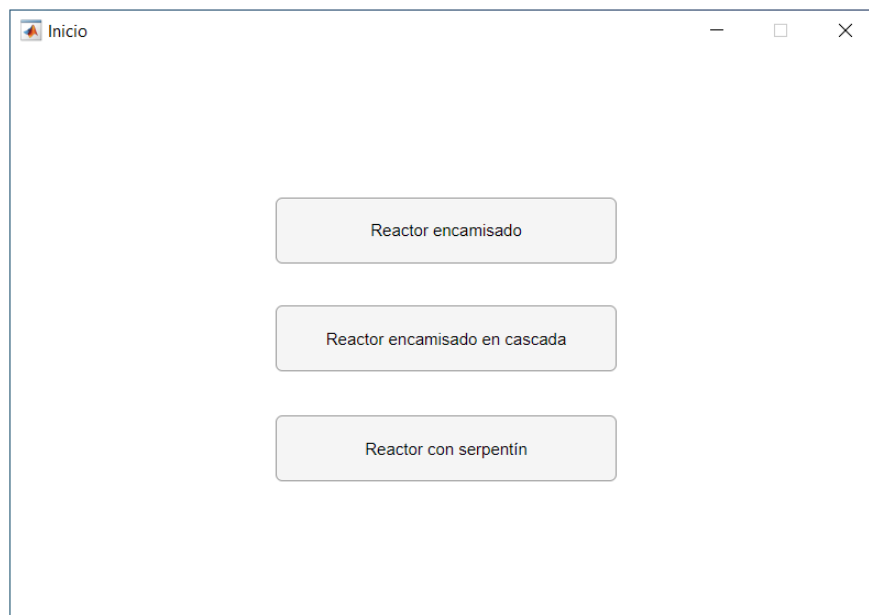


Figura A5. Ventana de inicio de la interfaz gráfica de usuario

Haciendo click sobre una de las cuatro opciones, se cargará la ventana principal (Figura A6) con la configuración correspondiente.

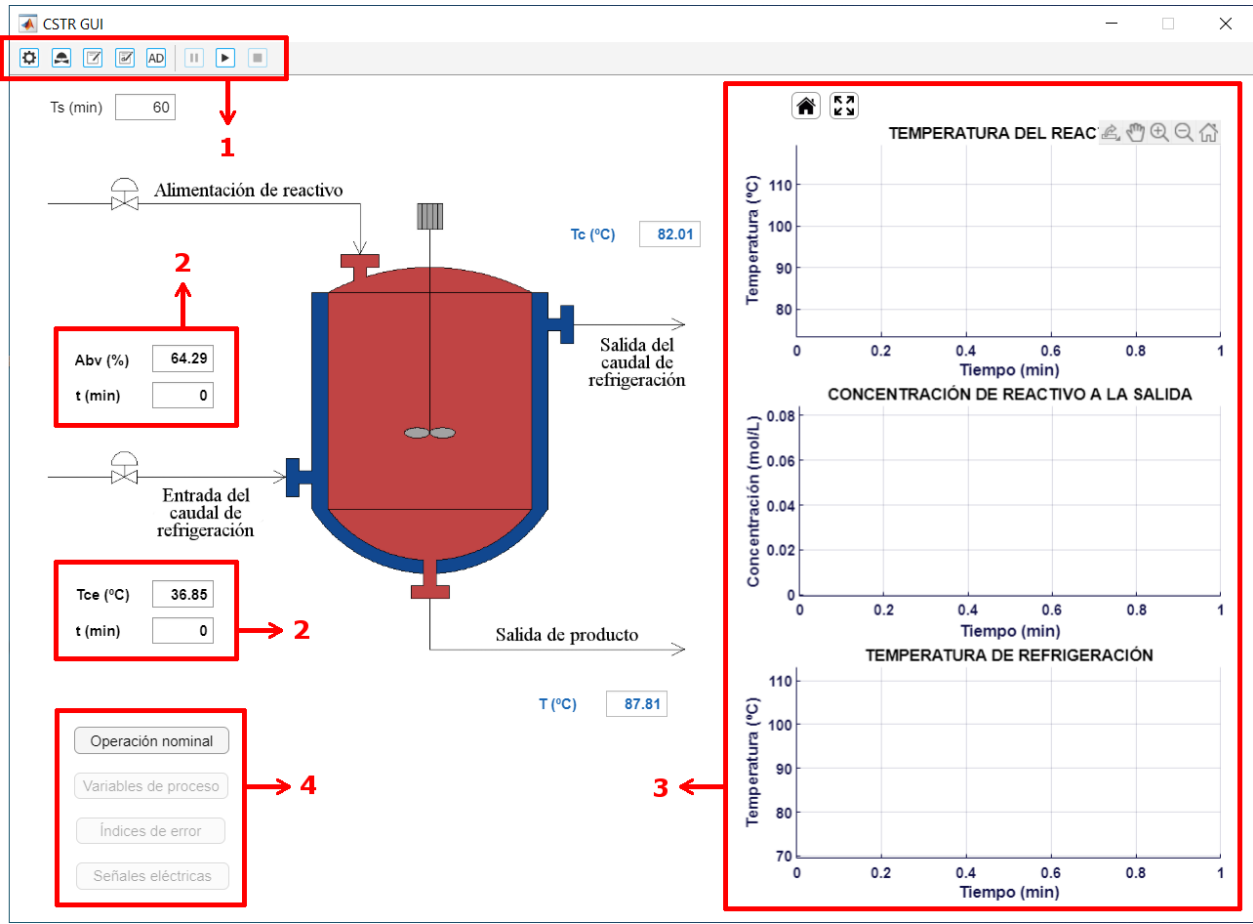


Figura A6. Ventana principal de la interfaz gráfica de usuario

En la barra superior (1) se tiene acceso a las opciones de simulación, opciones de control, ventana de ensayos experimentales y ventana dedicada al control adaptativo. Al lado, se dispone de tres botones destinados a la simulación del proceso. Para obtener información sobre los diferentes elementos de la interfaz gráfica, sitúe el cursor sobre cualquiera de ellos y se mostrará en un cuadro una breve descripción del objeto seleccionado.

Alrededor de la figura del reactor, se disponen de varios campos de datos (2), correspondientes a los valores de entrada a la simulación (color negro) y valores de salida (color azul). En función del tipo de control seleccionado, la GUI determina qué campos son editables, existiendo las siguientes posibilidades:

- Modo manual: Se permite modificar la abertura de la válvula de refrigeración y el valor de la temperatura de refrigeración a la entrada.
- Control regulatorio básico: Se permite modificar la referencia de temperatura del reactor y el valor de la temperatura de refrigeración a la entrada.
- Control en cascada: Se permite modificar la referencia de temperatura del reactor, el valor de la temperatura de refrigeración a la entrada y el valor de la temperatura de entrada al reactor.
- Control en cascada con el lazo maestro en modo manual: Se permite modificar la referencia de la temperatura de refrigeración y el valor de la temperatura de refrigeración a la entrada.

En la zona de la derecha (3), se muestran las gráficas donde se representará la evolución temporal de las variables de estado obtenidas tras la simulación del proceso. Estas variables son la temperatura del reactor, la concentración de reactivo a la salida y la temperatura de refrigeración a la salida de la camisa.

En la parte inferior se dispone de cuatro botones (4), cuyas funciones se detallan a continuación:

- Operación nominal: Devuelve los cuadros editables a sus valores por defecto, que corresponden a la operación nominal del reactor.
- Variables de proceso: Muestra la evolución temporal del estado interno del proceso, de la temperatura de refrigeración de entrada, de la potencia calorífica disipada, de la velocidad de reacción, del caudal de refrigeración y del porcentaje de abertura de la válvula de refrigeración (Figura A7).

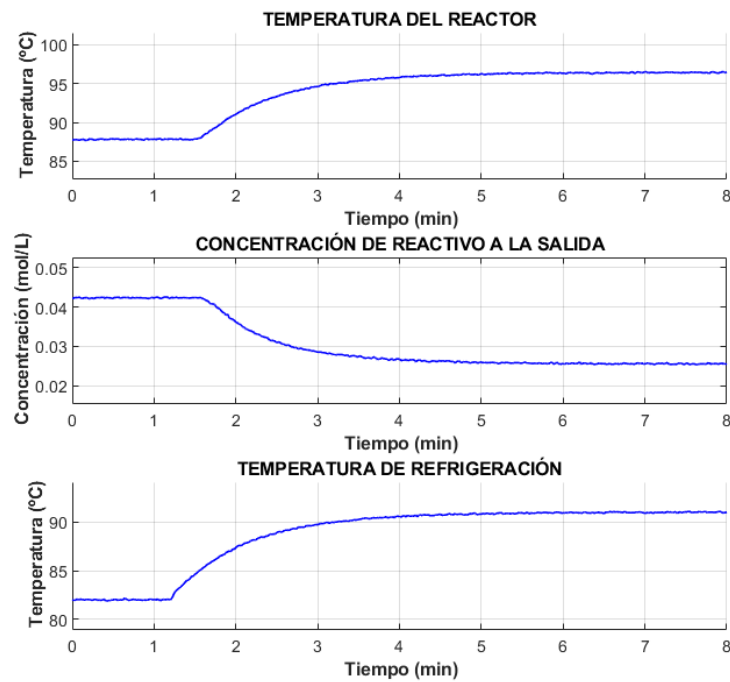


Figura A7. Evolución del estado interno del proceso obtenido mediante el botón “Variables de proceso”

- Índices de error: Muestra la evolución temporal del error y de la integral del error (valor en cada instante de la integral del error aproximada por Euler) (Figura A8).

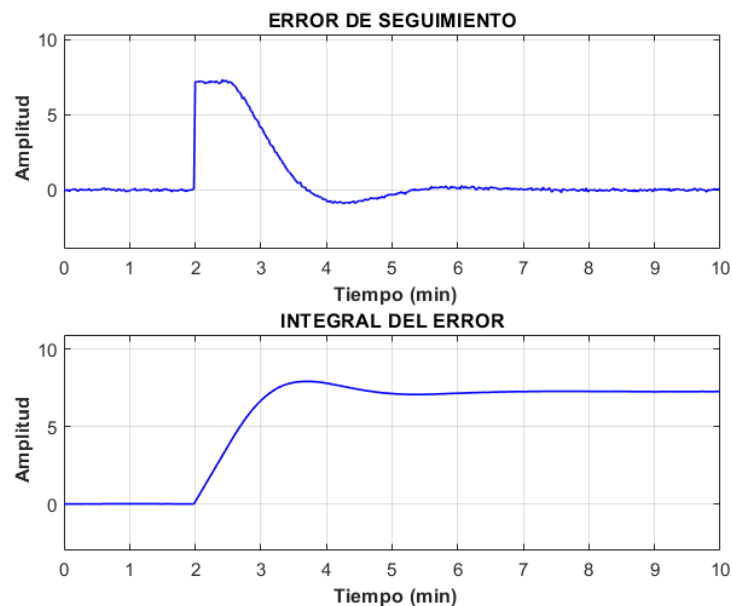


Figura A8. Evolución del error y de la integral del error obtenidos mediante el botón “Índices de error”

- Señales eléctricas: Muestra la evolución temporal de las variables correspondientes a las señales eléctricas (salida de sensores y salida del controlador) (Figura A9).

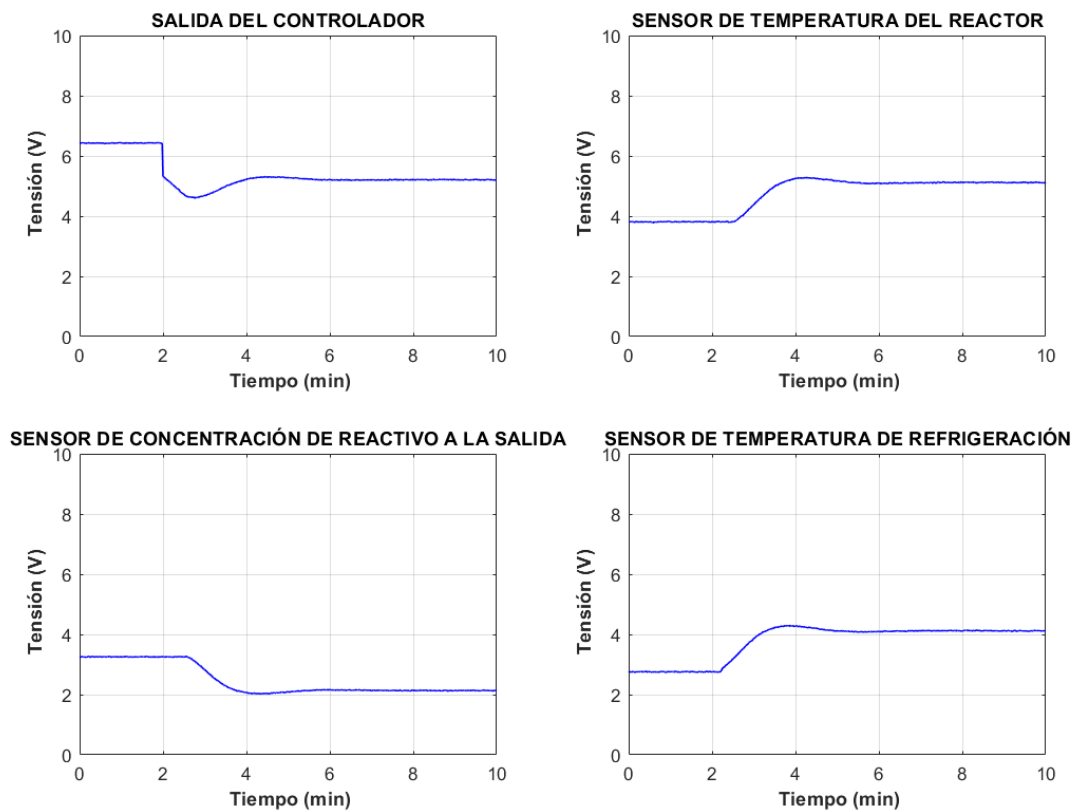


Figura A9. Evolución de las señales eléctricas obtenidas mediante el botón “Señales eléctricas”

En la Tabla A1 se muestran los rangos de medición de los sensores-transmisores.

Tabla A1. Rango de medición de los sensores-transmisores

Variable	Límite inferior	Límite superior
Temperatura del reactor	66,85 (°C)	121,85 (°C)
Temperatura de refrigeración	66,85 (°C)	121,85 (°C)
Concentración de reactivo a la salida	0 mol/l	0,13 mol/l
Temperatura de refrigeración a la entrada	16,85 (°C)	61,85 (°C)

3. Configuración de la simulación

Para abrir el menú de configuración, haga click sobre el botón “Configuración de Simulación” en la barra superior. Se mostrará la siguiente ventana (Figura A10):

Figura A10. Ventana gráfica de configuración de la simulación

En el primer desplegable se puede cambiar el caso de estudio. Hay un total de 3 casos para simular:

- Reactor CSTR refrigerado mediante serpentín
- Reactor CSTR encamisado
- Reactor CSTR encamisado con control en cascada

En el segundo desplegable se permite cambiar el método de integración numérica. Las opciones son:

- Método de Euler
- Método de Runge-Kutta de orden 4

Con el desplegable “Tipo de simulación” se pueden seleccionar dos opciones:

- Simulación estática: Se simula el sistema a lo largo del tiempo establecido en la casilla “Tiempo de simulación”. Los resultados se muestran en las gráficas tan pronto como finalice el bucle de simulación.
- Simulación dinámica: Cada ciclo de simulación y representación de datos se realiza con un tiempo de espaciamiento. Permite la simulación en tiempo real o acelerado en función del multiplicador indicado en la casilla de “Factor de tiempo”.

Los dos últimos desplegables corresponden, respectivamente, a la magnitud y rango de la señal de salida de los sensores y de la señal de salida del controlador. Las posibilidades son:

- De 0 a 10 V
- De 4 a 20 mA

En el panel inferior es posible modificar la desviación típica del ruido de medida simulado para la temperatura del reactor y de entrada al reactor, concentración de reactivo a la salida y temperatura de refrigeración a la entrada y a la salida de la camisa. Esta desviación típica hace referencia a las magnitudes de las señales de transmisión eléctricas (V o mA) y no a las magnitudes físicas ($^{\circ}\text{C}$ y mol/L).

A la derecha, se dispone de tres campos editables donde es posible configurar con un valor numérico las siguientes opciones:

- **Tiempo de simulación (min):** Horizonte de tiempo en el que se simulan los datos en el algoritmo de simulación estática. El valor indicado debe estar entre 1 min y 200 min.
- **Factor de tiempo:** Relación entre el tiempo simulado y el tiempo real. Determina la velocidad de cálculo del simulador. El valor indicado debe estar entre 1 y 120.

Al establecer un factor de tiempo de 1, los ciclos del bucle de simulación se ejecutan con un periodo igual al tiempo de muestreo, que es de 1,2 s (simulación en tiempo real). Un factor de tiempo de 120 implica reducir el periodo a 0,01 s.

- **Horizonte de visualización de datos (min):** Horizonte de tiempo representado en las gráficas en el algoritmo de simulación dinámica. El valor indicado debe estar entre 1 min y 60 min.
- **Tamaño del historial (min):** Tamaño del historial donde se registran los datos en el algoritmo de simulación dinámica. El valor indicado debe estar entre 10 min y 120 min.

El botón “Restaurar TF” devolverá a sus valores por defecto los parámetros de las funciones de transferencia guardadas en memoria.

Para restablecer las opciones por defecto, haga click en el botón “Restablecer” y, posteriormente, acepte los cambios (esto no afectará a los modelos lineales guardados en memoria).

4. Simulación del proceso

4.1. Simulación I. Algoritmo de simulación estática

Para realizar una simulación, en primer lugar, consigne los valores de entrada deseados en los campos editables, así como los instantes en los que desea aplicarlos y haga click en el botón de simulación, situado en la barra superior. Por defecto, estos campos contienen los valores correspondientes al punto de equilibrio desde el cual partirá la simulación. Posteriormente, se mostrará en las gráficas la evolución de los estados y se habilitarán los botones “Variables de proceso”, “Índices de error” y “Señales eléctricas”.

Sobre las gráficas se dispone de dos botones, “Restaurar Vista” y “Ajustar Vista” (Figura A11), que pueden utilizarse para acercar o alejar el zoom automáticamente.

Al situar el ratón sobre las gráficas, se despliega un menú en la parte superior (Figura A11) desde donde se permite exportarlas a formatos .png, .jpg, .tif y .pdf. También se permiten las opciones de: copiar como imagen, copiar como gráfico vectorial, desplazamiento, zoom y restaurar vista.

Para limpiar las gráficas haga click en el botón “Reset” situado en la barra superior. Esta acción dejará preparada la interfaz para realizar una nueva simulación.

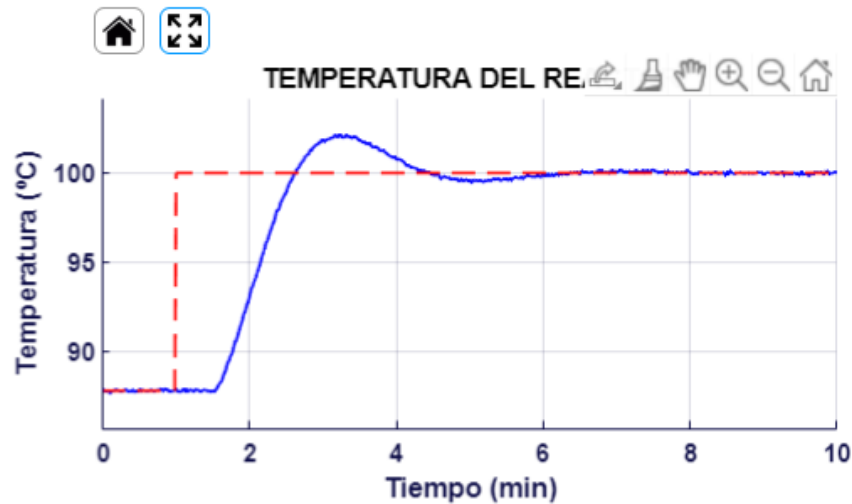


Figura A11. Opciones de gráficas

4.2. Simulación II. Algoritmo de simulación dinámica

En caso de haber seleccionado el modo de simulación dinámica, los cuadros editables de tiempo de la ventana principal quedarán inhabilitados. Se mostrarán dos casillas en la parte superior donde es posible modificar el factor de tiempo (FT) y el horizonte de visualización de datos (HVD) (véase apartado 3). Es posible seleccionar cualquier valor comprendido entre los siguientes límites:

- Rango disponible para el factor de tiempo: $FT = [1 ; 120]$ (un valor de $FT = 1$ corresponde a una simulación en tiempo real).
- Rango disponible para el horizonte de visualización de datos: $HVD = [1 ; 60] \text{ min}$

Para llevar a cabo una simulación, deben introducirse los datos de entrada y, posteriormente, hacer click en el botón “Simulación”. Los campos editables se bloquearán y comenzarán a mostrarse los resultados en las gráficas. En este modo, se habilita un botón denominado “Autoajustar Vista”, que permite hacer un ajuste automático de los ejes y actualizarlo en cada ciclo de representación de datos (Figura A12).

Para pausar temporalmente la simulación, haga click sobre el botón de “Pausa”. Las gráficas dejarán de mostrar resultados y se habilitarán de nuevo los campos editables. A partir de este momento, es posible modificar los datos de entrada que se tendrán en cuenta una vez se reanude el algoritmo. Al hacer click en “Simular”, continuará la simulación desde el punto anterior, con los nuevos datos actualizados que se hayan introducido durante la pausa. Para detener la simulación y resetear las gráficas haga click en el botón “Reset”.

En este modo de simulación es posible realizar un guardado de datos en la memoria de la aplicación, con la finalidad de llevar a cabo una identificación paramétrica. Esta opción está disponible para las siguientes configuraciones:

- Cualquier caso de estudio con un controlador configurado en manual.
- Caso de estudio del reactor encamisado en cascada con el lazo interno controlado y el externo en modo manual.

Para ello, se dispondrá de dos nuevos botones en la barra superior (Figura A12), de forma que, haciendo click en el botón “Grabar Ensayo” (representado por un círculo rojo), comenzarán a registrarse los resultados de la simulación. Durante la recopilación de datos, es posible realizar cambios en las condiciones de entrada a la simulación mediante el botón “Pausa” y “Simulación” y se puede cancelar la operación con el botón “Reset”, lo que producirá el borrado de todos los datos recopilados durante el proceso. Para finalizar exitosamente el guardado de datos se tienen las siguientes opciones:

- Hacer click en el botón “Finalizar Grabación” (representado con un cuadrado rojo).
- Si se alcanza un tiempo de simulación de 120 minutos, la aplicación finalizará automáticamente la grabación.
- Si se introduce una séptima entrada en escalón, la aplicación finalizará automáticamente la grabación (la última entrada no se registrará).

Con cualquiera de estas opciones, siempre que se hayan recopilado suficientes datos, se mostrará un mensaje confirmando que la operación se ha realizado con éxito. Los datos se almacenan temporalmente en la aplicación, por lo que se eliminarán al cerrar la ventana principal (para exportar los datos véase apartado 5.2).

Una vez realizado el proceso, en la ventana de “Ensayos” se habilitará un nuevo botón que permite cargar los datos guardados por la aplicación.

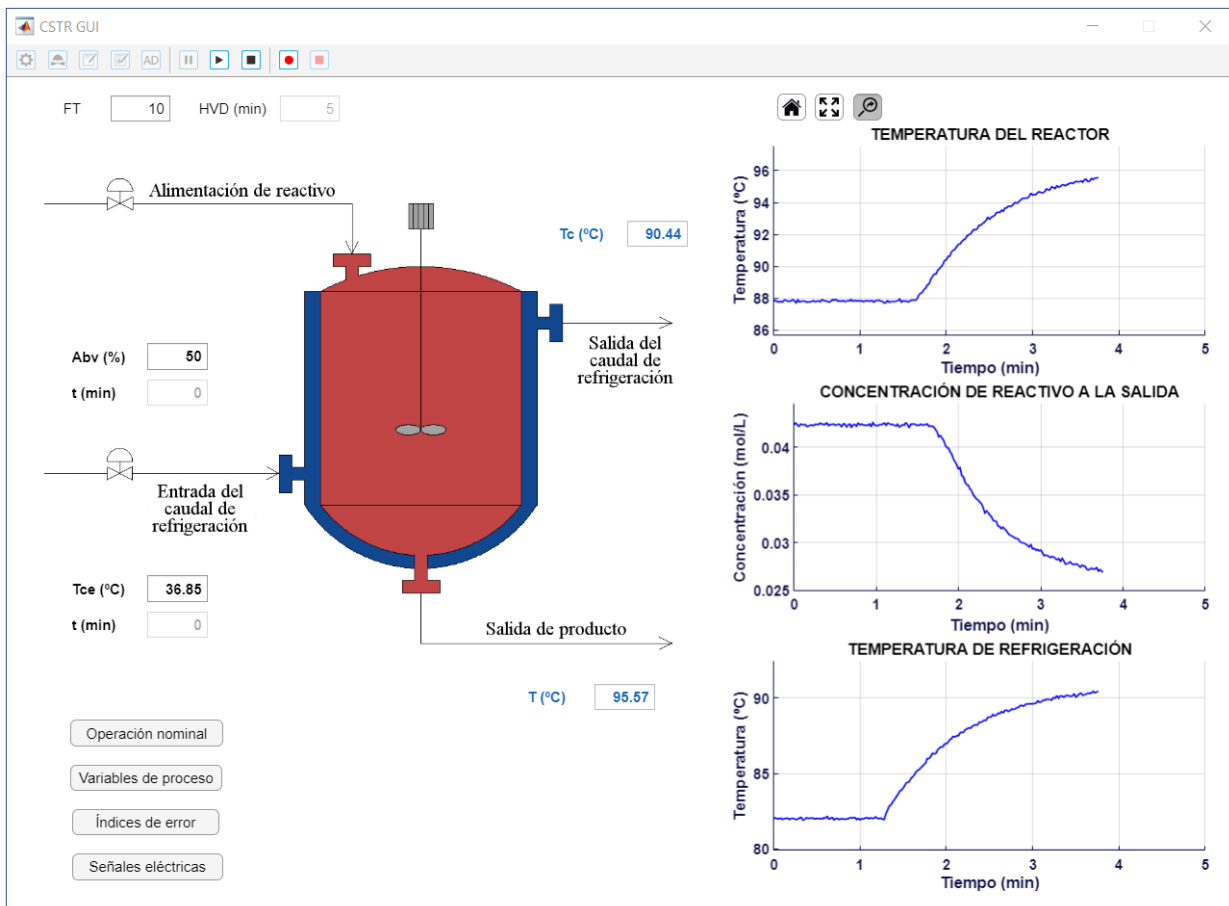


Figura A12. Guardado de datos en el modo de simulación dinámica

5. Realización de ensayos con entrada en escalón para identificación paramétrica

Para realizar ensayos con entrada en escalón, se dispone de dos opciones:

- Ensayos para identificación usando la variable manipulada.
- Ensayos para identificación usando la variable de perturbación.

Ambas opciones son accesibles desde la barra superior mediante sus botones correspondientes. La ventana gráfica de la herramienta es la siguiente (Figura A13):

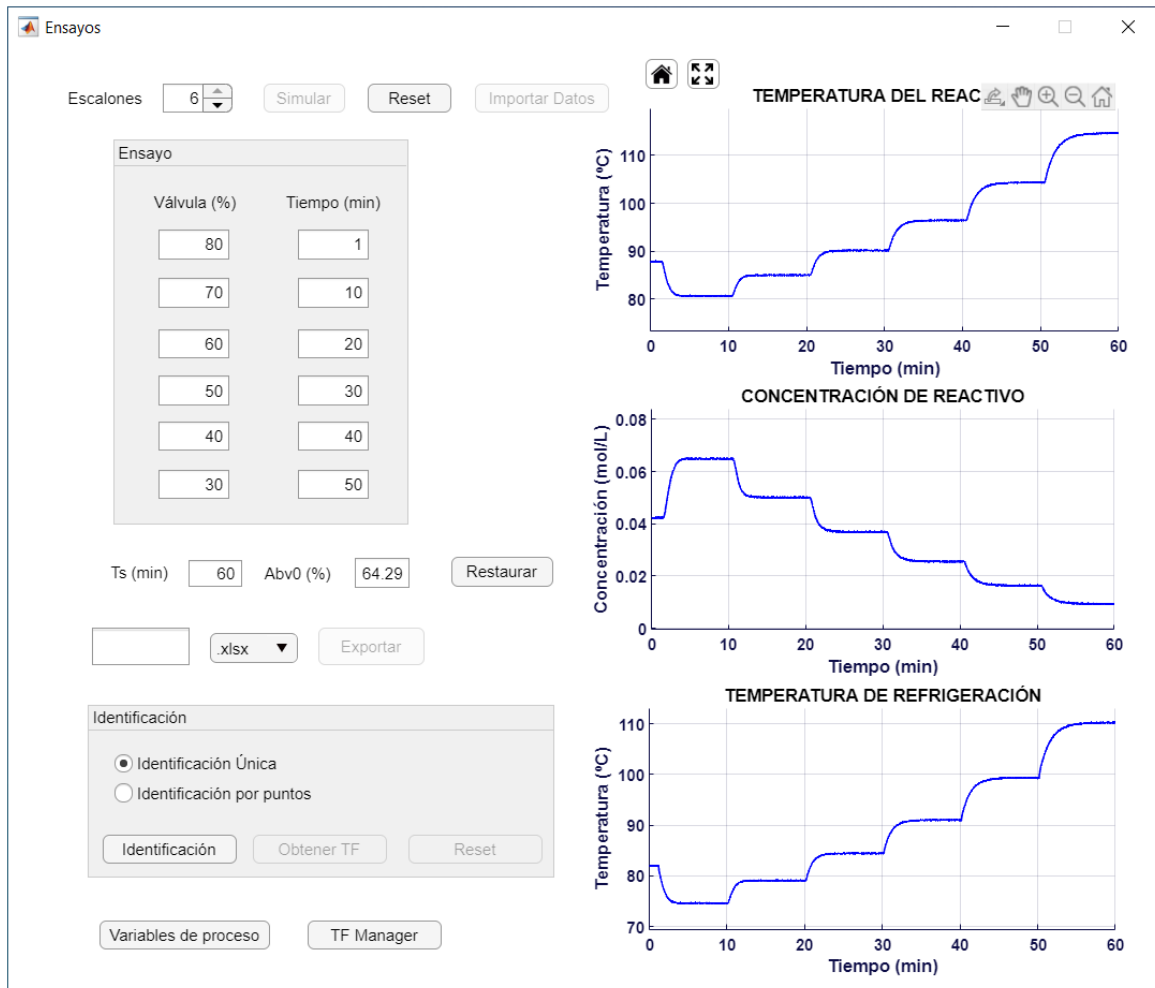


Figura A13. Ventana gráfica de identificación paramétrica

El experimento realizado puede variar en función del tipo de caso de estudio y del lazo de control establecido. Dependiendo de la configuración de control, la variable de entrada puede ser:

- La abertura de la válvula de la línea de refrigeración, en caso de haber seleccionado inicialmente la herramienta de ensayos con la variable manipulada.
- La referencia de temperatura de refrigeración, en caso de haber seleccionado inicialmente la herramienta de ensayos con la variable manipulada y tener configurado un control en cascada con el lazo interno cerrado.
- La temperatura de refrigeración de entrada, en caso de haber seleccionado inicialmente la herramienta de ensayos con la variable de perturbación.
- La temperatura de entrada al reactor, en caso de haber seleccionado inicialmente la herramienta de ensayos con la variable de perturbación y tener configurado un control en cascada con el lazo interno cerrado.

Asimismo, la variable indicadora de punto de funcionamiento puede ser:

- La abertura inicial de la válvula de refrigeración, en caso de control regulatorio básico y en caso de control en cascada en modo manual.
- La referencia inicial de temperatura de refrigeración a la salida de la camisa, en caso de control en cascada con el lazo interno cerrado.

5.1. Obtención de datos para identificación

En el panel superior puede seleccionarse el número de entradas que se aplicarán en la simulación. A medida que se vayan agregando, aparecerán en el panel nuevos campos editables para rellenarlos con los datos del ensayo.

Se dispone de dos columnas de datos de entrada, siendo la primera la correspondiente a los valores de la variable de entrada y la segunda, los instantes en los que se van a aplicar. Debajo del panel, se muestran dos campos editables en los que puede especificarse el tiempo de simulación (T_s) y un valor inicial para la variable indicadora de punto inicial de funcionamiento. Una vez configuradas las condiciones del ensayo, haga click en el botón simular. Los resultados se mostrarán en las gráficas de la derecha.

Otra forma de obtener datos para la identificación consiste en realizar una grabación desde la ventana principal, activando el modo de simulación dinámica. Para ello, seleccione el valor inicial de la válvula de refrigeración, haga click en el botón “Guardar datos” localizado en la barra superior de la ventana principal e inicie una simulación (véase apartado 4.2). Los datos registrados previamente a la introducción del primer escalón no se utilizarán en el algoritmo de identificación. De esta manera, es posible seleccionar el punto de equilibrio en torno al cual se desea llevar a cabo el experimento.

En caso de haber datos registrados en memoria, al abrir la ventana de “Ensayos” se mostrará el botón “Cargar Datos” y, haciendo click sobre él, se representarán en las gráficas los resultados recopilados durante la grabación. Los datos se cargan desde la memoria interna de la aplicación y pueden ser exportados a una hoja de cálculo o a un documento de texto (ver apartado 5.2).

Asimismo, si se dispone de un archivo con datos guardados, también es posible cargarlos en la herramienta mediante el botón “Importar Datos” (véase Figura A13). Para ello, debe seleccionar el tipo de extensión (.xlsx o .txt) mediante el menú desplegable de la nueva ventana emergente (Figura A14) y, posteriormente, seleccionar el archivo deseado y hacer click en “Abrir”.

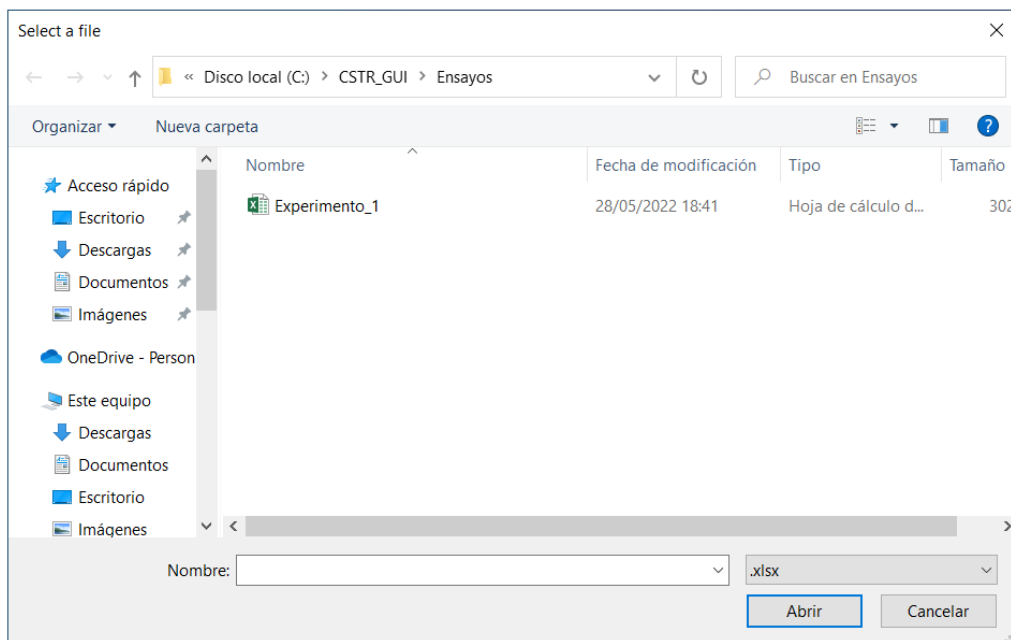


Figura A14. Importación de datos

Para obtener un conjunto de datos válido para la identificación, se recomienda:

- Introducir el primer escalón una vez el sistema se haya estabilizado.
- Evitar que la primera entrada sea igual al valor nominal (escalón de valor nulo).

- Dejar un margen de separación temporal entre distintos escalones, de forma que el sistema siempre se estabilice.
- Evitar que dos entradas consecutivas tengan un valor muy próximo.

Haciendo click en el botón “Variables de Proceso” se generarán tres gráficas: En una se representa la evolución temporal de los estados, tanto del proceso como del modelo identificado. En otra se representa la evolución de la variable de perturbación, la potencia calorífica disipada y la constante cinética. En la última, se representan los datos del caudal de refrigeración y de porcentaje de abertura de la válvula.

5.2. Exportación de datos de simulación

Tras obtener los datos con cualquiera de los métodos disponibles (simulación desde la herramienta, carga de datos o importación de datos), en la parte inferior se habilitará una nueva zona, donde se permite exportar los datos de la simulación. El usuario puede elegir el tipo de extensión del archivo, disponiendo de las siguientes opciones:

- Hoja de cálculo “.xlsx”.
- Archivo de texto “.txt” (con datos separados por comas).

El nombre del archivo deberá indicarse en el correspondiente campo editable antes de llevar a cabo la exportación de datos. Al hacer click en el botón “Exportar” se abrirá una nueva ventana en la que deberá seleccionar el directorio de destino y, pulsando en “Seleccionar carpeta”, se mostrará un mensaje confirmando la ruta donde se han almacenado los datos (Figura A15).

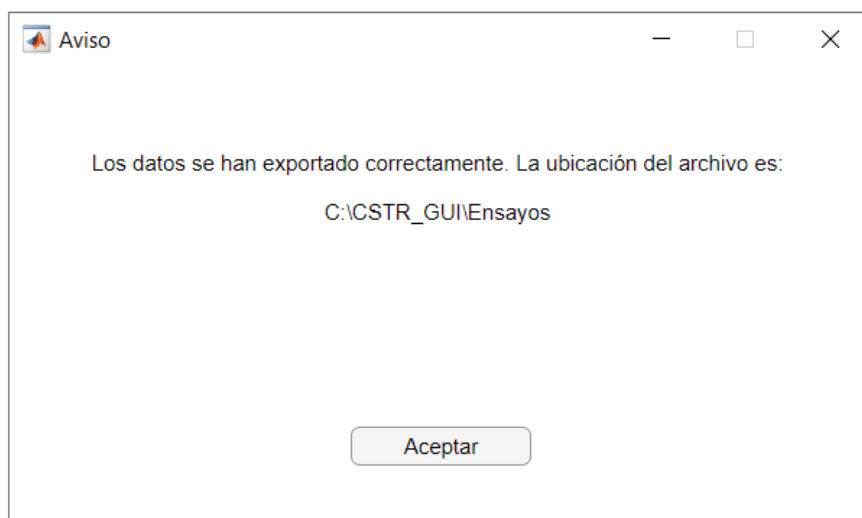


Figura A15. Mensaje de confirmación de exportación correcta de datos y directorio de ubicación

5.3. Identificación paramétrica

En la parte inferior, se dispone de un panel donde es posible realizar la identificación paramétrica de una función de transferencia aproximada, a partir de los datos de la simulación. Existen dos posibilidades:

- Identificación única: Se identifica una única función de transferencia para cada estado interno del proceso, a partir de los datos de simulación.
- Identificación por puntos: Se identifica una función de transferencia por cada escalón introducido en el panel de ensayo, a partir de los datos de simulación y para cada estado interno del proceso.

Al hacer click sobre el botón “Identificación”, se realizará la identificación con la opción seleccionada, seguida de una simulación con la función de transferencia calculada. Los nuevos datos se mostrarán superpuestos en las gráficas (Figura A16), permitiendo observar las diferencias entre la respuesta temporal del modelo del reactor y del modelo calculado.

En el algoritmo de identificación, los datos se registran a partir del primer escalón introducido, por lo que es recomendable aplicar dicho escalón en un instante que no comprometa la estabilización inicial. Por ello, en la herramienta de ensayos, el menor instante de tiempo admisible es de un minuto, aunque se recomienda que sea mayor en caso de que el valor inicial de la variable manipulada sea diferente al nominal. Asimismo, en caso de seleccionar el método de identificación por puntos, los escalones no pueden ser nulos y debe seleccionarse un primer valor para la variable manipulada diferente al valor inicial.

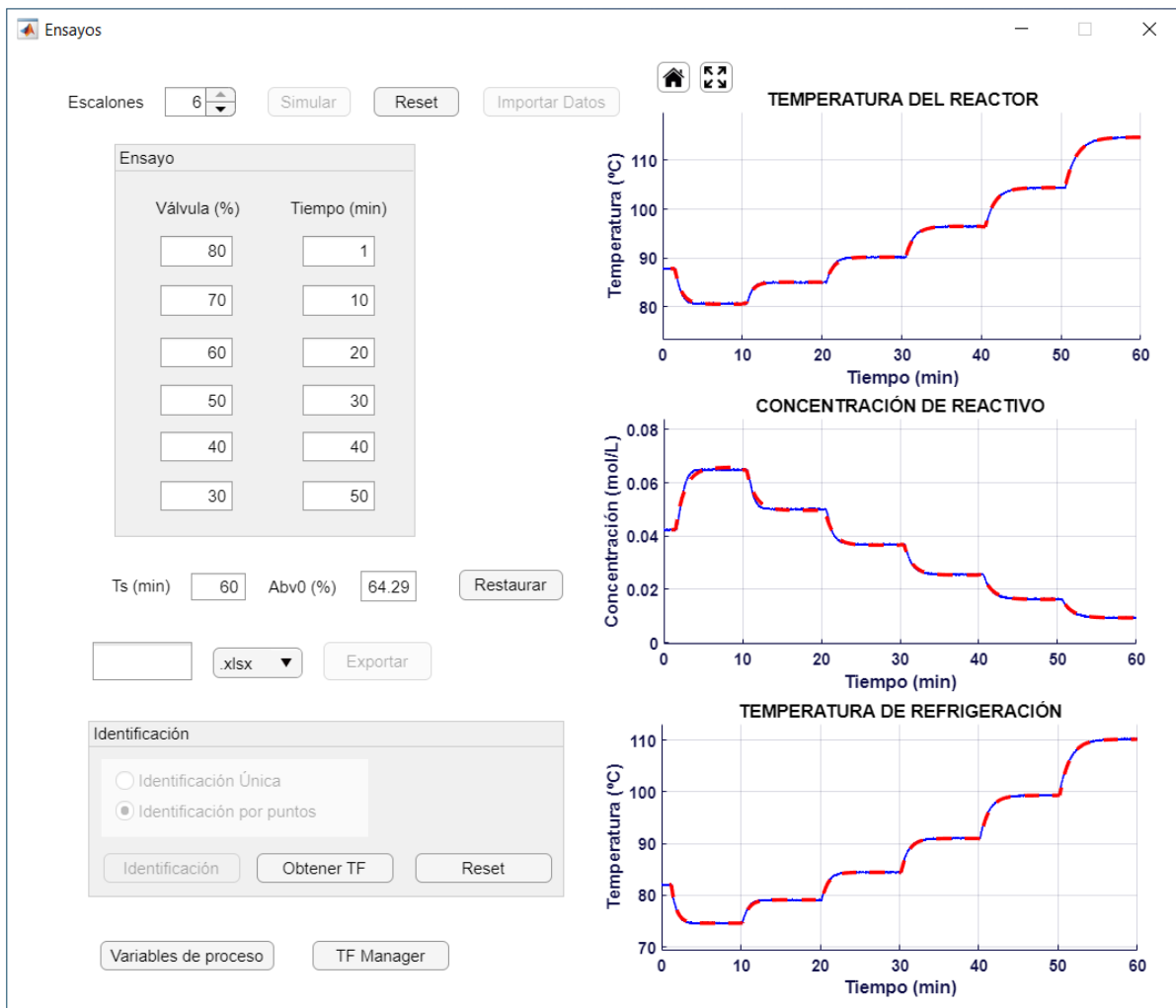


Figura A16. Resultados de la identificación paramétrica mediante la opción “Identificación por puntos”

Con el botón “Obtener TF” emergerá una nueva ventana (Figura A17) donde se mostrarán las expresiones matemáticas de las funciones de transferencia en tiempo continuo que se han obtenido en la identificación. Estas funciones engloban la dinámica del proceso (asociada a cada una de las tres salidas consideradas) y de la válvula localizada en la línea de refrigeración. En caso de haber seleccionado la opción “Identificación por puntos”, se mostrarán de forma numerada todos los modelos obtenidos a partir de los datos registrados para cada entrada en escalón.

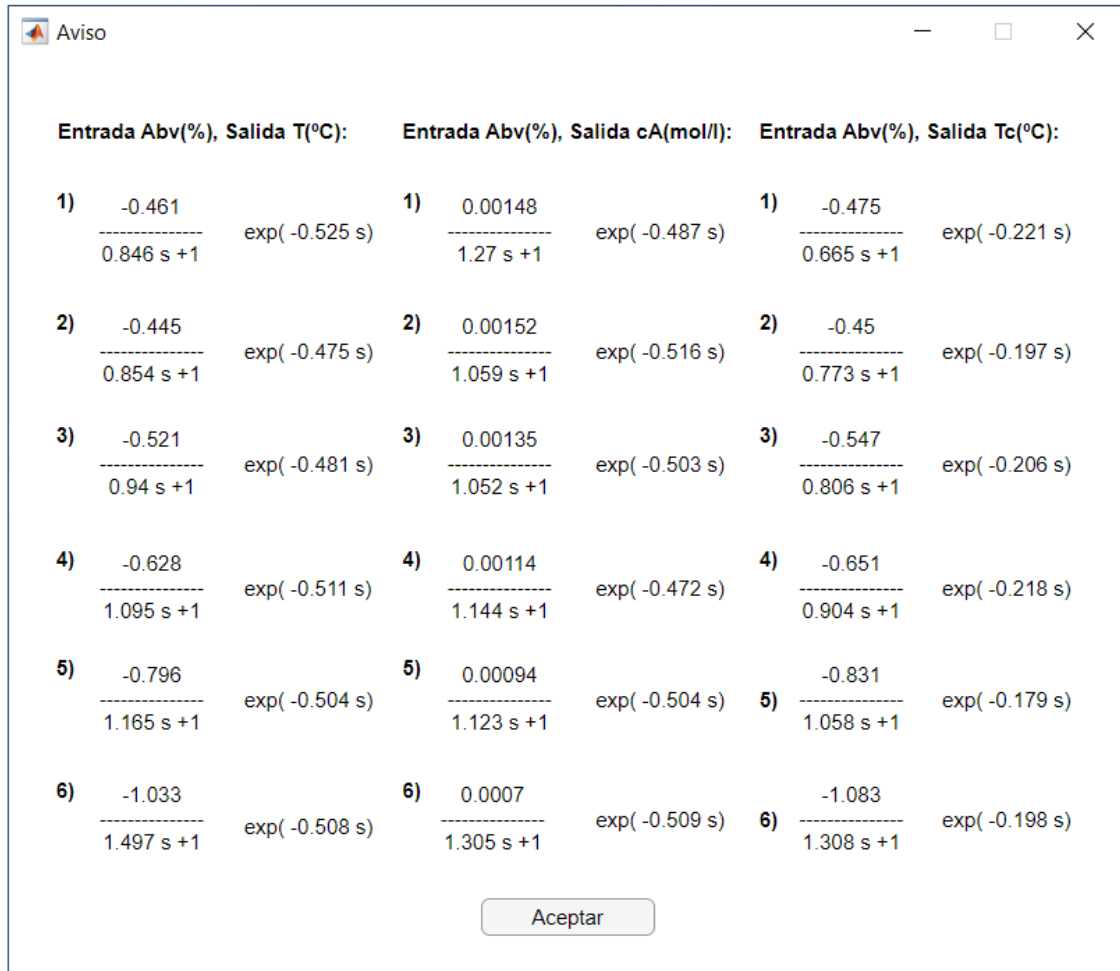


Figura A17. Modelos lineales obtenidos con una identificación paramétrica mediante la opción “Identificación por puntos”

La numeración de los modelos lineales obtenidos mediante el método de identificación por puntos hace referencia al orden en el cual se han aplicado los cambios en los valores de la variable de entrada, siendo el número 1 el relativo al primer escalón introducido y el número 6, al último. En la parte superior, se indica qué relación de entrada/salida corresponde a cada columna representada, que dependerá del tipo de configuración establecida para el controlador, así como el tipo de modelo seleccionado.

Con el botón “Reset”, se eliminarán los datos de la identificación, permitiendo realizarla de nuevo con cualquiera de las dos opciones disponibles.

5.4. Herramienta de análisis, guardado en memoria y exportación de funciones de transferencia

En la parte inferior de la ventana, se dispone del botón “TF Manager”, mediante el cual se accede a una herramienta para analizar funciones de transferencia (Figura A18).

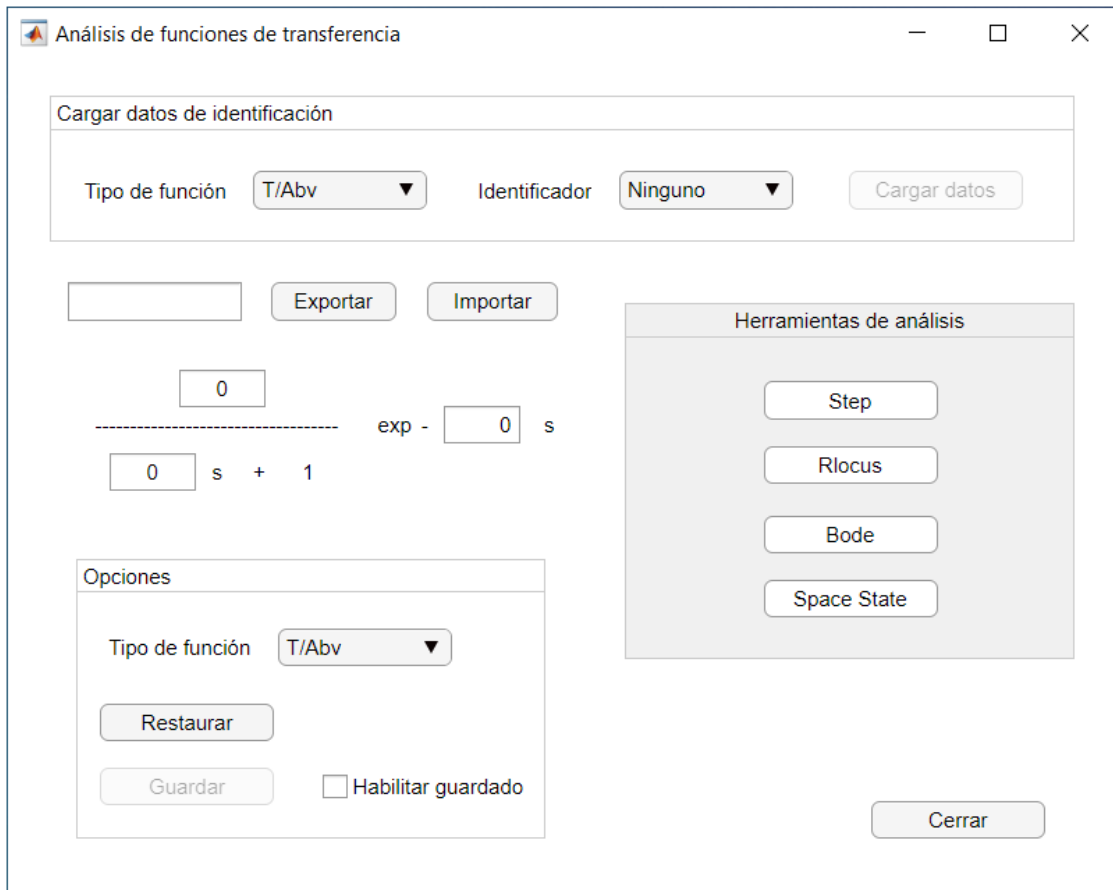


Figura A18. Herramienta para el análisis de funciones de transferencia

Dentro de la herramienta, en la parte superior, se muestra un área destinada a cargar los datos obtenidos de la identificación paramétrica, en caso de haberla realizado previamente. En el menú desplegable del identificador debe seleccionarse un valor entero, que corresponde al número de modelo identificado (ver apartado 5.3) y en el menú “Tipo de función” debe indicarse la relación que se desea cargar. Con el botón “Cargar” se mostrarán en la función de transferencia los parámetros del modelo correspondiente al identificador y al tipo de función seleccionados.

Por otro lado, es posible exportar los parámetros de una función de transferencia a un archivo de texto de extensión .txt, con datos separados por comas. Para ello, indique el nombre del archivo en el recuadro de texto y haga click en el botón “Exportar”. Seleccione la carpeta donde desea que se guarde el archivo y, posteriormente, haga click en “Seleccionar Carpeta”. Se mostrará una ventana confirmando la exportación y la ruta donde se ha guardado (Figura A15). Mediante el botón “Importar” es posible seleccionar un archivo de texto previamente exportado y cargar los datos en los campos editables (esto no modificará la función de transferencia guardada en la memoria de la aplicación).

En la zona inferior, se dispone de un marcador que habilita el guardado en memoria de la función de transferencia. Para realizar esta acción, es necesario indicar en el menú desplegable del panel de opciones el tipo de función de transferencia que corresponde a los parámetros introducidos en los campos editables. Posteriormente, haciendo click en el botón “Guardar”, se mostrará un mensaje si la operación ha resultado exitosa.

Las funciones de transferencia son utilizadas por la aplicación para las siguientes operaciones:

- Cálculo de los parámetros de control de un PID mediante reglas basadas en IMC
- Cálculo de los parámetros del control P mediante reglas basadas en ITAE
- Cálculo de los parámetros de la acción anticipativa

- Deducción del modelo en escalón utilizado por el algoritmo de control predictivo DMC

Los modelos guardados en la GUI son accesibles desde la herramienta. Para ello, seleccione el tipo de relación en el menú desplegable y haga click en el botón “Restaurar”. Se cargarán en los campos editables los parámetros del modelo correspondiente.

En la memoria de la aplicación, se dispone de las funciones de transferencia que relacionan cada variable de estado interno (temperatura del reactor, concentración de reactivo a la salida y temperatura de refrigeración a la salida) con las siguientes entradas:

- Abertura de la válvula de refrigeración.
- Temperatura de refrigeración a la entrada.
- Temperatura de entrada al reactor.
- Temperatura de refrigeración de consigna.

Todas estas funciones son accesibles y pueden analizarse o modificarse mediante esta herramienta. Para ello debe acceder previamente a la ventana de ensayos correspondiente, según el tipo de entrada y en función de si se ha configurado o no un controlador interno (véase apartado 5).

Para restaurar las funciones de transferencia por defecto (que son cargadas en la memoria de la aplicación al inicio de la misma), diríjase a la ventana de “Configuración de Simulación” y haga click en el botón “Restaurar TF” (Véase apartado 3).

En la zona de la derecha se dispone de varias opciones para analizar la función de transferencia correspondiente a los valores de los parámetros introducidos en los campos editables:

- “Step response”: muestra la respuesta temporal del sistema para una entrada en escalón unitario.
- “Rlocus”: Muestra una gráfica del lugar de las raíces.
- “Bode”: Muestra un diagrama de Bode y los valores de los márgenes de fase y de ganancia.
- “Space State”: Muestra el modelo lineal equivalente en espacio de estados, acompañado de información sobre el tipo de salida y la controlabilidad del sistema. Para ello, debe indicarse en el menú desplegable del panel de opciones el tipo de función de transferencia que corresponde a los datos introducidos en los campos editables. El resto de funciones de transferencia necesarias para obtener el modelo se cargarán de la memoria de la aplicación (pueden haberse importado y/o guardado previamente en memoria).

Al hacer click en el botón “Space State”, la aplicación realiza las siguientes acciones:

1. Registra los datos introducidos en los campos editables y lo relaciona con el tipo de función de transferencia indicado en el panel de opciones (los botones de importar y cargar rellenan automáticamente estos campos).
2. El resto de dinámicas necesarias para la representación en espacio de estados son tomadas de la memoria interna de la aplicación. Todos los modelos lineales guardados pueden ser modificados, por lo que es posible definir el modelo completo y evaluarlo.

6. Selección del algoritmo de control

Para cambiar el tipo de controlador, haga click en el botón “Opciones de control” de la barra superior. Se abrirá una ventana nueva (Figura A19) cuyo contenido dependerá del caso de estudio seleccionado.

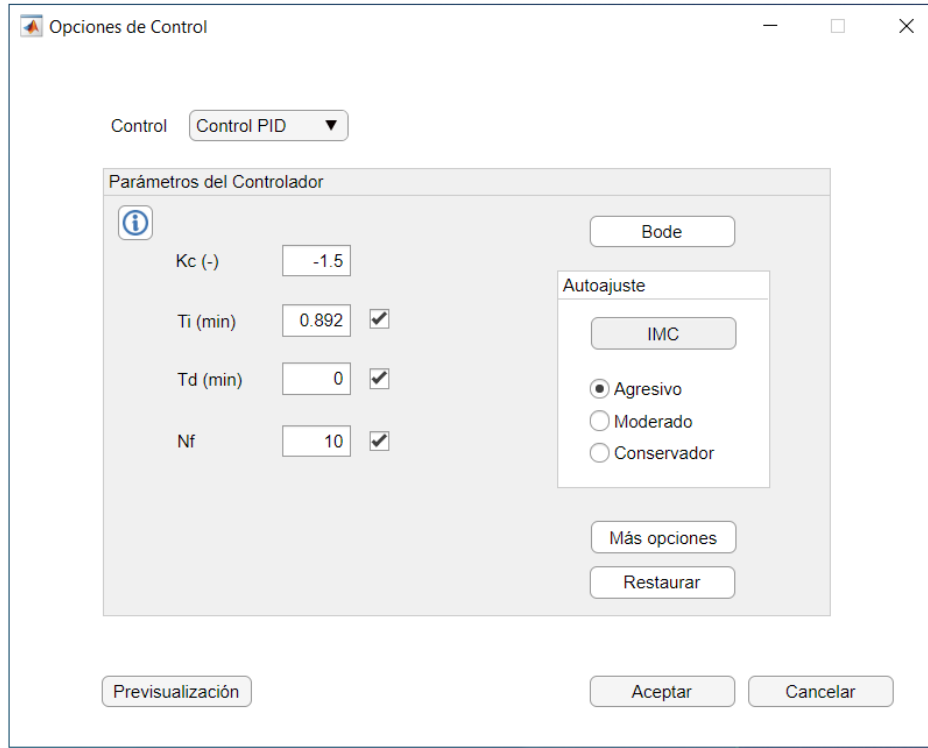


Figura A19. Ventana gráfica de opciones de control

En el menú desplegable puede seleccionarse el tipo de control. Los parámetros por defecto del controlador vienen dados en los recuadros, a modo de sugerencia, pudiendo editarse para proporcionar una solución diferente. Mediante el botón “Restaurar”, se devuelven estos campos a sus valores iniciales. Los algoritmos de control disponibles son:

- Modo manual
- Control P, PI, PD y PID
- Control PI con acción anticipativa
- Control por realimentación del estado con precompensador
- Control por realimentación del estado con acción integral
- Control predictivo DMC

Desde la ventana de configuración es posible obtener una previsualización de la respuesta temporal para una entrada en escalón unitario. Los datos se generan a partir de una simulación del proceso sin ruido con condiciones nominales de operación y se muestran en una figura aislada (Figura A20).

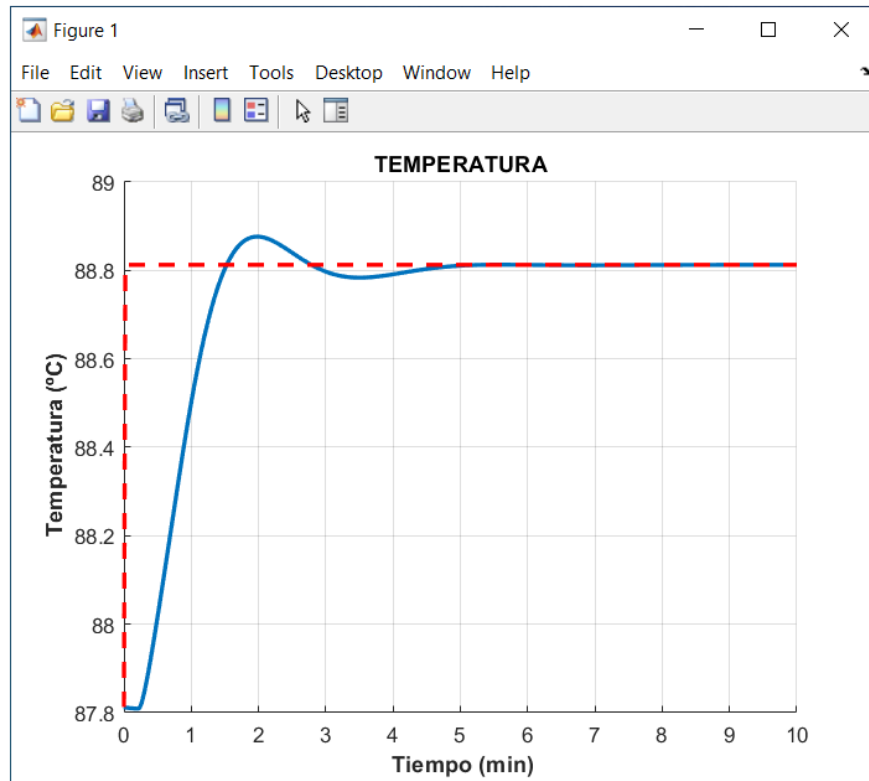


Figura A20. Previsualización de la respuesta temporal del proceso controlado para una referencia unitaria

Haciendo click sobre el botón de información, situado en la esquina superior izquierda, se desplegará una ventana con un breve resumen sobre los fundamentos teóricos del controlador seleccionado (Figura A21). Con las flechas de la parte inferior se puede navegar entre las diferentes páginas del documento.

Información

Dada las ecuaciones del modelo linealizado en el espacio de estados:

$$x(k+1) = Ax(k) + Bu(k)$$

Se utiliza la siguiente ley de control:

$$u(k) = -K_c x(k)$$

Con lo que se tiene:

$$x(k+1) = (A - BK_c)x(k)$$

El vector de estado está formado por las siguientes variables:

$$x_1 = cA$$

$$x_2 = T$$

$$x_3 = Tc$$

◀ ▶

Aceptar

Figura A21. Ventana gráfica de información sobre el controlador seleccionado

6.1. Sintonización de un controlador PID

Seleccionando el algoritmo de control PID en el menú desplegable se muestra la siguiente interfaz (Figura A22):

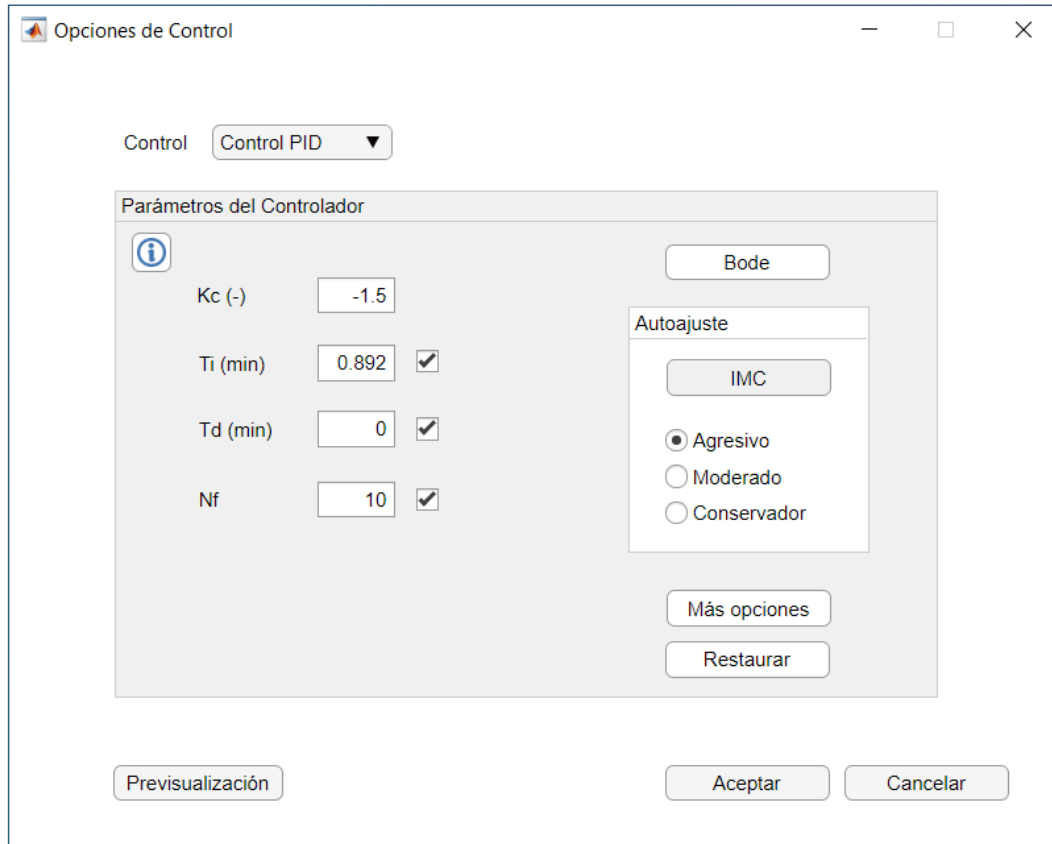


Figura A22. Sintonización de un controlador PID

Se dispone de cuatro campos editables, correspondientes a la acción proporcional (K_c), tiempo integral (T_i), tiempo derivativo (T_d) y constante del filtro derivativo (N_f). Los marcadores habilitan o deshabilitan las acciones I y D del controlador y el filtro derivativo, pudiendo seleccionar cualquier combinación.

El botón “Más opciones” mostrará una nueva ventana (Figura A23) con opciones extra de sintonización. En caso de tener deshabilitadas las acciones I y D, se permitirá seleccionar sobre qué variable se aplica el control proporcional, existiendo dos posibilidades:

- Sobre la variable incremental de salida
- Sobre la señal de error

Asimismo, en caso de habilitar la acción D y haciendo click sobre “Más opciones”, se mostrará otra ventana donde se permitirá seleccionar sobre qué variable se aplica la acción derivativa (Figura A24). Las opciones disponibles son las mismas que para el caso proporcional. Si a la acción derivativa se le añade un filtro, se mostrarán las ecuaciones correspondientes al cálculo de la misma (Figura A25) y las opciones serán también las mismas.

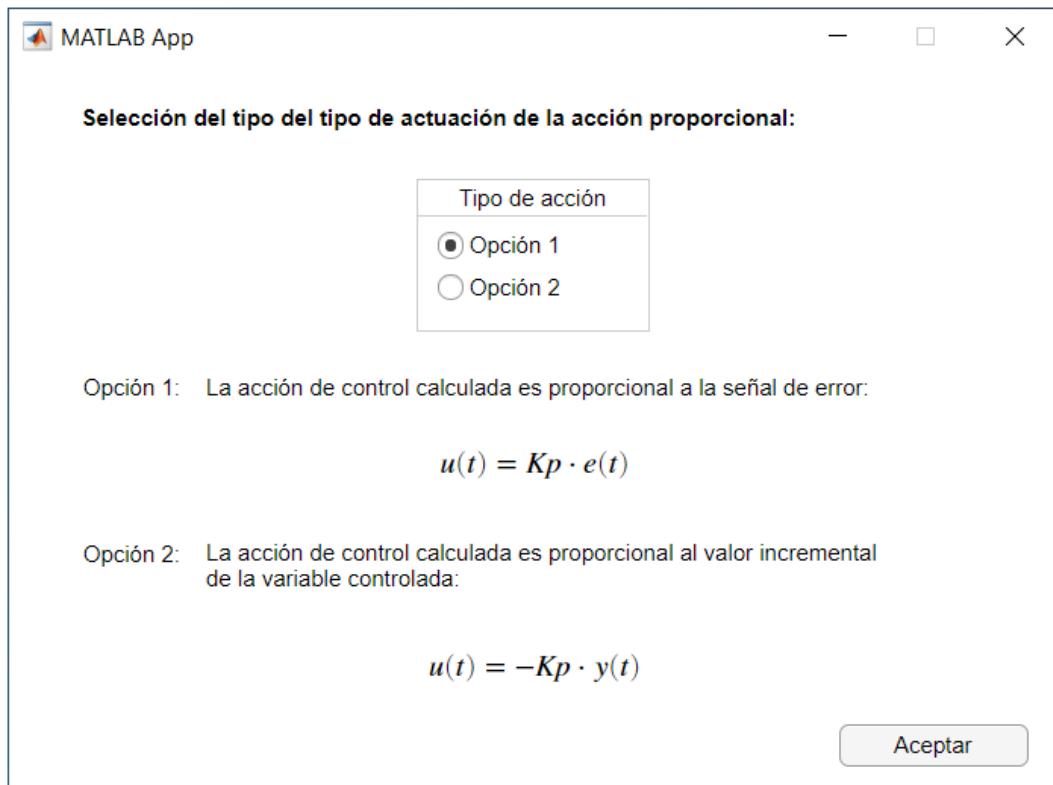


Figura A23. Configuración del tipo de acción proporcional

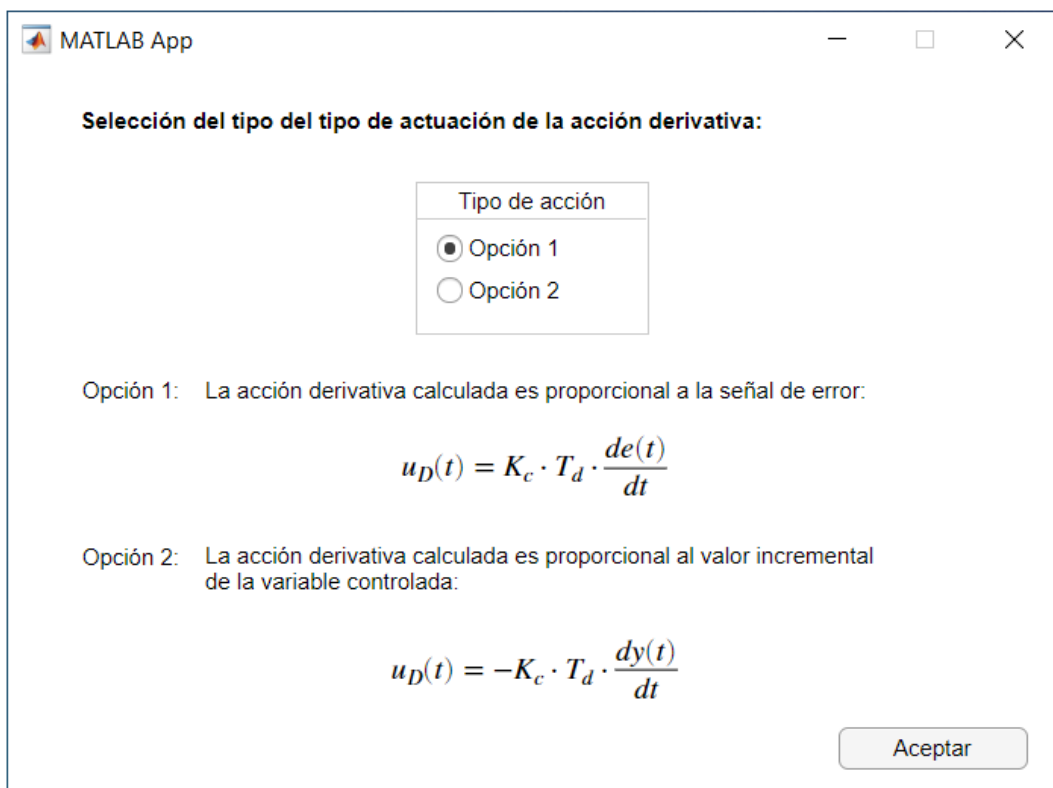


Figura A24. Configuración del tipo de acción derivativa

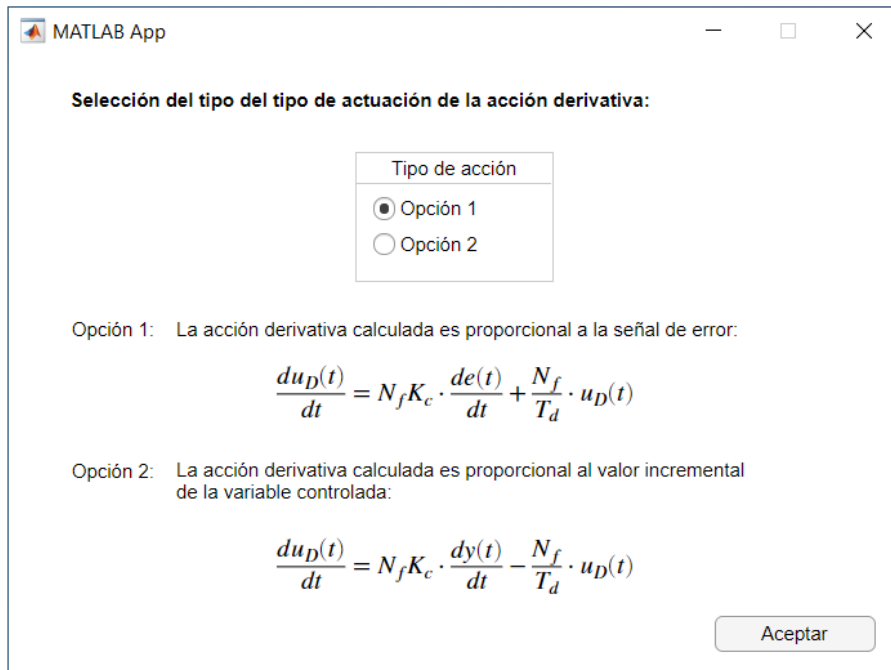


Figura A25. Configuración del tipo de acción derivativa con filtro

Mediante el panel “Autoajuste”, es posible obtener soluciones basadas en reglas de sintonización IMC (o basado en ITAE en caso de sólo proporcional). La solución proporcionada dependerá de las acciones incluidas mediante los marcadores y del tipo de respuesta deseada, pudiendo elegir entre:

- Respuesta agresiva
- Respuesta moderada
- Respuesta conservadora

Haciendo click en el botón “Bode” se mostrarán en dos nuevas ventanas el diagrama de Bode del sistema controlado en lazo abierto con los valores de los márgenes de fase y ganancia. En una ventana se representará la amplitud en valor absoluto (Figura A26) y en la otra, en decibelios (Figura 27).

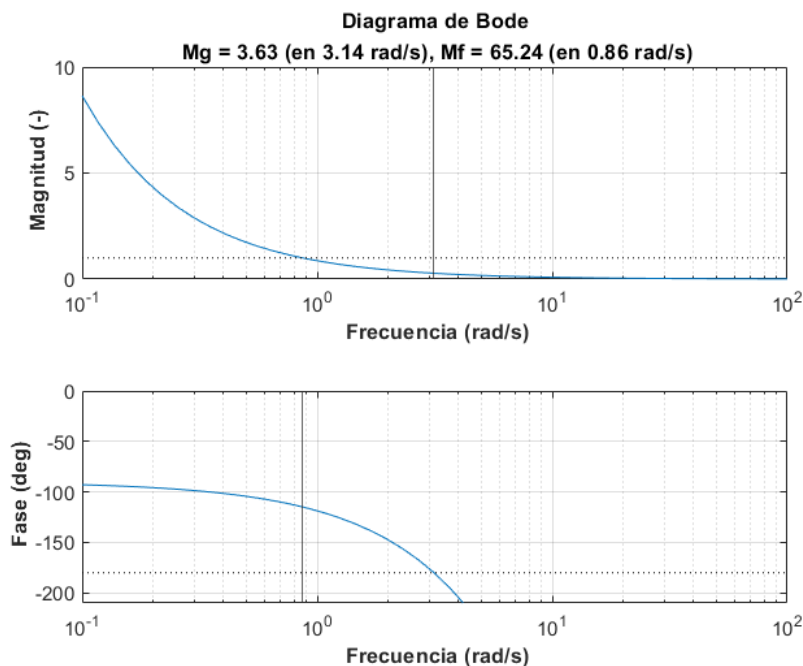


Figura A26. Diagrama de bode del proceso controlado en lazo abierto (I)

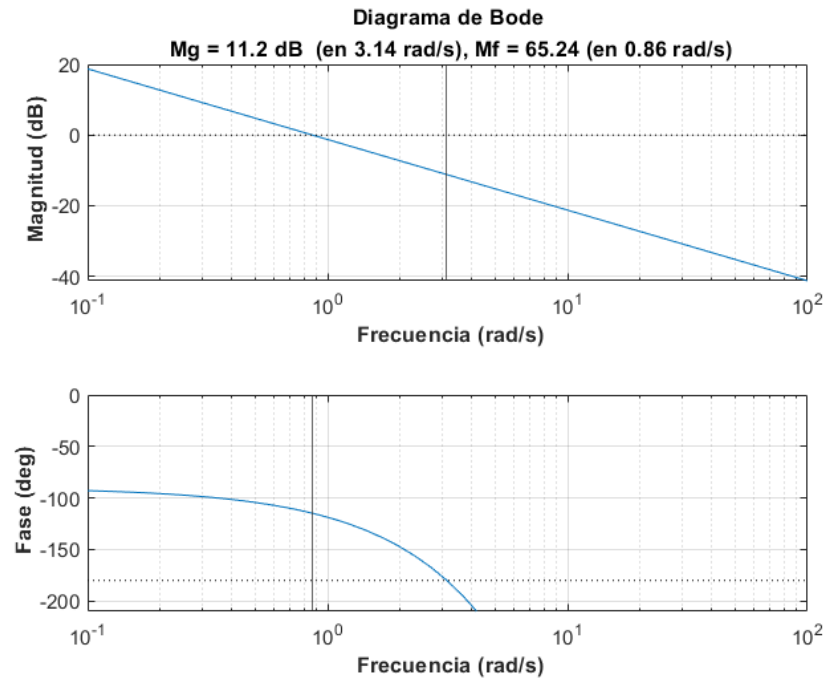


Figura A27. Diagrama de bode del proceso controlado en lazo abierto (II)

Para sintonizar manualmente un controlador basado en PID desde la ventana de configuración, con el fin de obtener una ganancia adimensional, debe utilizarse el siguiente modelo:

$$f_c \cdot G_{av}(s) \cdot G_v(s) \cdot G_p(s) \cdot G_{TD}(s) \cdot f_{pct}$$

donde $G_v(s) \cdot G_p(s)$ corresponde a la dinámica del proceso, incluyendo la válvula de control, y equivale al sistema obtenido en la identificación paramétrica. $G_{TD}(s)$ es la relación del sensor-transmisor, que depende del tipo de señal configurada, y el factor f_{pct} es la sensibilidad del bloque conversor de la salida del transmisor (en mA o V) a un valor porcentual. El producto $G_{TD}(s) \cdot f_{pct}$ puede obtenerse utilizando la Tabla A1. El controlador simulado está configurado para transformar la señal de control a la magnitud eléctrica necesaria para accionar la válvula de control (V o mA, según se tenga configurado), por lo que no es necesario incluir la dinámica del actuador de la válvula en el diseño del controlador $f_c \cdot G_{av}(s) = 1$.

6.2. Sintonización de la acción anticipativa

En caso de seleccionar desde el menú desplegable el algoritmo PI con acción anticipativa, se mostrará la ventana representada en la Figura A28.

Se dispone de dos campos editables para introducir los valores de la constante proporcional, K_c y el tiempo integral T_i . Mediante el botón “Feedforward”, se accederá a una herramienta para configurar la acción anticipativa (Figura A29).

Desde esta herramienta, en la zona izquierda de la ventana, es posible indicar las funciones de transferencia que van a tenerse en cuenta en el cálculo del controlador (relativas al proceso y al efecto de la perturbación). Para ello, rellene los campos editables o haga click en el botón “Cargar”, con lo que se rellenará automáticamente, a partir de la información guardada en la memoria de la aplicación (para modificar los modelos aproximados guardados en memoria, véase apartado 5.4). De manera análoga, es posible cargar los datos guardados en un archivo de texto (.txt) mediante el botón “Importar”. La función de transferencia del proceso indicada incluye la dinámica de la válvula de control.

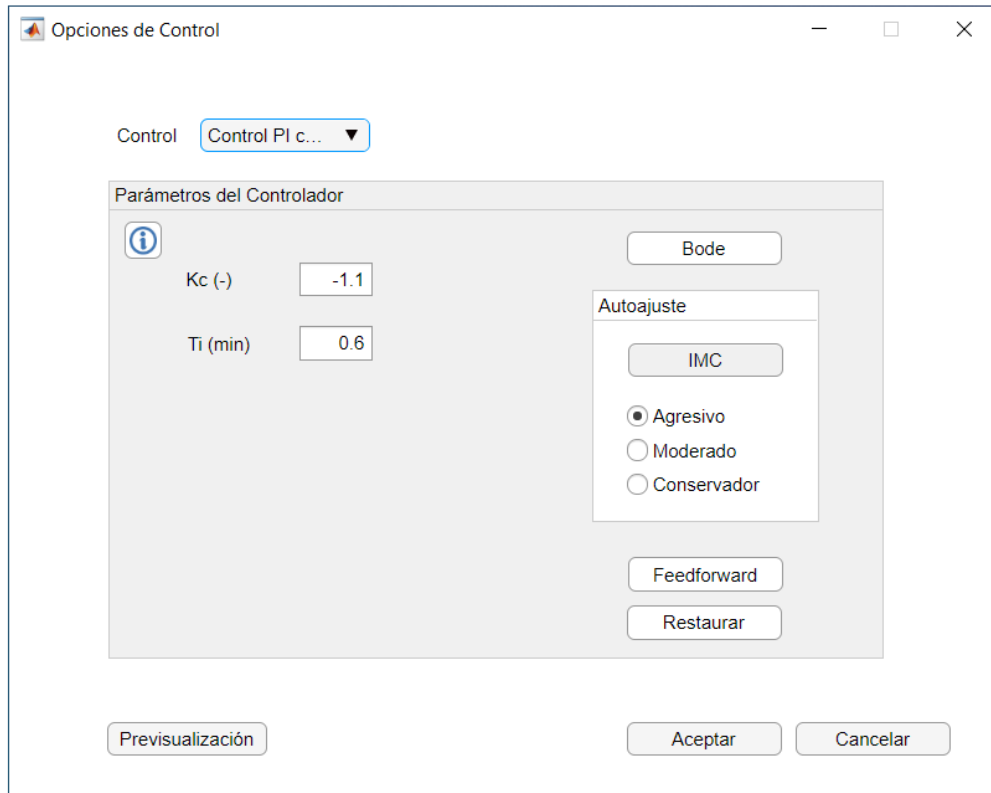


Figura A28. Sintonización de un controlador PI con acción anticipativa

A la derecha, se muestra la expresión del controlador. La sintonización es posible mediante las siguientes opciones:

- Introduciendo manualmente los parámetros del “feedforward”
- Haciendo click en el botón “Deducir del modelo”, con lo que se realizará el cálculo a partir de las funciones de transferencia indicadas en la zona izquierda de la ventana.

Más abajo, se mostrará en un recuadro el valor de la ganancia estática correspondiente al controlador introducido (este campo no es modificable).

Si desea llevar a cabo el proceso completo de identificación y posterior sintonización, deberá realizar los siguientes pasos:

- 1) En la ventana de ensayos con cambio en la variable manipulada, acceder a la herramienta de análisis de las funciones de transferencia.
- 2) Guardar en memoria interna o exportar datos de la función de transferencia aproximada.
- 3) Repetir los pasos 1 y 2 en la ventana de ensayos con cambio en la variable de perturbación.
- 4) Acceder a la configuración del controlador desde la ventana principal y seleccionar un controlador PI con acción anticipativa.
- 5) Hacer click en el botón “Feedforward”.
- 6) Cargar o importar el modelo asociado a la dinámica de perturbación.
- 7) Cargar o importar el modelo asociado a la dinámica del proceso incluyendo la correspondiente a la válvula de control.
- 8) Hacer click en el botón “Deducir del modelo”.

Figura A29. Herramienta para sintonizar la acción anticipativa

El cálculo de la acción anticipativa que se lleva a cabo es el siguiente:

$$G_{FF}(s) = -\frac{G_d(s)}{G_v(s) \cdot G_p(s) \cdot G_{TD}(s) \cdot f_{pct}}$$

Desde el panel “Tipo de acción anticipativa” puede seleccionarse si el controlador se configurará en modo estático o en dinámico.

- Modo estático: Se utiliza el dato de la ganancia estática de la acción anticipativa.
- Modo dinámico: Se utilizan los datos dinámicos indicados en la función de transferencia de la acción anticipativa.

El panel “Previsualización” contiene los elementos necesarios para obtener una previsualización del modelo controlado, para una entrada unitaria de perturbación. Además, se dispone de un campo editable donde es posible modificar el valor de la referencia para evaluar la influencia de la perturbación en distintos puntos de funcionamiento y un marcador para activar o desactivar la acción anticipativa. Los parámetros del controlador PI utilizados en esta simulación de prueba, son los correspondientes a los campos editables de la ventana de configuración de control, (K_c y T_i), que deberán seleccionarse antes de acceder a esta herramienta.

Una vez seleccionado el tipo de acción anticipativa deseada desde el panel correspondiente y haciendo click sobre el botón “Aplicar”, la ventana se cerrará y se guardarán los cambios de manera provisional en la memoria de la aplicación. Para hacer efectivos dichos cambios, deberá de aceptarse la configuración del PI desde la ventana de configuración de control, haciendo click en el botón “Aceptar”. En la ventana principal se mostrará el diagrama de proceso del reactor controlado con acción anticipativa (Figura A30).

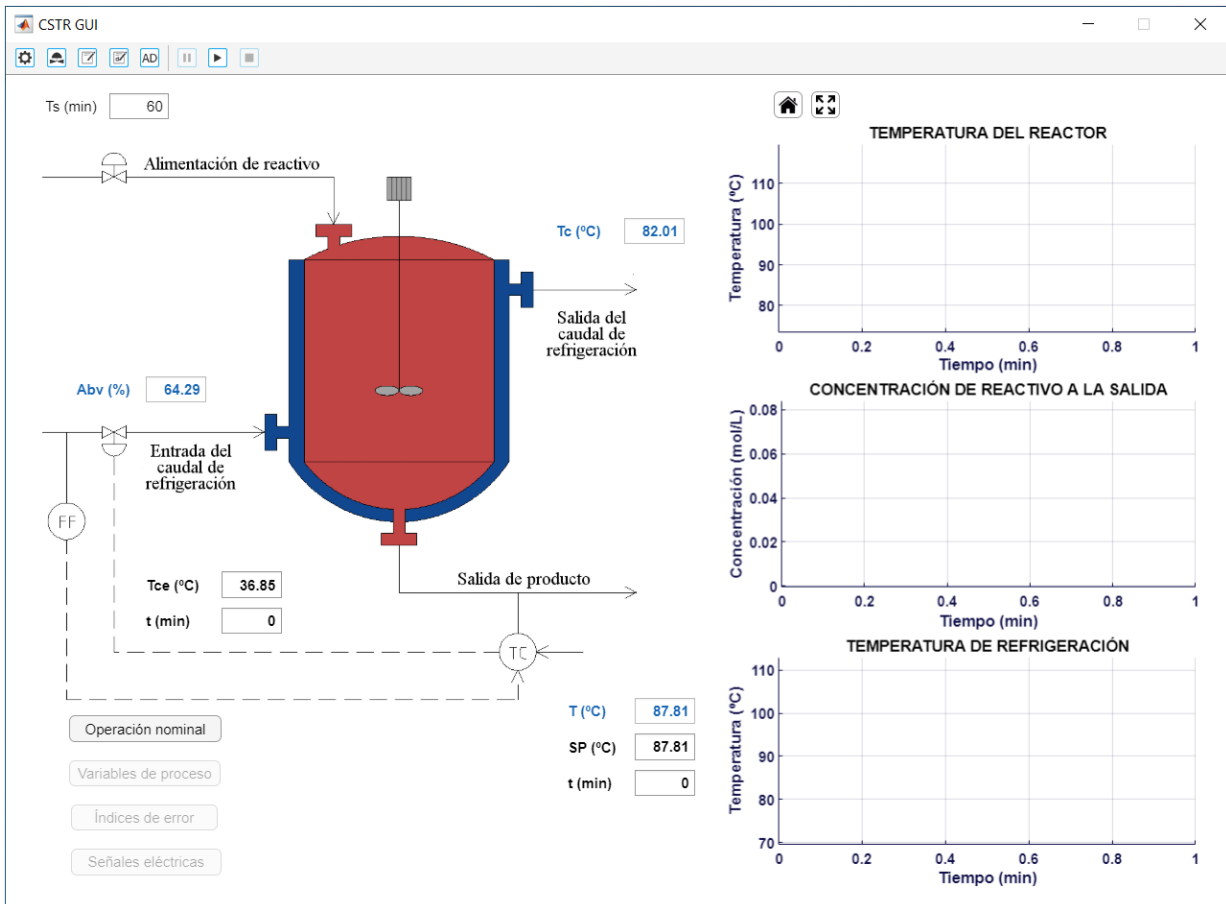


Figura A30. Lazo de control configurado con acción anticipativa

6.3. Sintonización de un controlador por realimentación del estado

En caso de seleccionar un algoritmo de control por realimentación del estado desde el menú desplegable, se mostrará la siguiente interfaz (Figura A31).

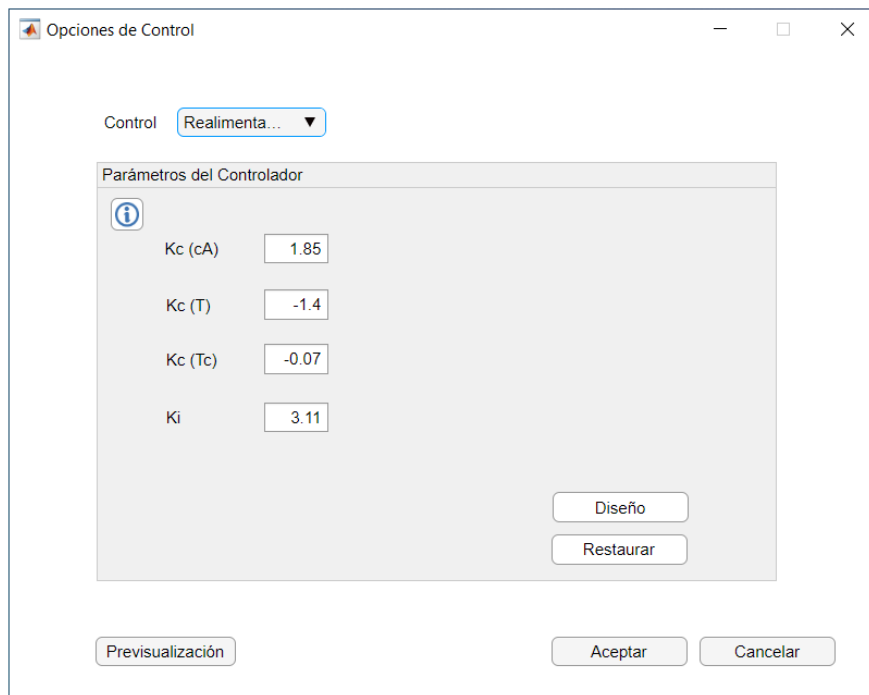


Figura A31. Sintonización de un controlador por realimentación del estado con acción integral

Los campos editables disponibles representan la ganancia aplicada a cada variable interna del proceso y a la integral del error (en caso de incluir acción integral). Mediante el botón “Diseño” se accede a una herramienta para diseñar los parámetros del controlador mediante los métodos de diseño de asignación de polos y LQR (Figura A32).

Una vez se haya accedido a la herramienta, deberá seleccionarse un método de diseño desde el panel “Método de diseño”. En función de la opción escogida, se habilitarán los campos editables correspondientes para ajustar el controlador.

Para la sintonización mediante asignación de polos, deberá indicar en los campos editables correspondientes los polos deseados para el sistema en lazo cerrado. En caso de introducir dos polos iguales, no se permitirá el cálculo de los parámetros del controlador y, por lo tanto, no se aplicará ningún cambio. Mediante el panel de autoajuste se permite seleccionar entre tres tipos de soluciones predeterminadas: agresiva, moderada y conservadora.

De igual forma, para sintonizar el controlador mediante LQR, deberá introducir el valor del factor de ponderación de la energía de control. Si el algoritmo seleccionado incluye acción integral, también podrá modificarse el factor de ponderación de los incrementos en la integral del error. La agresividad del controlador se verá potenciada por:

- Un valor elevado del parámetro γ
- Un valor reducido del parámetro μ

Mediante el panel de previsualización, se permite obtener una simulación del proceso sin ruido para una referencia unitaria, a partir de los parámetros indicados para el controlador.

Asignación de polos:

p1

p2

p3

p4

LQR:

$$Q = \begin{bmatrix} 1/0.01^2 & 0 & 0 & 0 \\ 0 & 1/1^2 & 0 & 0 \\ 0 & 0 & 1/1^2 & 0 \\ 0 & 0 & 0 & \gamma \cdot 1/1^2 \end{bmatrix}$$

$$R = \mu \cdot 1/1^2$$

μ

γ

Autoajuste Asignación

Agresivo

Moderado

Conservador

Vector de estados:

$$x = \begin{bmatrix} cA \\ T \\ Tc \\ x_i \end{bmatrix} \begin{Bmatrix} mol/l \\ ^\circ C \\ ^\circ C \\ ^\circ C \end{Bmatrix}$$

Autoajuste LQR

Agresivo

Moderado

Conservador

Método de diseño

Asignación

LQR

Figura A32. Herramienta para sintonización de un controlador por realimentación del estado

Una vez seleccionado el método de diseño y sintonizado el controlador, al hacer click sobre el botón “Aplicar” se cerrará la herramienta, se calcularán los elementos de la matriz de realimentación del estado (incluyendo K_i en caso de seleccionar acción integral) y se modificarán los campos editables correspondientes a los parámetros del controlador.

6.4. Sintonización de un controlador predictivo DMC

Al seleccionar desde el menú desplegable el algoritmo de control predictivo DMC, se mostrará la siguiente interfaz (Figura A33):

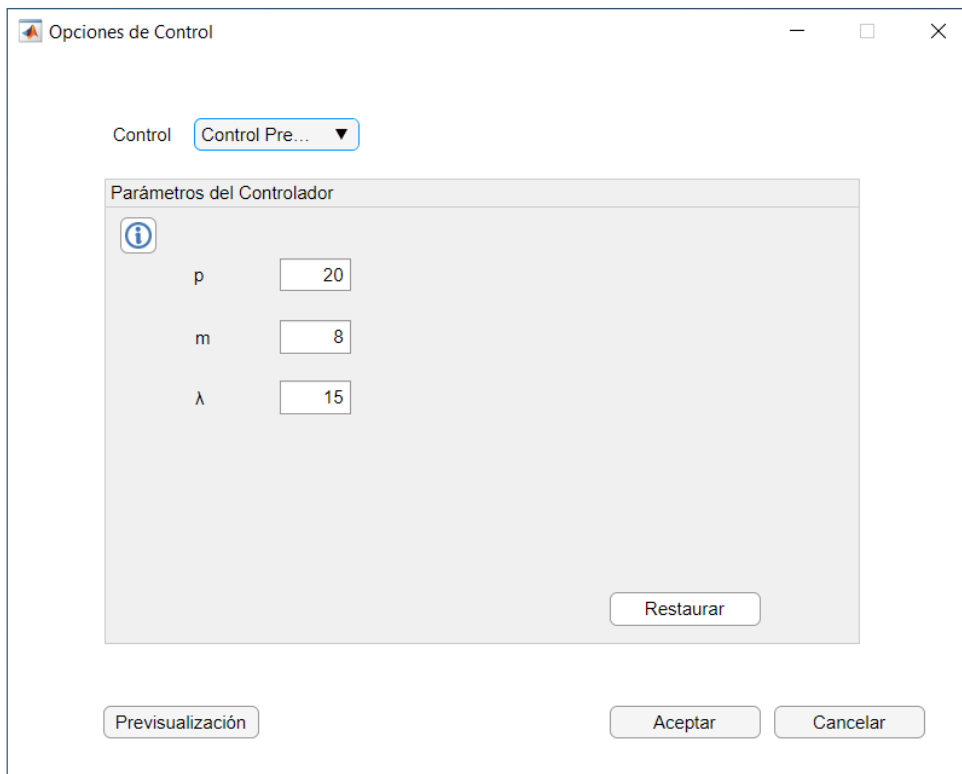


Figura A33. Sintonización de un controlador predictivo DMC

Los parámetros modificables son:

- El horizonte de predicción “ p ”, expresado en múltiplos del tiempo de muestreo.
- El horizonte de control “ m ”, expresado en número de elementos de la secuencia de control calculada.
- El factor de ponderación de los esfuerzos de control “ λ ” (adimensional).

El tiempo de muestreo del controlador puede observarse haciendo click en el botón de información y está fijado en un valor de 0,18 minutos.

La agresividad viene determinada, principalmente, por el valor del factor “ λ ”, de forma que un valor bajo corresponde a una respuesta más rápida y un valor alto a una respuesta con menor gasto energético.

Para deducir el modelo de respuesta en escalón utilizado por el algoritmo, se parte de la función de transferencia cargada en la memoria de la aplicación. Dicha función puede modificarse desde la herramienta de ensayos con entrada en escalón, accediendo a la ventana de análisis de funciones de transferencia y guardando en memoria el modelo deseado (Véase apartado 5.4).

6.5. Configuración del control en cascada

En caso de haber seleccionado, desde el menú de configuración de simulación, el modelo de reactor encamisado con estructura de control en cascada, en la ventana de opciones de control se mostrarán dos desplegables en lugar de uno (Figura A34), que corresponden a los algoritmos del controlador maestro y esclavo. En este caso las opciones disponibles para el controlador maestro son:

- Modo manual
- Control P
- Control PI
- Control PI con acción anticipativa
- Control predictivo DMC

Para el controlador esclavo, las opciones disponibles son el modo manual, el control P y el control PI.

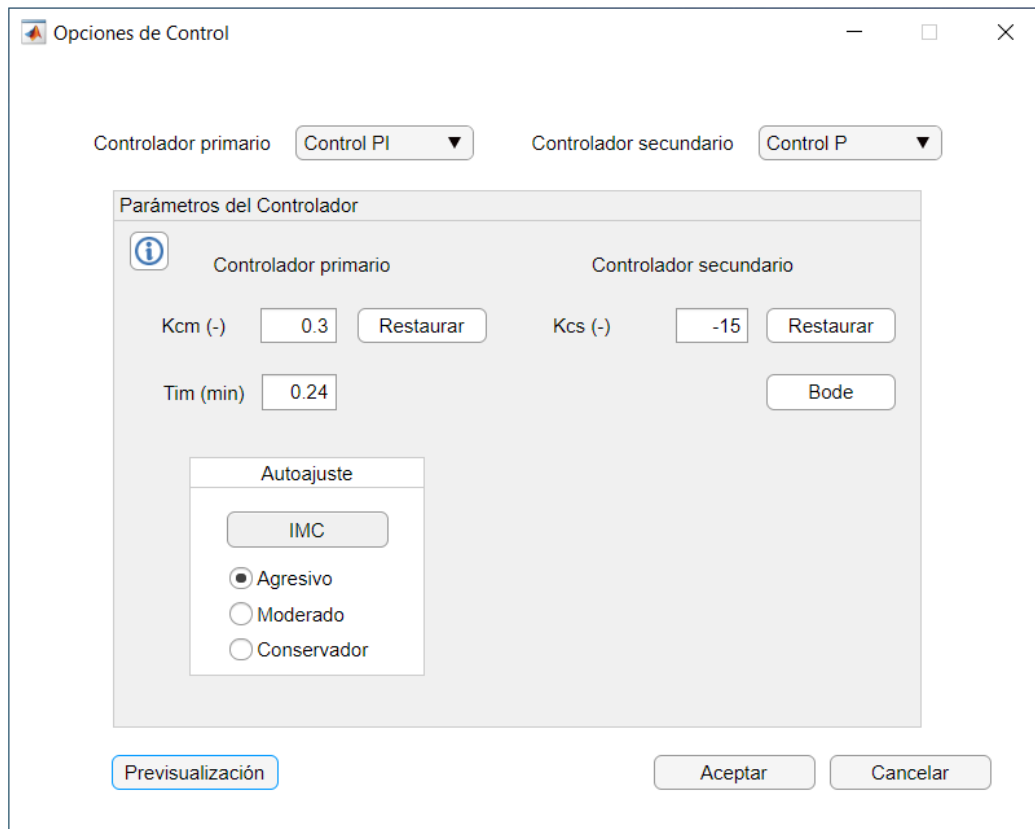


Figura A34. Ventana gráfica de opciones de control para el reactor encamisado con estructura de control en cascada

Si se configura un control P o PI para el lazo interno y el controlador primario se configura en manual, el sistema realizará un seguimiento de referencias para la variable de temperatura de refrigeración. En el caso opuesto, seleccionando un algoritmo para el primario y configurando el modo manual en el secundario, el sistema se configurará en lazo abierto, permitiendo manipular la válvula de refrigeración.

Para llevar a cabo una sintonización por IMC/ITAE a partir de modelos previamente identificados, deben realizarse los siguientes pasos:

- 1) Configurar el controlador esclavo en modo manual.
- 2) Acceder a la herramienta de ensayos con cambios en la variable manipulada.
- 3) Realizar una identificación (opcional).

- 4) Acceder a la ventana de análisis de funciones de transferencia.
- 5) Guardar en la memoria interna la función de transferencia aproximada.
- 6) Acceder a la ventana de configuración de control desde la ventana principal.
- 7) Configurar un controlador esclavo. En caso de seleccionar el control PI, éste podrá sintonizarse automáticamente a partir del modelo guardado anteriormente (Botón “IMC”). Si, por el contrario, el algoritmo seleccionado es el control P, los pasos anteriores serán opcionales, ya que deberá ajustarse manualmente, para lo cual, se recomienda sintonizar una ganancia elevada, pero que estabilice al sistema para referencias de temperatura de refrigeración.
- 8) Acceder de nuevo a la herramienta de ensayos con cambios en la variable manipulada, que ahora será la referencia de la temperatura de refrigeración.
- 9) Repetir los pasos 3 (opcional), 4, 5 y 6.
- 10) Configurar un controlador maestro. Este controlador podrá sintonizarse automáticamente a partir del modelo guardado anteriormente. En caso de seleccionar un controlador DMC, no se dispondrá de la opción de sintonización automática y el modelo de predicción utilizado será el obtenido en el paso 9.

De manera análoga, partiendo del paso 7, es posible sintonizar la acción anticipativa para la variable de perturbación externa:

- 8) Acceder a la herramienta de ensayos con cambios en la variable de perturbación, que ahora será la temperatura de entrada al reactor.
- 9) Repetir los pasos 3 (opcional), 4, 5 y 6.
- 10) Seleccionar la opción “Control PI con feedforward” para el controlador maestro, y hacer click en el botón “Feedforward”.
- 11) Cargar o importar el modelo de perturbación y el del proceso (incluyendo la dinámica de la válvula).
- 12) Hacer click en el botón “Deducir del modelo”
- 13) Hacer click en “Aplicar”. La ventana se cerrará y se guardarán los cambios temporalmente.
- 14) Hacer click en “Aceptar” para hacer efectivo los cambios

Si no se ha llevado a cabo un guardado en memoria de las funciones de transferencia necesarias, se utilizarán las configuradas por defecto al inicio de la aplicación.

El algoritmo de control DMC también tiene en cuenta un modelo de perturbación de la dinámica externa (temperatura del reactor a la entrada), que puede ser modificado desde la ventana de ensayos con cambios en la variable de perturbación (véase apartado 5.4).

Haciendo click sobre el botón “Bode” se mostrará en una figura el diagrama de Bode del sistema controlado en lazo abierto. En caso de seleccionar un control P/PI para el control maestro, el diagrama de Bode se calculará cerrando el lazo interno con los parámetros indicados y dejando abierto el externo.

7. Ensayos con control adaptativo y autoajustable

En caso de tener configurado el modelo encamisado, podrá accederse a una ventana destinada a realizar ensayos con controladores adaptativos y autoajustables (Figura A35), haciendo click en el botón correspondiente localizado en la barra superior.

Una vez se haya accedido a la herramienta, en la parte superior, se dispone de un menú desplegable donde es posible seleccionar el algoritmo de control, existiendo tres posibilidades: *Gain Scheduling PI*, *Gain Scheduling DMC* y Controlador Autoajustable RLS.

En el panel “Parámetros” se muestran los campos editables de los parámetros del controlador, que variarán en función del algoritmo de control seleccionado.

En el panel inferior pueden seleccionarse los valores de entrada del ensayo (referencias de temperatura, variable de perturbación e instantes de aplicación), así como establecer el tiempo de simulación. Al hacer click en el botón “Simulación” se mostrará la evolución de los estados del proceso en las gráficas de la parte derecha de la ventana. Para observar la evolución temporal de las variables del proceso, haga click en el botón “Variables de proceso”, localizado en la parte inferior de la ventana. Mediante el botón “Variación de parámetros” se mostrará la evolución temporal de los parámetros del controlador seleccionado.

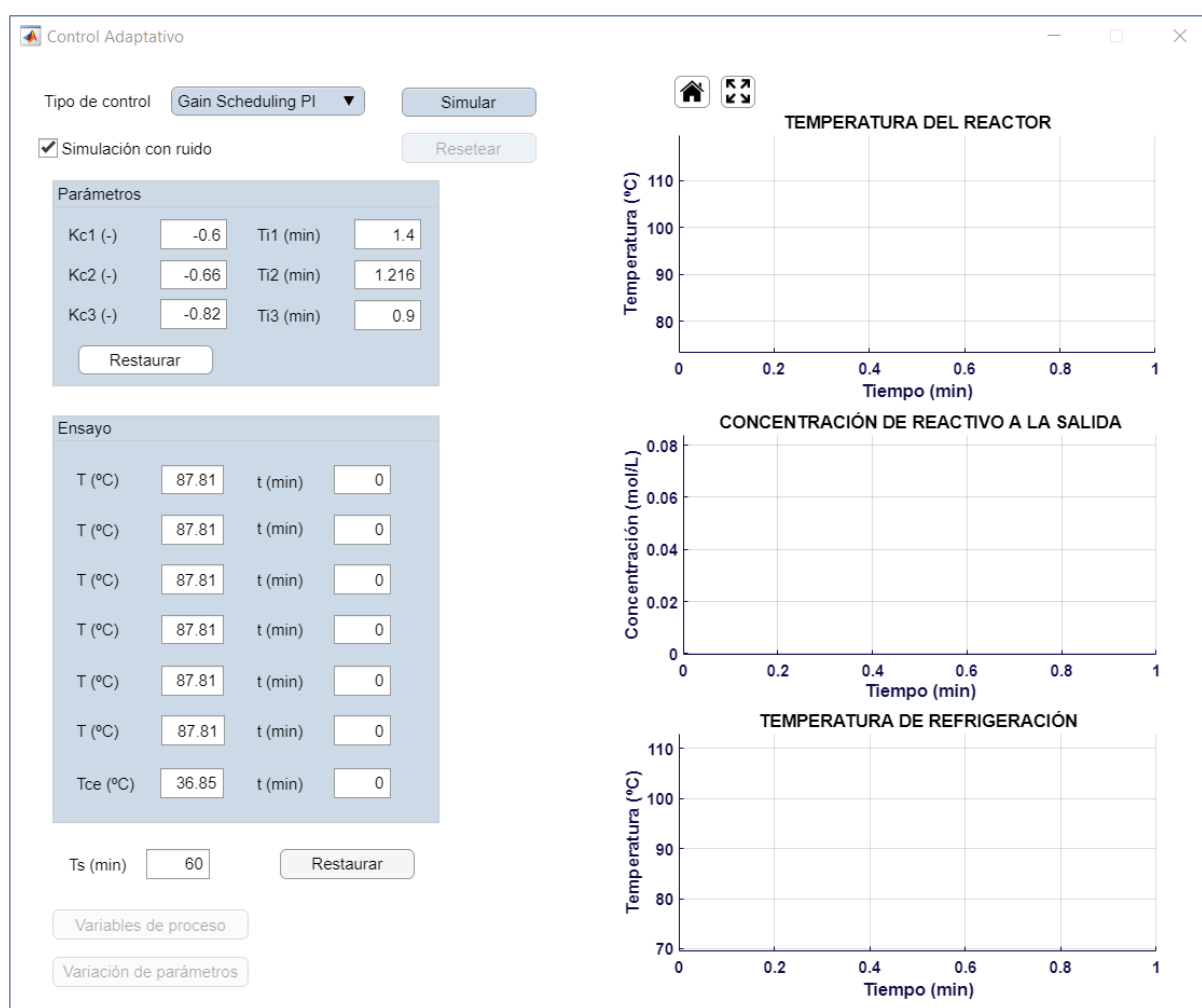


Figura A35. Ventana gráfica de control adaptativo y autoajustable

7.1. Configuración del control Gain Scheduling PI

Con el algoritmo Gain Scheduling PI seleccionado, se permite establecer fuera de línea un conjunto de tres controladores PI, definiendo una tabla donde se relaciona cada par de parámetros (K_C y T_i) con la zona que van a controlar. Cada una de estas zonas está determinada por un rango de temperatura, de forma que, en función de la medida de la temperatura del reactor, actuará el controlador correspondiente, de acuerdo a la información descrita en la tabla.

En el panel “Parámetros” se muestran los campos editables de la configuración del algoritmo de control (Figura A36), de manera que se pueden definir los siguientes valores:

- K_{c1} y T_{i1} : Constante proporcional y tiempo integral del controlador PI que actuará en el rango de [105 ; 114] °C
- K_{c2} y T_{i2} : Constante proporcional y tiempo integral del controlador PI que actuará en el rango de [85 ; 105] °C
- K_{c3} y T_{i3} : Constante proporcional y tiempo integral del controlador PI que actuará en el rango de [78,5 ; 85] °C

Tipo de control: Gain Scheduling PI ▼ Simular

Simulación con ruido Resetear

Parámetros			
Kc1 (-)	-0.6	Ti1 (min)	1.4
Kc2 (-)	-0.66	Ti2 (min)	1.216
Kc3 (-)	-0.82	Ti3 (min)	0.9

Restaurar

Figura A36. Parámetros de ajuste para el control PI con estrategia “Gain Scheduling”

La conmutación entre un controlador y otro se lleva a cabo de manera progresiva y teniendo en consideración un valor de tolerancia de 3 °C, de forma que se evita oscilaciones en el cambio de parámetros. Los rangos de operación, así como el valor de la tolerancia, no son modificables.

7.2. Configuración del control autoajustable RLS

El algoritmo de control autoajustable está basado en un controlador PI sintonizado de forma automática mediante reglas IMC, a partir del valor indicado en el campo editable correspondiente a la constante de tiempo deseada en lazo cerrado. La supervisión del modelo lineal considerado se lleva a cabo mediante un algoritmo de identificación basado en mínimos cuadrados recursivos (RLS), que es accionado por un detector de excitación.

Los campos editables disponibles para configurar el algoritmo (Figura A37) son los siguientes:

- K_{c0} : Constante proporcional del controlador PI inicial
- T_{i0} : Tiempo integral del controlador PI inicial
- τ_{lc} : Constante de tiempo deseada para el sistema en lazo cerrado.
- λ_o : Factor de olvido de la identificación paramétrica. Seleccionar un valor bajo para una mayor velocidad de identificación o un valor alto para una mayor robustez en la identificación.

Tipo de control Auto-Ajustable (R... ▼ Simular

Simulación con ruido Resetear

Parámetros

Kc0 (-)	<input type="text" value="-0.69"/>	λo	<input type="text" value="0.995"/>
Ti0 (min)	<input type="text" value="0.892"/>		
τlc (min)	<input type="text" value="1"/>		

Restaurar

Figura A37. Parámetros de ajuste para el control autoajustable basado en RLS

7.3. Configuración del control Gain Scheduling DMC

En el menú desplegable se dispone de una estrategia “*Gain Scheduling*” basado en el algoritmo de control predictivo DMC.

Los campos editables del panel “Parámetros” correspondientes a esta configuración (Figura A38) son los siguientes:

- p : Horizonte de predicción
- m : Horizonte de control
- λ_1 : Factor de ponderación de las acciones de control en el rango de [105 ; 114] °C
- λ_2 : Factor de ponderación de las acciones de control en el rango de [85 ; 105] °C
- λ_3 : Factor de ponderación de las acciones de control en el rango de [78,5 ; 85] °C

Tipo de control Gain Scheduling ... ▼ Simular

Simulación con ruido Resetear

Parámetros

p	<input type="text" value="20"/>	λ1	<input type="text" value="15"/>
m	<input type="text" value="8"/>	λ2	<input type="text" value="15"/>
		λ3	<input type="text" value="15"/>

Restaurar

Figura A38. Parámetros de ajuste para el control predictivo DMC con estrategia “*Gain Scheduling*”

La aplicación dispone en su configuración de tres modelos de respuesta en escalón preestablecidos, que se utilizarán en función del punto de operación (determinado por la medida de temperatura). Al igual que en la configuración “*Gain Scheduling PP*”, está configurado un valor de tolerancia de 3 °C (véase apartado 7.1).

7.4. Sintonización de controladores adaptativos

Para sintonizar un controlador adaptativo con estrategia “*Gain Scheduling PI*” se recomienda obtener tres modelos lineales, correspondientes a las tres zonas de operación, y utilizarlos en el diseño de un controlador individual para cada modelo (puede hacerse uso de las ventanas de “Ensayo” y del botón IMC de la ventana de “Configuración de control”). Los parámetros obtenidos deberán indicarse en el panel correspondiente, dentro de la ventana de configuración del control adaptativo.

Para el caso del controlador autoajustable, basta con obtener un modelo para el punto de operación deseado, diseñar, a partir de éste, el controlador PI inicial y determinar la constante de tiempo deseada para el sistema en lazo cerrado.

De manera alternativa, puede partirse de un conocimiento nulo del proceso, introduciendo unos parámetros arbitrarios para el controlador PI inicial e indicando la constante de tiempo deseada para el sistema en lazo cerrado. El algoritmo variará los parámetros del controlador para ajustarse a dicha especificación de diseño.

Respecto al control “*Gain Scheduling DMC*”, se recomienda ajustar los parámetros manualmente, introduciendo en primer lugar los horizontes de predicción y control y, posteriormente, variando los valores de λ en función de la agresividad deseada para cada punto de operación.

ANEXO B. RESULTADOS EXPERIMENTALES DE ALGORITMOS DE IDENTIFICACIÓN

1. Experimento de identificación paramétrica por mínimos cuadrados

Con el objetivo de evaluar la efectividad del método de identificación paramétrica por mínimos cuadrados, se ha llevado a cabo un ensayo experimental simulado. Los resultados se discuten en el apartado 4.2.

1.1. Descripción del experimento

Se realiza una simulación del proceso en lazo abierto, de acuerdo a unos cambios en la variable manipulable previamente definidos y se registran todos los datos de simulación. Posteriormente, se construye la base de datos de entrada y se ejecuta el algoritmo de identificación paramétrica por el método de mínimos cuadrados. La función de transferencia resultante se simula, se transforman los resultados a variables absolutas y se comparan en una gráfica con la respuesta del proceso.

1.2. Configuración de la simulación

La configuración de la simulación y las condiciones iniciales para la identificación se muestran en la Tabla B1.

Los parámetros calculados en los algoritmos de identificación corresponden a los de la siguiente función de transferencia en el dominio de z :

$$G(z) = \frac{bz^{-1}}{1 + az^{-1}}$$

donde a es el “parámetro 1” y b , “el parámetro 2” del vector paramétrico.

Los parámetros equivalentes en tiempo continuo (ganancia y constante de tiempo) han sido obtenidas mediante la siguiente relación:

$$G(s) = \frac{\frac{b}{1+a}}{\frac{\Delta t_k}{|-\ln(-a)|}s + 1}$$

El tiempo muerto se considera como información conocida a priori en el algoritmo de identificación y deberá aportarse como argumento de entrada al mismo (el conocimiento de τ_{mp} es fácilmente aproximable mediante la observación de los resultados experimentales).

Tabla B1. Configuración de la simulación en el experimento de identificación paramétrica mediante LS (I)

VARIABLE	DESCRIPCIÓN	VALOR
MI	Método de integración numérica: <i>MI</i> = 1: Runge-Kutta orden 4 <i>MI</i> = 2: Euler	1
T0	Temperatura inicial del reactor (en °C)	87,81
cA0	Concentración inicial de reactivo a la salida (en mol/L)	0,04
Tc0	Temperatura inicial de refrigeración (en °C)	82
qc0	Caudal inicial de refrigeración (L/min)	90
valv0	Apertura inicial de la válvula de refrigeración (en %)	64,29
n	Simulación con ruido: <i>n</i> = 0: Simulación sin ruido <i>n</i> = 1: Simulación con ruido	0
Mode.ST	Tipo de sensor: <i>Mode.ST</i> = 1: Señal de [0 – 10] V <i>Mode.ST</i> = 2: Señal de [4 – 20] mA	1
Mode.Cout	Señal de salida del controlador: <i>Mode.Cout</i> = 1: Señal de [0 – 10] V <i>Mode.Cout</i> = 2: Señal de [4 – 20] mA	1
tstp	Vector de instantes de aplicación de cambios en la variable manipulable (en minutos)	[1; 10; 20; 30; 40]
stp	Vector de valores de la variable manipulable (en %)	[69,29; 59,29; 69,29; 59,29; 69,29]
Ts	Tiempo de simulación (en minutos)	50

1.3. Resultados

Los resultados obtenidos aparecen en la Tabla B2.

En la Figura B1 se muestra gráficamente la superposición de la respuesta del proceso simulada y la salida estimada con los parámetros identificados. La salida estimada corresponde a los datos obtenidos tras multiplicar el regresor por el vector paramétrico obtenido de la identificación, mientras que la salida del proceso identificado corresponde a la simulación del modelo lineal en tiempo continuo para las mismas condiciones de entrada y tiempo de simulación.

Tabla B2. Resultados del experimento de identificación paramétrica LS

VARIABLE	DESCRIPCIÓN	VALOR
a	Valor del parámetro 1 obtenido mediante LS (adimensional)	-0,98
b	Valor del parámetro 2 obtenido mediante LS (adimensional)	-0,01
K	Valor de la ganancia estática obtenida mediante el método LS (en $\Delta^{\circ}\text{C}/\Delta\%$)	-0,54
τ	Valor de la constante de tiempo obtenida mediante el método LS (en minutos)	1,05
em	Promedio del valor absoluto del error de estimación (en $\Delta^{\circ}\text{C}$)	0,01

RESULTADOS DE LA IDENTIFICACIÓN PARAMÉTRICA TEMPERATURA DEL REACTOR CSTR ENCAMISADO

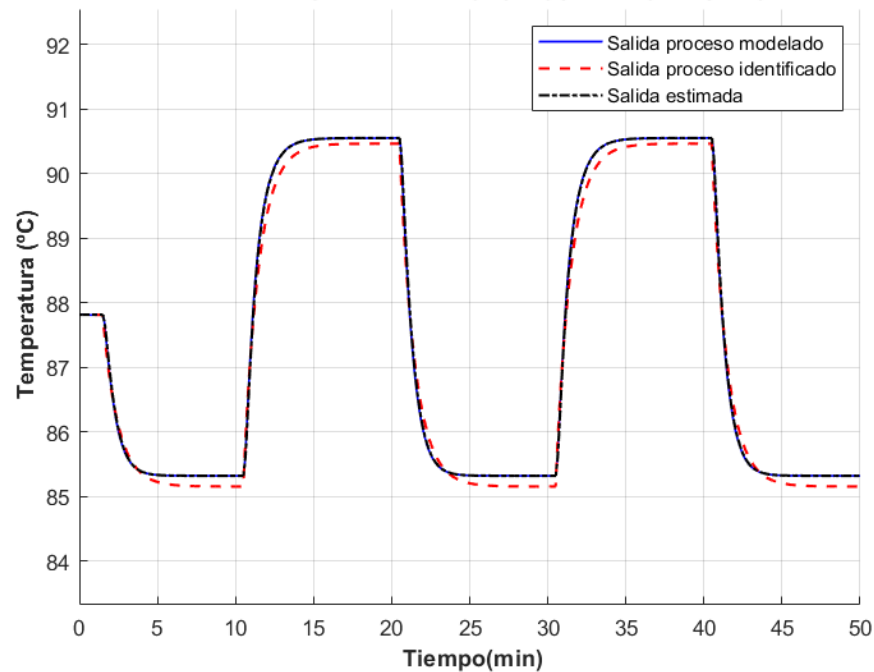


Figura B1. Resultados de la identificación paramétrica LS

2. Experimento de identificación paramétrica por mínimos cuadrados recursivos

Al igual que en el apartado 1, con el objetivo de evaluar la efectividad del método de identificación paramétrica por mínimos cuadrados recursivos, se ha llevado a cabo un ensayo experimental simulado. Los resultados se discuten en el apartado 6.2.

2.1. Descripción del experimento

Se simula el proceso controlado por un PI, para un tren de referencias previamente definido. En la simulación se incluye el algoritmo de identificación paramétrica por el método de mínimos cuadrados recursivos, que es accionado por el sistema de detección de excitación. Se registran los valores de los parámetros identificados y se comparan con los resultados obtenidos tras identificar fuera de línea el proceso mediante mínimos cuadrados.

2.2. Configuración de la simulación

La configuración de la simulación, así como las condiciones iniciales para la identificación recursiva en línea se muestran en la Tabla B3.

Tabla B3. Configuración de la simulación en el experimento de identificación paramétrica RLS

VARIABLE	DESCRIPCIÓN	VALOR
MI	Método de integración numérica: <i>MI</i> = 1: Runge-Kutta orden 4 <i>MI</i> = 2: Euler	1
T0	Temperatura inicial del reactor (en °C)	87,81
cA0	Concentración inicial de reactivo a la salida (en mol/L)	0,04
Tc0	Temperatura inicial de refrigeración (en °C)	82
qc0	Caudal inicial de refrigeración (L/min)	90
valv0	Abertura inicial de la válvula de refrigeración (en %)	64,29
Kp	Constante proporcional del controlador PI (en $\Delta^{\circ}\text{C}/\Delta\%$)	-1,5
Ti	Tiempo integral del controlador PI (en minutos)	0,89
theta0	Vector paramétrico inicial	[0; 0]
P	Matriz de covarianza inicial	$\begin{bmatrix} 100 & 10^4 \\ 10^4 & 100 \end{bmatrix}$
olvido	Factor de olvido (adimensional)	0,99
n	Simulación con ruido: <i>n</i> = 0: Simulación sin ruido <i>n</i> = 1: Simulación con ruido	0
Mode.ST	Tipo de sensor: <i>Mode.ST</i> = 1: Señal de [0 – 10] V <i>Mode.ST</i> = 2: Señal de [4 – 20] mA	1
Mode.Cout	Señal de salida del controlador: <i>Mode.Cout</i> = 1: Señal de [0 – 10] V <i>Mode.Cout</i> = 2: Señal de [4 – 20] mA	1
tref	Vector de instantes de aplicación de las referencias (en minutos)	[1; 10; 20; 30; 40]
ref	Vector de referencias (en °C)	[92,81; 82,81; 92,81; 355,96; 92,81]
Ts	Tiempo de simulación (en minutos)	50

La configuración de la simulación y condiciones iniciales para la identificación fuera de línea se muestran en la Tabla B4.

Tabla B4. Configuración de la simulación en el experimento de identificación paramétrica mediante LS (II)

VARIABLE	DESCRIPCIÓN	VALOR
MI	Método de integración numérica: <i>MI</i> = 1: Runge-Kutta orden 4 <i>MI</i> = 2: Euler	1
T0	Temperatura inicial del reactor (en °C)	87,81
cA0	Concentración inicial de reactivo a la salida (en mol/L)	0,042
Tc0	Temperatura inicial de refrigeración (en °C)	82
qc0	Caudal inicial de refrigeración (L/min)	90
valv0	Apertura inicial de la válvula de refrigeración (en %)	64,29
n	Simulación con ruido: <i>n</i> = 0: Simulación sin ruido <i>n</i> = 1: Simulación con ruido	0
Mode.ST	Tipo de sensor: <i>Mode.ST</i> = 1: Señal de [0 – 10] V <i>Mode.ST</i> = 2: Señal de [4 – 20] mA	1
Mode.Cout	Señal de salida del controlador: <i>Mode.Cout</i> = 1: Señal de [0 – 10] V <i>Mode.Cout</i> = 2: Señal de [4 – 20] mA	1
tstp	Vector de instantes de aplicación de cambios en la variable manipulable (en minutos)	[1; 10; 20; 30; 40]
stp	Vector de valores de la variable manipulable (en %)	[74,29; 54,29; 74,29; 54,29; 74,29]
Ts	Tiempo de simulación (en minutos)	50

2.3. Resultados

Los resultados obtenidos están recogidos en la Tabla B5 y se discuten en el apartado 6.2.

Tabla B5. Resultados de la identificación paramétrica RLS

VARIABLE	DESCRIPCIÓN	VALOR
<i>am_RLS</i>	Valor medio del parámetro 1 obtenido mediante el método RLS, una vez eliminados los datos de estimación inicial (adimensional)	-0,98
<i>bm_RLS</i>	Valor medio del parámetro 2 obtenido mediante el método RLS, una vez eliminados los datos de estimación inicial (adimensional)	-0,01
<i>Km_RLS</i>	Valor medio de la ganancia estacionaria obtenida mediante el método RLS, una vez eliminados los datos de estimación inicial (en $\Delta^{\circ}\text{C}/\Delta\%$)	-0,55
<i>taum_RLS</i>	Valor medio de la constante de tiempo obtenida mediante el método RLS, una vez eliminados los datos de estimación inicial (minutos)	0,92
<i>em_RLS</i>	Promedio del valor absoluto del error de estimación mediante RLS, una vez eliminados los datos de estimación inicial (en $\Delta^{\circ}\text{C}$)	$5 \cdot 10^{-3}$
<i>a_LS</i>	Valor del parámetro 1 obtenido mediante LS (adimensional)	-0,98
<i>b_LS</i>	Valor del parámetro 2 obtenido mediante LS (adimensional)	-0,01
<i>K_LS</i>	Valor de la ganancia estacionaria obtenida mediante el método LS (en $\Delta^{\circ}\text{C}/\Delta\%$)	-0,54
<i>tau_LS</i>	Valor de la constante de tiempo obtenida mediante el método LS (minutos)	1,08
<i>em_LS</i>	Promedio del valor absoluto del error de estimación mediante LS (en $\Delta^{\circ}\text{C}$)	0,01

La Figura B2 ilustra los resultados del experimento.

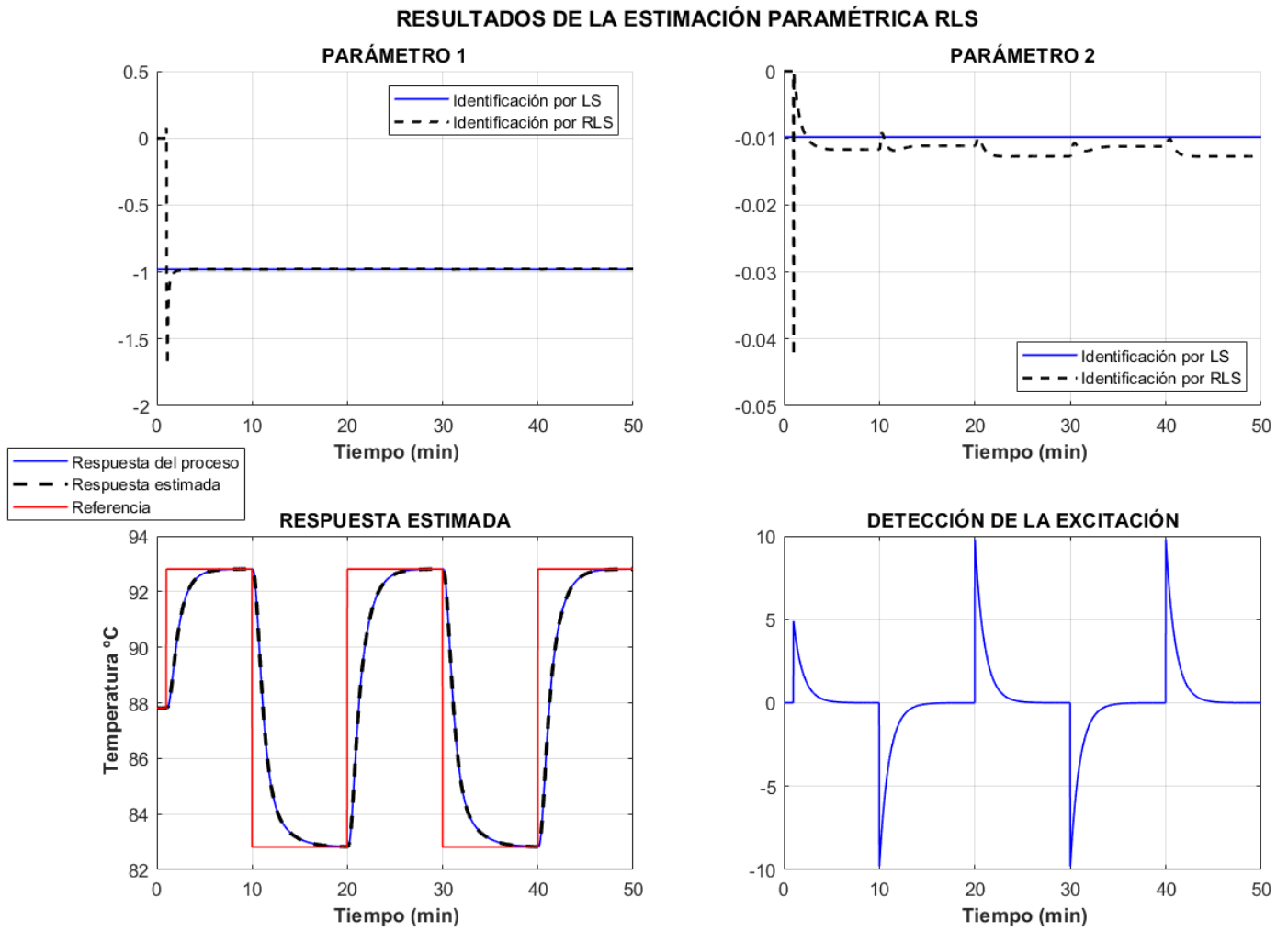


Figura B2. Resultados de la identificación paramétrica RLS y comparación con el método LS

