



UNIVERSIDAD DE CHILE

FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS

DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

**METODOLOGÍA DE IDENTIFICACIÓN DIFUSA BASADA EN EL  
ESTUDIO DE CONTROLABILIDAD DE SISTEMAS DINÁMICOS**

**MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL  
ELÉCTRICO**

**TESIS PARA OPTAR AL GRADO DE MAGÍSTER EN CIENCIAS DE  
LA INGENIERÍA, MENCIÓN ELÉCTRICA**

**LEONEL ALEJANDRO GUTIÉRREZ ÁLVAREZ**

PROFESORA GUÍA:  
DORIS SÁEZ HUEICHAPAN

PROFESOR CO-GUÍA:  
FELIPE VALENCIA ARROYAVE

MIEMBROS DE LA COMISIÓN:  
MARCOS ORCHARD CONCHA  
JUAN YUZ EISSMANN

SANTIAGO DE CHILE  
2016

**RESUMEN DE LA TESIS PARA OPTAR  
AL TÍTULO DE:** Ingeniero Civil Eléctrico y  
Grado de Magíster en Ciencias de la Ingeniería,  
Mención Eléctrica  
**POR:** Leonel Alejandro Gutiérrez Álvarez  
**FECHA:** 08 de Abril del 2016  
**PROF. GUÍA:** Doris Sáez Hueichapan

## **Metodología de Identificación Difusa Basada en el Estudio de Controlabilidad de Sistemas Dinámicos**

Hoy en día la tarea de obtener un modelo de un sistema se ha vuelto fundamental en la teoría de control, y es en dicho ámbito donde los modelos difusos de Takagi-Sugeno han destacado debido a su capacidad de aproximar dinámicas no lineales. No obstante, para el diseño de controladores no basta sólo predecir el comportamiento de una planta, sino que el modelo debe heredar la propiedad de controlabilidad del sistema original. Dicha propiedad puede entenderse como la capacidad de un sistema para alcanzar un estado deseado a partir de cualquier estado inicial, y por lo tanto es una característica fundamental para el diseño de estrategias de control, debido a que si el modelo no posee dicha propiedad, es posible que las acciones de control inferidas sean erróneas, afectando el rendimiento en lazo cerrado del sistema. Luego, el principal objetivo de este trabajo es diseñar métodos de identificación difusa que permitan obtener modelos controlables (o en su defecto que aumenten el espacio de controlabilidad), de tal forma de poder diseñar adecuadamente estrategias de control. Para ello se plantean dos enfoques: incluir restricciones de controlabilidad en el procedimiento de identificación, e incluir términos de orden superior en los consecuentes del modelo difuso.

La metodología a seguir para el diseño del algoritmo de identificación con restricciones de controlabilidad se basa en analizar la aproximación de primer orden del modelo difuso en cada punto de operación, y a partir de dicha aproximación imponer las restricciones de controlabilidad al momento de identificar los parámetros de las consecuencias del modelo difuso. Por otro lado, en la metodología para obtener modelos difusos con términos de orden superior se consideran aproximaciones de segundo orden en cada consecuente del modelo. Una vez obtenidas las representaciones matemáticas controlables se diseñan estrategias de control predictivo difuso, las cuales son aplicadas a distintas plantas de estudio para validar las metodologías propuestas.

Como resultados de este trabajo, se logra implementar exitosamente un procedimiento de identificación difusa capaz de generar modelos controlables, los cuales permiten un diseño adecuado de controladores predictivos, mejorando la respuesta de los sistemas en comparación a utilizar un modelo no controlable. A su vez, el principal aporte de esta tesis se remonta al planteamiento de métodos de identificación difusa que no han sido reportados en la literatura, con las cuales se pueden obtener modelos controlables, y cuya metodología puede ser extendida a otro tipo de restricciones, con el fin de garantizar otras propiedades del sistema original.

---

*A mis padres,  
hermanos,  
y a todos aquellos cercanos que ya no están.*

## Agradecimientos

Quisiera aprovechar este espacio para expresar mi gratitud a todos aquellos que me han ayudado en esta larga y difícil etapa. Primero que todo, agradecer a mis padres, quienes siempre me han brindado su cariño, confianza y apoyo para cumplir cada desafío que me he planteado. Sin duda han sido un pilar fundamental, todo lo que soy es gracias a ustedes; gracias por los valores inculcados y la educación brindada, gracias por su esfuerzo para darme siempre lo mejor. En especial agradecer a Lili, quien siempre se ha preocupado por mí, desde despertarme cuando me quedaba dormido, hasta cuando me llevaba un pan en los momentos en que no tenía tiempo para comer porque tenía que terminar los trabajos. Sin ti esto hubiese sido mucho más difícil.

A mis hermanos, quienes siempre han estado a mi lado, teniendo que aguantarme durante tantas jornadas de estrés, se llevaron la peor parte jaja. Gracias por compartir tantas risas y experiencias, gracias por su apoyo y ayuda.

Agradezco a los amigos que me ha dado la vida, a aquellos que conozco desde que tengo memoria, hasta los que conocí durante la época escolar. Gracias por su apoyo, por su motivación en los momentos difíciles, y por su cariño durante tantos años. Espero siempre contar con su amistad y seguir reviviendo y creando nuevas anécdotas.

A todas aquellas personas que conocí en la universidad. A ese grupo que se formó desde los primeros días de clases, y que hasta el día de hoy vemos el crecimiento de cada uno; a la gente que conocí en el departamento de ingeniería eléctrica, con quienes he compartido el día a día durante los últimos años. Muchas gracias por tantas jornadas...no sé si de estudio o leseo, por esas jornadas de cansancio y estrés, donde no podíamos dejar de mentir en cada trabajo que hacíamos, pero por sobre todo, gracias por cada uno de esos momentos para compartir y reír. Muchas gracias por su amistad y ayuda, por hacer que los recuerdos del pasar por la universidad sólo traigan sonrisas. No cabe duda de que serán grandes en cada área en que se desempeñen.

También quiero agradecer a los profesores que participaron en mi formación, en especial a la profe Doris y Felipe. Gracias por su confianza, por todas sus enseñanzas, por su tiempo y respaldo en cada momento. También agradecer por sus comentarios y críticas, los cuales me ayudaron a crecer como profesional. A su vez, agradecer a la Fundación Moisés Mellado, Universidad de Chile, CONICYT, Instituto Sistemas Complejos de Ingeniería ICM: P-05-004-F y Solar Energy Research Center (SERC) CONICYT: FBO16, CONICYT/FONDAP/15110019; por los aportes económicos otorgados durante mi estadía en la universidad.

Por último, quiero agradecer a Paula, por ser la persona que me motiva a ser mejor cada día, por estar siempre a mi lado y por apoyarme aun cuando no tendría que apoyarme. Gracias por regalarme tantas sonrisas y por llenar cada uno de mis días de felicidad.

# Tabla de Contenido

1.	Introducción .....	1
2.	Identificación de Sistemas Difusos .....	9
2.1.	Modelos Difusos de Takagi-Sugeno .....	15
2.2.	Diseño de Experimento de Identificación .....	17
2.3.	Selección de Variables Relevantes.....	23
2.3.1.	Análisis de Sensibilidad.....	24
2.3.2.	Método Heurístico.....	24
2.3.3.	Estudio de Correlación/Autocorrelación .....	25
2.4.	Optimización de la Estructura.....	26
2.5.	Identificación de Parámetros.....	28
2.5.1.	Parámetros de las Premisas .....	29
2.5.2.	Parámetros de las Consecuencias .....	33
3.	Controlabilidad .....	36
3.1.	Controlabilidad de Sistemas Lineales .....	37
3.2.	Controlabilidad de Sistemas Difusos.....	40
3.3.	Controlabilidad de Aproximaciones de Orden Superior.....	43
4.	Identificación Difusa con Restricciones de Controlabilidad .....	46
4.1.	Algoritmo de Identificación .....	46
4.2.	Casos de Estudio .....	52
4.2.1.	Reactor de Estanque Continuamente Agitado.....	52
4.2.1.1.	Identificación del CSTR.....	52
4.2.1.2.	Control Predictivo Difuso del CSTR.....	77
4.2.2.	Sistema no Lineal Teóricamente Controlable.....	82
4.2.2.1.	Identificación del SNLC.....	84
4.2.2.2.	Control Predictivo Difuso del SNLC.....	99
5.	Identificación Difusa con Términos de Orden Superior.....	104
5.1.	Aproximaciones de Segundo Orden .....	104
5.2.	Modelos Difusos de Takagi-Sugeno con Términos de Segundo Orden .....	108
5.3.	Caso de Estudio: Reactor de Estanque Continuamente Agitado .....	109
5.3.1.	Identificación del CSTR.....	109
5.3.2.	Control Predictivo Difuso del CSTR.....	115
6.	Conclusiones .....	120

6.1.	Aportes de la Tesis .....	123
6.2.	Trabajo Futuro .....	124
6.3.	Publicaciones .....	126
7.	Bibliografía.....	127
8.	Anexos.....	135
8.1.	Anexo A: Algoritmos de Optimización .....	135
8.1.1.	Método Quasi-Newton .....	135
8.1.2.	Método Nelder-Mead .....	135
8.1.3.	Optimización por Enjambre de Partículas.....	136
8.2.	Anexo B: Control Predictivo Difuso.....	139
8.2.1.	Algoritmos de Solución .....	143
8.2.1.1.	Programación Cuadrática Secuencial.....	144
8.2.1.2.	Método de Punto Interior .....	144
8.2.2.	Sintonización de Controladores .....	145
8.2.3.	Consideraciones en el Diseño de Controlador Predictivo .....	147
8.2.3.1.	Horizonte de Predicción .....	147
8.2.3.2.	Algoritmo de Solución .....	148
8.2.3.3.	Tiempo de Muestreo y Tiempo de Aplicación .....	149
8.3.	Anexo C: Resultados del Proceso de Identificación .....	152
8.3.1.	Modelos Difusos Controlables de 2do Orden CSTR .....	152
8.3.2.	Modelo Difuso no Controlable CSTR.....	154
8.4.	Anexo D: Formas Canónicas de Modelos Lineales .....	155
8.4.1.	Forma Canónica Controlable .....	155
8.4.2.	Forma Canónica Observable .....	157
8.5.	Anexo E: Estabilidad de Modelos Difusos.....	159

# Índice de Tablas

Tabla 2.1: Coeficientes para Generar PRBS de Largo Máximo.....	19
Tabla 4.1: Parámetros PSO para Identificación del CSTR. ....	58
Tabla 4.2: Resultados Identificación del CSTR en Lazo Abierto a un Paso. ....	59
Tabla 4.3: Resultados Identificación del CSTR en Lazo Abierto a $N$ Pasos. ....	59
Tabla 4.4: Estructura Mejor Modelo Identificación a un Paso del CSTR con Datos en Lazo Abierto.....	62
Tabla 4.5: Estructura Mejor Modelo Identificación a $N$ Pasos del CSTR con Datos en Lazo Abierto.....	66
Tabla 4.6: Resultados Identificación del CSTR en Lazo Cerrado a un Paso.....	69
Tabla 4.7: Resultados Identificación del CSTR en Lazo Cerrado a $N$ Pasos.....	69
Tabla 4.8: Estructura Mejor Modelo Identificación a un Paso del CSTR con Datos en Lazo Cerrado.....	72
Tabla 4.9: Estructura Mejor Modelo Identificación a $N$ Pasos del CSTR con Datos en Lazo Cerrado.....	76
Tabla 4.10: Resultados Control Predictivo Difuso del CSTR con Modelos de Predicción Obtenidos con Datos en Lazo Abierto. ....	78
Tabla 4.11: Resultados Control Predictivo Difuso del CSTR con Modelos de Predicción Obtenidos con Datos en Lazo Cerrado. ....	80
Tabla 4.12: Resultados Identificación del SNLC en Lazo Cerrado a un Paso.....	89
Tabla 4.13: Resultados Identificación del SNLC en Lazo Cerrado a $N$ Pasos.....	90
Tabla 4.14: Estructura Mejor Modelo Identificación a un Paso del SNLC con Datos en Lazo Cerrado.....	92
Tabla 4.15: Estructura Mejor Modelo Identificación a $N$ Pasos del SNLC con Datos en Lazo Cerrado.....	96
Tabla 4.16: Resultados Control Predictivo Difuso del SNLC con Modelos de Predicción Obtenidos con Datos en Lazo Cerrado. ....	100
Tabla 5.1: Resultados Identificación Difusa Con Términos de 1er y 2do Orden. ....	110
Tabla 5.2: Parámetros Mejor Modelo Identificación Difusa con Términos de Segundo Orden del CSTR: Regla 1 y Regla 2. ....	112
Tabla 5.3: Parámetros Mejor Modelo Identificación Difusa con Términos de Segundo Orden del CSTR: Regla 3 y Regla 4. ....	113
Tabla 5.4: Resultados Control Predictivo Difuso del CSTR con Modelos de Predicción Difusos con Términos de 1er y 2do Orden. ....	115
Tabla 8.1: Tiempos de Cómputo Variando $N_p$ .....	148
Tabla 8.2: Parámetros PSO para Controlador Predictivo Difuso. ....	148
Tabla 8.3: Tiempos de Cómputo Variando Algoritmos de Optimización.....	149
Tabla 8.4: Error en Estado Estacionario Usando Control Predictivo Difuso en el CSTR Usando Distintos Tiempos de Aplicación.....	151
Tabla 8.5: Resultados Identificación en Lazo Abierto Usando $n_y = 2, n_u = 1$ , Método QN/NM/PSO, CI's Conocidas e Identificación a un Paso. ....	152
Tabla 8.6: Resultados Identificación en Lazo Abierto Usando $n_y = 2, n_u = 1$ , Método NM, CI's Conocidas e Identificación a $N$ Pasos. ....	152

---

Tabla 8.7: Resultados Identificación en Lazo Cerrado Usando $n_y = 2, n_u = 1$ , Método NM, CI's Conocidas e Identificación a un Paso. ....	153
Tabla 8.8: Resultados Identificación en Lazo Cerrado Usando $n_y = 2, n_u = 1$ , Método QN-PSO, CI's Conocidas e Identificación a $N$ Pasos. ....	153



# Índice de Figuras

Figura 1.1: Diagrama de Bloques CSTR.....	2
Figura 1.2: Respuesta del CSTR Usando $u(t) = t + 300$ . ....	3
Figura 1.3: Resultado CSTR con MPC con Modelo de Predicción de 1er Orden. ....	4
Figura 2.1: Esquema de un Sistema de Lógica Difusa. ....	10
Figura 2.2: Procedimiento de Identificación de Sistemas Difusos. ....	16
Figura 2.3: PRBS [26].....	18
Figura 2.4: Espectro en Frecuencia de PRBS [26].....	20
Figura 2.5: Esquema de Identificación en Lazo Abierto.....	21
Figura 2.6: Esquema de Identificación en Lazo Cerrado.....	21
Figura 2.7: Ejemplo de APRBS con Amplitud entre -4 y +4.....	22
Figura 2.8: Principio de Parsimonia [99]. ....	27
Figura 2.9: Combinación de <i>Clusters</i> Compatibles [10]. ....	28
Figura 2.10: Curva Descrita por $(z - v)^T F^{-1}(z - v) = 1$ [10]. ....	31
Figura 3.1: Izquierda: Sistema Controlable. Derecha: Sistema Localmente Controlable [43]. ....	37
Figura 4.1: Respuesta al Escalón del CSTR. ....	53
Figura 4.2: Diagrama de Bloques Identificación CSTR en Lazo Abierto.....	54
Figura 4.3: Diagrama de Bloques Identificación CSTR en Lazo Cerrado.....	54
Figura 4.4: Datos de Identificación del CSTR en Lazo Abierto. a) Entrada $u(t)$ [° K] (APRBS) vs Tiempo [s]. b) Datos de Salida $y(t)$ vs Tiempo [s]. ....	55
Figura 4.5: Datos de Identificación del CSTR en Lazo Cerrado. a) Entrada $u(t)$ [° K] vs Tiempo [s]. b) APRBS (Referencia) y Datos de Salida $y(t)$ vs Tiempo [s].....	56
Figura 4.6: Gráfico Función de Autocorrelación Parcial CSTR vs Número de Retardos. a) Datos de Identificación en Lazo Abierto. b) Datos de Identificación en Lazo Cerrado.....	57
Figura 4.7: Resultado Mejor Modelo Identificación a un Paso del CSTR con Datos en Lazo Abierto. Arriba: Conjunto de Entrenamiento. Medio: Conjunto de Validación. Abajo: Conjunto de Prueba.....	61
Figura 4.8: Conjuntos Difusos de los Regresores $y(k - j)$ para el Mejor Modelo Identificación a un Paso del CSTR con Datos en Lazo Abierto. ....	63
Figura 4.9: Conjuntos Difusos de los Regresores $u(k - j)$ para el Mejor Modelo Identificación a un Paso del CSTR con Datos en Lazo Abierto. ....	64
Figura 4.10: Resultado Mejor Modelo Identificación a $N$ Pasos del CSTR con Datos en Lazo Abierto. Arriba: Conjunto de Entrenamiento. Medio: Conjunto de Validación. Abajo: Conjunto de Prueba.....	65
Figura 4.11: Conjuntos Difusos de los Regresores $y(k - j)$ para el Mejor Modelo Identificación a $N$ Pasos del CSTR con Datos en Lazo Abierto.....	67
Figura 4.12: Conjuntos Difusos de los Regresores $u(k - j)$ para el Mejor Modelo Identificación a $N$ Pasos del CSTR con Datos en Lazo Abierto.....	68
Figura 4.13: Resultado Mejor Modelo Identificación a un Paso del CSTR con Datos en Lazo Cerrado. Arriba: Conjunto de Entrenamiento. Medio: Conjunto de Validación. Abajo: Conjunto de Prueba.....	71

Figura 4.14: Conjuntos Difusos de los Regresores $y(k - j)$ para el Mejor Modelo Identificación a un Paso del CSTR con Datos en Lazo Cerrado. ....	73
Figura 4.15: Conjuntos Difusos de los Regresores $u(k - j)$ para el Mejor Modelo Identificación a un Paso del CSTR con Datos en Lazo Cerrado. ....	74
Figura 4.16: Resultado Mejor Modelo Identificación a $N$ Pasos del CSTR con Datos en Lazo Cerrado. Arriba: Conjunto de Entrenamiento. Medio: Conjunto de Validación. Abajo: Conjunto de Prueba. ....	75
Figura 4.17: Resultados Control Predictivo Difuso del CSTR Usando Modelos de Predicción Obtenidos con Datos en Lazo Abierto e Identificación a un Paso. a) Salida $y(t)$ vs Tiempo [s]. b) Entrada $u(t)$ [K] vs Tiempo [s]. ....	79
Figura 4.18: Resultados Control Predictivo Difuso del CSTR Usando Modelos de Predicción Obtenidos con Datos en Lazo Cerrado e Identificación a un Paso (NM). a) Salida $y(t)$ vs Tiempo [s]. b) Entrada $u(t)$ [K] vs Tiempo [s]. ....	81
Figura 4.19: Simulación no Forzada SNLC. ....	84
Figura 4.20: Respuesta al Escalón SNLC. ....	85
Figura 4.21: Comparación Señal de Control $u(t)$ con y sin Perturbación $p(t)$ . ....	87
Figura 4.22: Datos de Identificación del SNLC en Lazo Cerrado. a) Entrada $u(t)$ vs Tiempo [s]. b) APRBS y Datos de Salida $y(t)$ vs Tiempo [s]. ....	88
Figura 4.23: Función de Autocorrelación Parcial SNLC. ....	89
Figura 4.24: Resultado Mejor Modelo Identificación a un Paso del SNLC. Arriba: Conjunto de Entrenamiento. Medio: Conjunto de Validación. Abajo: Conjunto de Prueba. ....	91
Figura 4.25: Conjuntos Difusos de los Regresores $y(k - j)$ para el Mejor Modelo Identificación a un Paso del SNLC con Datos en Lazo Cerrado. ....	93
Figura 4.26: Conjuntos Difusos de los Regresores $u(k - j)$ para el Mejor Modelo Identificación a un Paso del SNLC con Datos en Lazo Cerrado. ....	94
Figura 4.27: Resultado Mejor Modelo Identificación a $N$ Pasos del SNLC. Arriba: Conjunto de Entrenamiento. Medio: Conjunto de Validación. Abajo: Conjunto de Prueba. ....	95
Figura 4.28: Conjuntos Difusos de los Regresores $y(k - j)$ para el Mejor Modelo Identificación a $N$ Pasos del SNLC con Datos en Lazo Cerrado. ....	97
Figura 4.29: Conjuntos Difusos de los Regresores $u(k - j)$ para el Mejor Modelo Identificación a $N$ Pasos del SNLC con Datos en Lazo Cerrado. ....	98
Figura 4.30: Resultados Control Predictivo Difuso del SNLC Usando Modelos de Predicción Obtenidos con Datos en Lazo Cerrado e Identificación a $N$ Pasos (QN). a) Salida $y(t)$ vs Tiempo [s]. b) Entrada $u(t)$ vs Tiempo [s]. ....	101
Figura 5.1: Resultado Control Predictivo Utilizando Modelos de Predicción de 1er Orden y de 2do Orden. ....	107
Figura 5.2: Resultado Mejor Modelo Identificación Difusa con Términos de Segundo Orden del CSTR. Arriba: Conjunto de Entrenamiento. Medio: Conjunto de Validación. Abajo: Conjunto de Prueba. ....	111
Figura 5.3: Conjuntos Difusos de Algunas Premisas del el Mejor Modelo Identificación Difusa con Términos de Segundo Orden del CSTR. ....	114

Figura 5.4: Resultados Control Predictivo Difuso del CSTR Usando Modelo de Predicción Difuso con Términos de 2do Orden. a) Salida  $y(t)$  vs Tiempo [s]. b) Entrada  $u(t)$  [K] vs Tiempo [s]. .....117

Figura 5.5: Resultados Control Predictivo Difuso del CSTR Frente a Perturbaciones Usando Modelo de Predicción con Aproximaciones de 2do Orden y Modelo de Predicción Difuso con Términos de 2do Orden. a) Salida  $y(t)$  vs Tiempo [s]. b) Entrada  $u(t)$  [K] vs Tiempo [s]. ..... 118

Figura 8.1: Puntos Utilizados en el Algoritmo Nelder-Mead [116]..... 136

Figura 8.2: Resultados Control Predictivo Difuso del CSTR Usando Distintos Tiempos de Aplicación. a) Salida  $y(t)$  vs Tiempo [s]. b) Entrada  $u(t)$  [K] vs Tiempo [s]. ..... 150

Figura 8.3: Resultado Modelo no Controlable Identificación a un Paso con Datos en Lazo Cerrado. Arriba: Conjunto de Entrenamiento. Medio: Conjunto de Validación. Abajo: Conjunto de Prueba..... 154

# 1. Introducción

Hoy en día el desarrollo de modelos matemáticos para la representación de procesos se ha transformado en una potente herramienta, ya sea para analizar, diseñar, predecir, entre otras tareas relacionadas con ingeniería. Para obtener dichos modelos existen diversos enfoques propuestos en la literatura, orientados a generar tanto modelos lineales como no lineales. En particular, los modelos no lineales permiten representar una mayor cantidad de condiciones de operación en comparación a los modelos lineales, por lo que ofrecen una mejor adaptación a los comportamientos de los sistemas. Dentro de la clase de modelos no lineales destacan los modelos difusos de Takagi-Sugeno (TS) [107], debido a su capacidad para aproximar diversas dinámicas, siendo aptos para representar con cierto grado de exactitud las no linealidades que se encuentran generalmente en los procesos de la industria.

Si bien se ha demostrado que los modelos difusos logran representar plantas con fuertes no linealidades, siendo catalogados como aproximadores universales y presentándose como una potente herramienta para la identificación de sistemas [34][114]; obtener un modelo capaz de predecir el comportamiento de una planta puede no ser suficiente desde el punto de vista de la teoría de control, ya que dicho modelo además debe ser capaz de reproducir las principales propiedades y características de la planta original. Dentro de dichas propiedades se encuentran la estabilidad, la observabilidad y la controlabilidad. En particular este trabajo aborda este último concepto para los modelos difusos TS.

Se dice que un sistema es completamente controlable en un intervalo de tiempo  $[t_0, t_1]$  si para cualquier estado inicial  $x_0$  en  $t_0$  y cualquier estado final  $x_1$  en  $t_1$ , existe un conjunto de acciones de control admisibles  $u(t)$  definido en  $[t_0, t_1]$  tal que  $x(t_1) = x_1$ , siendo  $x(t)$  la trayectoria de estados de la planta [46][95]. Es decir, la controlabilidad puede entenderse como la capacidad de un sistema para alcanzar un estado deseado a partir de cualquier estado inicial, en un tiempo finito; por lo tanto es una característica fundamental para el diseño de estrategias de control. Si bien la mayor parte de los procesos físicos son controlables, los modelos matemáticos que los representan puede que no posean dicha característica, por lo que es necesario estudiar las condiciones donde se cumple esta propiedad [78]. En particular, si el modelo no es completamente controlable la acción de control inferida a partir de éste puede llevar a resultados erróneos.

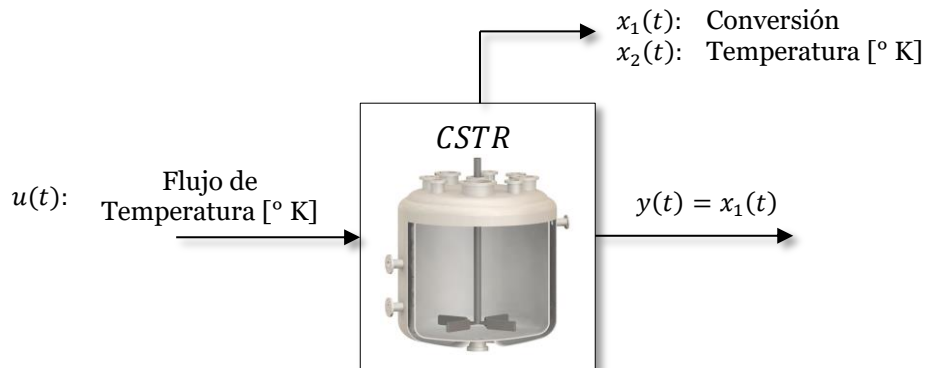
A modo de ilustrar lo anteriormente expuesto, considérese el modelo de un reactor de estanque continuamente agitado (*Continuous Stirred Tank Reactor*, CSTR), en donde se lleva a cabo una reacción exotérmica reversible de primer orden representada como  $A \rightleftharpoons B$ , con  $K_1(t)$  la constante de velocidad para la reacción  $A \rightarrow B$ , y  $K_2(t)$  la constante de velocidad para la reacción  $A \leftarrow B$ . El modelo en el espacio de estados que describe el comportamiento del CSTR es el siguiente [79]:

$$\frac{dx_1(t)}{dt} = -x_1(t) + K_1(t) \cdot (1 - x_1(t)) - K_2(t)x_2(t), \quad (1.1)$$

$$\frac{dx_2(t)}{dt} = u(t) - x_2(t) + 5 \cdot [K_1(t) \cdot (1 - x_1(t)) - K_2(t)x_2(t)], \quad (1.2)$$

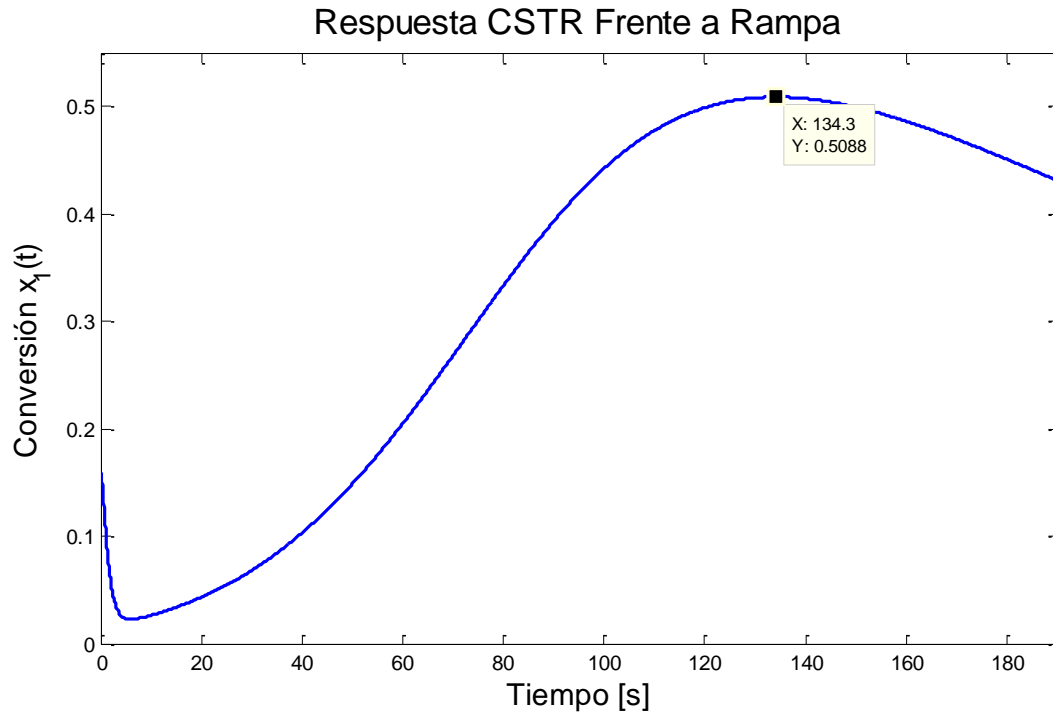
$$K_1(t) = 3 \times 10^5 \cdot \exp\left(-\frac{5000}{x_2(t)}\right); \quad K_2(t) = 6 \times 10^7 \cdot \exp\left(-\frac{7500}{x_2(t)}\right), \quad (1.3)$$

donde  $x(t) = [x_1(t), x_2(t)]^T$  es el estado del sistema, con  $x_1(t)$  la conversión y  $x_2(t)$  [ $^{\circ}$  K] la temperatura dentro del reactor. A su vez,  $u(t)$  [ $^{\circ}$  K] representa la temperatura del flujo de entrada del fluido de refrigeración y actúa como la entrada del proceso, la cual está limitada al rango [300,490] [ $^{\circ}$  K]. Además, la salida del sistema,  $y(t)$ , está dada por el valor de conversión, es decir,  $y(t) = x_1(t)$ . En la Figura 1.1 se observa el diagrama de bloques de la planta a utilizar.



**Figura 1.1: Diagrama de Bloques CSTR.**

En [79] se estudia el comportamiento del reactor mediante una prueba en lazo abierto utilizando una rampa  $u(t) = t + 300$  como entrada. A partir de la respuesta de la planta en la Figura 1.2 se puede observar que el máximo de la conversión es aproximadamente 0.5088, mientras que el mínimo se aproxima a 0.02. Luego, el objetivo de control de este sistema es diseñar una estrategia que permita llevar al reactor a operar al máximo valor de conversión.



**Figura 1.2: Respuesta del CSTR Usando  $u(t) = t + 300$ .**

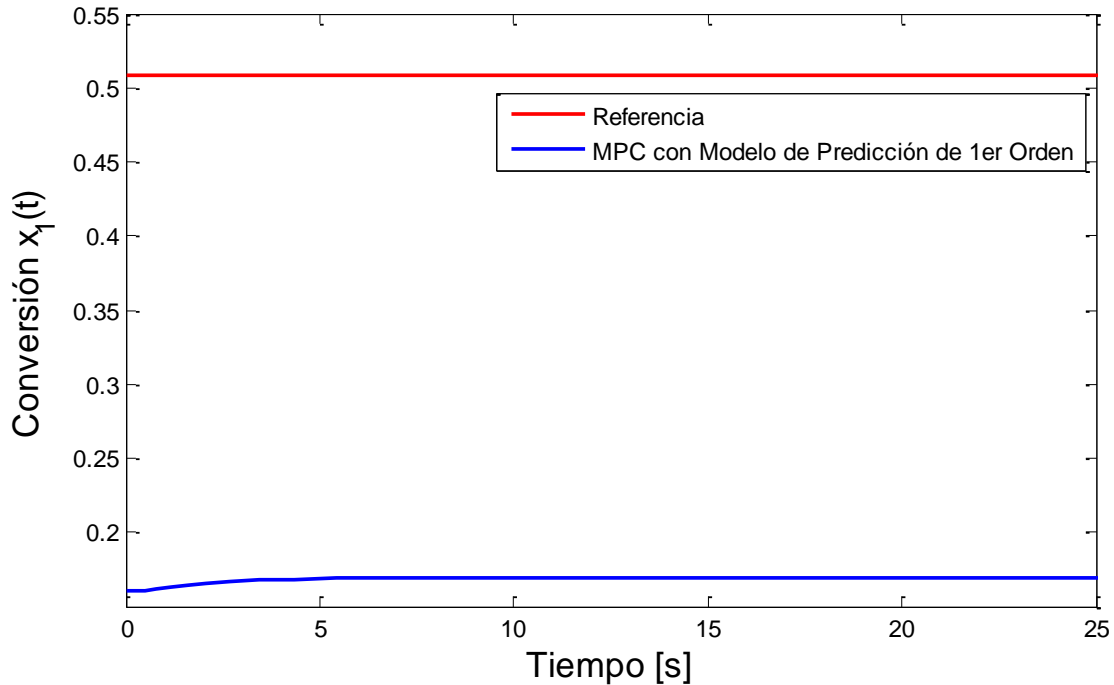
Para lograr el objetivo de control planteado se utiliza una estrategia de control predictivo basado en modelos (*Model-Based Predictive Control*, MPC), la cual tiene la función de manipular la temperatura del flujo de entrada del fluido de refrigeración para proveer las condiciones favorables dentro del reactor para incrementar la conversión. Con este propósito, se emplea una aproximación de primer orden del reactor para predecir la trayectoria de los estados de la planta.

Sea  $C = [1,0]$  y  $x(k) = [x_1(k), x_2(k)]^T$ , entonces  $\hat{y}(k) = Cx(k)$ . A su vez, acorde a (1.1)-(1.3), considérese  $f(x(t), u(t)) = dx(t)/dt$ . Luego, utilizando series de Taylor y aproximando  $dx(t)/dt \approx x(k+1) - x(k)$ , el modelo de predicción utilizado está dado por

$$\hat{y}(k+1) = C \left( x(k) + f(x_0(k), u_0(k)) + J_x(k)\Delta x(k) + J_u(k)\Delta u(k) \right), \quad (1.4)$$

donde  $\hat{y}(k+1)$  es la salida predicha,  $\Delta x(k) = x(k) - x_0(k)$ ,  $\Delta u(k) = u(k) - u_0(k)$  y  $(x_0(k), u_0(k))$  representa el punto de operación en el instante  $k$ . Además,  $J_x(k)$  y  $J_u(k)$  representan el Jacobiano del modelo del CSTR con respecto a los estados y la entrada respectivamente. La respuesta en lazo cerrado del CSTR utilizando el controlador predictivo con el modelo de predicción en (1.4) se presenta en la Figura 1.3, donde se observa que el MPC diseñado no logra controlar adecuadamente la planta.

## CSTR con Control Predictivo con Modelo de Predicción de 1er Orden



**Figura 1.3: Resultado CSTR con MPC con Modelo de Predicción de 1er Orden.**

Dicho resultado puede ser justificado matemáticamente analizando el problema de optimización formulado en el controlador predictivo. En este caso, la función objetivo a minimizar está dada por

$$J(x(k), u(k)) = e^T(k+1)Qe(k+1) + \Delta u^T(k)R\Delta u(k), \quad (1.5)$$

donde  $e(k+1) = r(k+1) - \hat{y}(k+1)$ , con  $\hat{y}(k+1)$  la trayectoria predicha por el modelo, y  $r(k+1)$  la referencia;  $Q, R > 0$  son factores de ponderación. Luego, asumiendo una implementación sin considerar las restricciones del proceso, la solución analítica para minimizar (1.5) está dada por

$$\begin{aligned} \frac{dJ(x(k), u(k))}{d\Delta u(k)} &= 0 \\ \rightarrow \Delta u^*(k) &= \left( (CJ_u(k))^T Q^T C J_u(k) + R^T \right)^{-1} (CJ_u(k))^T Q^T \left[ r(k+1) \right. \\ &\quad \left. - C \left( x(k) + f(x_0(k), u_0(k)) + J_x(k)\Delta x(k) \right) \right]. \end{aligned} \quad (1.6)$$

Cabe destacar que en (1.1)-(1.3) la trayectoria de la conversión dentro del reactor es independiente de la variable manipulada  $u(k)$ , por lo que el Jacobiano con respecto a la entrada está dado por  $J_u(k) = [0, 1]^T$ . Luego,  $CJ_u(k) = 0$  y en consecuencia  $\Delta u^*(k) = 0, \forall k$ . Por lo tanto, la acción de control no varía a lo largo de la simulación, por lo que no

es posible controlar adecuadamente la conversión dentro del CSTR utilizando este tipo de modelo.

Por otro lado, si el problema de optimización incorpora restricciones sobre la acción de control de la forma  $u_{min} \leq u(k) \leq u_{max}$ , la función de Lagrange correspondiente resulta

$$L(u(k), \lambda) = J(x(k), u(k)) + \lambda^T g(u(k)), \quad (1.7)$$

con  $g(u(k)) = [g_1(u(k)), g_2(u(k))] = [u_{min} - \Delta u(k) - u_0(k), \Delta u(k) - u_{max} - u_0(k)]^T$  las restricciones del sistema y  $\lambda^T = [\lambda_1, \lambda_2]$  los multiplicadores de Lagrange. Imponiendo las restricciones de Karush-Kuhn-Tucker (KKT) para resolver (1.7) se tiene:

$$\begin{aligned} \frac{dL(u(k), \lambda)}{d\Delta u(k)} &= \frac{dJ(x(k), u(k))}{d\Delta u(k)} - \lambda_1 + \lambda_2 = 0 \\ g(u(k)) &\leq 0 \\ \lambda^T g(u(k)) &= 0 \\ \lambda_1, \lambda_2 &\geq 0. \end{aligned} \quad (1.8)$$

Luego, las soluciones factibles para el problema de optimización son:

$$\begin{aligned} \Delta u^*(k) &= 0 \\ \Delta u^*(k) &= u_{min} - u_0(k) \\ \Delta u^*(k) &= u_{max} - u_0(k). \end{aligned} \quad (1.9)$$

Al evaluar las posibles soluciones en la función objetivo, nuevamente se obtiene como solución óptima  $\Delta u^*(k) = 0, \forall k$ .

A partir del desarrollo anterior, el modelo de predicción de primer orden no permite inferir una acción de control adecuada para controlar al CSTR, lo que empeora el desempeño en lazo cerrado del sistema al momento de seguir una referencia deseada.

Por otro lado, mediante un procedimiento de identificación de sistemas es posible obtener un modelo solamente autorregresivo de una planta para predecir su comportamiento, y en consecuencia completamente no controlable, lo cual imposibilita desarrollar una estrategia de control basada en dicho modelo, debido a que las acciones de control son variables exógenas.

Por lo tanto, utilizar un modelo inadecuado, y en particular uno no controlable, influye negativamente en las acciones de control obtenidas mediante un controlador, afectando el rendimiento del sistema. Cabe destacar que lo anterior puede ocurrir en distintas clases de sistemas, y no exclusivamente en el reactor presentado.



Teniendo lo anterior en consideración, el principal objetivo de este trabajo es diseñar métodos de identificación de sistemas difusos (aprovechando la capacidad de aproximación de este tipo de modelos) que permitan obtener modelos controlables, o en su defecto obtener modelos que aumenten el espacio de controlabilidad, de tal forma de poder diseñar estrategias que permitan alcanzar los objetivos de control planteados.

Para ello, se plantean como objetivos generales de esta investigación:

- Diseñar un algoritmo de identificación de sistemas difusos donde se incluyan restricciones acerca de la controlabilidad del modelo a obtener.
- Diseñar un algoritmo de identificación de sistemas difusos que incluya términos de orden superior con el fin de aumentar el espacio de controlabilidad.

Mientras que los objetivos específicos consisten en:

- Analizar una clase de sistemas difusos (cuya estructura se basa en utilizar funciones de pertenencia gaussianas y consecuencias lineales), formular condiciones de controlabilidad e incluirlas dentro del proceso de identificación.
- Proponer aproximaciones de segundo orden para ser incluidas dentro de la modelación difusa de Takagi-Sugeno.
- Validar las metodologías propuestas mediante el diseño de estrategias de control predictivo difuso y su posterior aplicación a casos de estudio.

A su vez, con el fin de acotar el trabajo a realizar, los alcances de esta tesis consideran realizar pruebas de identificación de sistemas en lazo abierto y lazo cerrado inicialmente, con el fin de estudiar el efecto del tipo de datos de identificación en el desempeño de una estrategia de control basada en el modelo obtenido del proceso, y de esta manera analizar lo indicado en la literatura especializada, la cual revela que es posible obtener un mejor desempeño en lazo cerrado empleando un modelo obtenido con datos de identificación en lazo cerrado.

Por otro lado, para identificar los parámetros de las consecuencias de un modelo difuso se plantea un problema de optimización cuya función objetivo es el error cuadrático a un paso (método de mínimos cuadrados), por lo que al utilizar una clase de modelos difusos con consecuencias lineales en los parámetros dicha función objetivo es convexa, siendo posible encontrar una solución única al problema de optimización. Sin embargo, las nuevas metodologías propuestas en este trabajo incluyen modificaciones de la función objetivo del problema de optimización para identificar los parámetros de las consecuencias, por lo que las propiedades de convexidad de dicha función no se pueden asegurar a priori. Un análisis matemático detallado de la convexidad de la función objetivo escapa del alcance de este trabajo.

Para validar las metodologías de identificación difusa propuestas se diseñan estrategias de control predictivo difuso, los cuales son sintonizados de tal manera de obtener bajos errores en estado estacionario. Un estudio matemático formal de la

estabilidad de los sistemas en lazo cerrado al emplear este tipo de estrategias de control esta fuera del alcance de la investigación realizada.

La implementación de las propuestas planteadas en esta tesis se realiza en ambiente MATLAB, utilizando como herramientas base un *toolbox* de identificación difusa de Takagi-Sugeno desarrollado por Núñez et al. [77], un *toolbox* de *clustering* difuso desarrollado por Balasko et al. [12], y un *toolbox* del algoritmo de optimización por enjambre de partículas (*Particle Swarm Optimization*, PSO) desarrollado por Jagatpreet Singh [40]. A su vez, las estrategias de control son implementadas en ambiente MATLAB-Simulink, y dado a que sólo se trabaja a nivel de simulaciones, se desprecia como limitante el tiempo necesario para que el controlador predictivo encuentre una acción de control óptima, sin embargo, en la Sección 8.2.3 de anexos se realiza un breve análisis respecto al esfuerzo computacional requerido por el controlador y se plantean alternativas para afrontar las complicaciones que nacen al momento en que encontrar una acción de control óptima se demora más que el tiempo de muestreo del sistema.

Cabe destacar que, dado a que el trabajo se realiza en base a simulaciones solamente, se asume conocida la planta en un principio con el fin de diseñar una estrategia de control para la obtención de datos de identificación en lazo cerrado. Posterior a la obtención de los datos de entrada-salida para la modelación del sistema, se deja de asumir que la planta es conocida y se utiliza el modelo obtenido para el diseño de estrategias de control.

Por último, todas las implementaciones y conclusiones derivadas de este trabajo se enmarcan en los sistemas de una entrada y una salida (*Single Input - Single Output*, SISO), por lo que no se debe asumir que las formulaciones planteadas y los resultados obtenidos pueden ser extrapolados directamente a sistemas de múltiples entradas y múltiples salidas (*Multiple Input - Multiple Output*, MIMO).

El resto de esta tesis se divide principalmente en seis capítulos, considerando el actual (Capítulo 1) donde se plantea la introducción del problema a abordar, la motivación para resolver dicho problema desde el punto de vista de la teoría de control, y los objetivos generales y específicos a desarrollar durante el transcurso de este trabajo.

El Capítulo 2 detalla el estudio de la identificación difusa en la literatura y presenta el procedimiento paso a paso para abordar un problema de este tipo, partiendo por el diseño de un experimento de identificación adecuado, hasta la determinación de los parámetros del modelo difuso, pasando por métodos para optimizar su estructura.

El Capítulo 3 describe el análisis de la controlabilidad de sistemas dinámicos realizado en la literatura, donde se plantean las diferentes maneras de corroborar esta propiedad tanto para sistemas lineales como para sistemas difusos.

En el Capítulo 4 se detalla la primera de las propuestas: incluir restricciones de controlabilidad en el procedimiento de identificación difusa con el fin de obtener

modelos controlables. Dentro de este capítulo se presenta la derivación de las restricciones de controlabilidad a incluir en el algoritmo y se detalla la formulación del problema de optimización para la identificación de los parámetros del modelo. Finalmente se plantean dos casos de estudio donde se valida la metodología propuesta.

En el Capítulo 5 se incluye la segunda de las propuestas de identificación difusa: incluir términos de orden superior en los consecuentes del modelo difuso. En particular, en este trabajo se emplean aproximaciones de segundo orden. En este capítulo se detalla la derivación de este tipo de aproximaciones, su aplicación al diseño de estrategias de control predictivo, el diseño de modelos difusos de Takagi-Sugeno con términos de segundo orden en sus consecuencias y la posterior descripción de una estrategia predictiva empleando estos modelos. Luego se emplean los controladores predictivos utilizando aproximaciones de segundo orden y modelos TS con términos de segundo orden como predictores en un caso de estudio, comparando sus desempeños.

Finalmente, se presentan las conclusiones de esta tesis, como también el trabajo futuro y las publicaciones asociadas a los largo de este desarrollo.

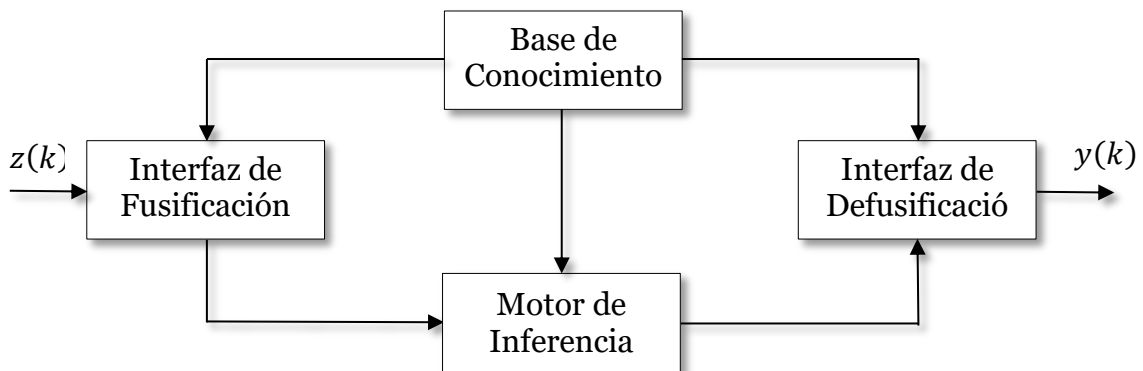
## 2. Identificación de Sistemas Difusos

Los modelos difusos están fundamentados en el concepto de lógica difusa propuesto por Zadeh en 1973 [119], donde se introduce la noción de variable lingüística, conjuntos y algoritmos difusos para representar comportamientos condicionales del estilo *Si-Entonces*, definidos a partir del conocimiento intrínseco de una planta. De esta forma se propone describir la dinámica de los procesos a partir de nociones conceptuales que abarquen un rango o grupo, denotando cierta incertidumbre o ambigüedad, y no un número determinístico como lo hace la lógica convencional.

Basado en lo anterior, en la literatura se han propuesto varios enfoques de modelación de sistemas difusos, donde destacan particularmente dos: sistemas difusos de Mamdani [63] y sistemas difusos de Takagi-Sugeno [107]. El primero de ellos está enfocado principalmente a aplicaciones de control de procesos, con el fin de emular el comportamiento de un operador frente a cambios en el estado de una planta, de tal forma de diseñar un controlador no lineal basado en un sistema de reglas empíricas del estilo *Si-Entonces*. Por otro lado, los sistemas difusos TS destacan por su versatilidad tanto en aplicaciones de control como de modelación de sistemas. Takagi & Sugeno proponen una metodología que se caracteriza por estar basada en un conjunto de reglas definidas sobre subespacios difusos, donde las consecuencias (salidas) de estas reglas corresponden generalmente a relaciones lineales entrada-salida. Estos modelos se presentan en mayor detalle en la Sección 2.1.

Un esquema difuso se caracteriza por poseer cuatro elementos principales: interfaz de fusificación, base de conocimientos, motor de inferencia e interfaz de defusificación, los cuales se relacionan unos con otros de la manera expuesta en la Figura 2.1 [53]. La interfaz de fusificación se encarga de transformar las variables de entrada o premisas,  $z(k)$ , en variables difusas, teniendo en consideración el grado en que pertenece cada variable a los conjuntos difusos del sistema. La base de conocimientos estipula la forma de relacionar las premisas con la salida mediante un conjunto de reglas. El motor de inferencia se encarga de obtener la relación entre las variables de salida  $y(k)$  y las variables de entrada, utilizando el conjunto de reglas definido por la base de conocimientos. Finalmente, la interfaz de defusificación se encarga de obtener un valor determinístico de  $y(k)$  a partir de lo entregado por el motor de inferencia.

Dado el esquema de un sistema difuso presentado, la principal diferencia entre el enfoque de Mamdani y el de Takagi-Sugeno se presenta en la etapa de interfaz de defusificación, debido a que Mamdani representa las consecuencias del modelo como un conjunto difuso, mientras que TS las representa mediante funciones.



**Figura 2.1: Esquema de un Sistema de Lógica Difusa.**

En particular, el enfoque de Takagi & Sugeno puede retornar un buen modelo de predicción cuando se utilizan consecuencias lineales, siendo suficiente para representar las dinámicas de un proceso, sin embargo, en la literatura también se pueden encontrar ejemplos donde los consecuentes son funciones no lineales. En [86] se proponen modelos difusos TS cuyas consecuencias son funciones sinusoidales de las premisas, y a partir de dicho modelo se pueden desarrollar estrategias de control. En este caso utilizan la técnica *Parallel Distributed Compensation* (PDC), la cual consiste en diseñar un controlador para cada consecuencia y luego combinar la información de cada controlador local. La motivación de utilizar este tipo de no linealidades en las consecuencias radica en que varios procesos pueden tener estados representados por un movimiento oscilatorio de tipo sinusoidal, como el péndulo rotacional, sistemas de acoplamiento de remolques, resortes, entre otros. Este estudio demuestra que el uso de modelos difusos con consecuencias con términos no lineales en estos casos trae consigo mejores resultados en comparación al utilizar modelos difusos con consecuencias lineales, mejorando el desempeño en lazo cerrado del sistema.

Otro estudio de este tipo es el reportado en [118], donde las aproximaciones locales en cada regla del modelo difuso son representadas por ecuaciones dinámicas no lineales afines, es decir, la  $r$ -ésima regla posee una consecuencia de la forma  $\dot{x}_r(t) = f_r(x(t)) + g_r(x(t))u(t)$ . En este trabajo se diseñan controladores que aseguren la estabilidad en lazo cerrado del sistema, y para ello hace uso del enfoque de Lyapunov, de donde se obtienen condiciones para asegurar la estabilidad asintótica. Luego, resolviendo el sistema de inecuaciones que se obtiene al imponer dichas condiciones, se deriva un controlador no lineal. Se establece también que los modelos difusos TS con no linealidades en sus consecuencias pueden necesitar una menor cantidad de reglas que el modelo difuso TS tradicional (con consecuencias lineales). Sin embargo, si la base de reglas es demasiado extensa, el cálculo de la acción de control puede resultar bastante complejo, debido a que el sistema de inecuaciones obtenido al imponer las condiciones de Lyapunov se vuelve más complicado de resolver.

Dadas las ventajas reportadas en la literatura del uso de modelos TS para representar el comportamiento de sistemas, diversas estrategias han sido abordadas para identificar un modelo difuso de un proceso. En 1994, Zhao et al. [121] detallan un procedimiento que integra tanto la identificación de la estructura y los parámetros del modelo difuso, como también la determinación de variables de entrada y salida. Esta metodología se basa en realizar una agrupación difusa (*clustering* difuso), utilizando el algoritmo de Gustafson-Kessel (GK), método basado en minimizar una función de costos para encontrar las agrupaciones o *clusters* difusos. Un *cluster* se define como un conjunto de datos que poseen cierta similitud entre sí, pero a su vez poseen distinta información que los elementos de otro *cluster*. De esta manera cada *cluster* representa una clase, y en términos de los modelos difusos TS, éstos representan una regla del sistema. A diferencia del *clustering* convencional, los elementos que componen un *cluster* difuso tienen asociado un grado de pertenencia que va desde 0 a 1, siendo así posible pertenecer a múltiples clases con distintos grados de pertenencia. Luego de obtener dichos *clusters* se emplea el algoritmo *Compatible Cluster Merging* (CCM), el cual compara los valores y vectores propios de las matrices de covarianza de los *clusters* encontrados con GK, y mezcla los que posean características similares, es decir, combina los conjuntos que contienen información relativamente equivalente. Una vez definidas las relaciones difusas, éstas son proyectadas en el espacio de variables de entrada, donde el modelo puede ser mejorado mediante el ajuste de funciones de pertenencia y la eliminación de premisas irrelevantes. En cuanto a las consecuencias de cada regla, se consideran relaciones lineales de las premisas, cuyos parámetros son estimados mediante el método de mínimos cuadrados.

El trabajo anterior propone una metodología básica, la cual es mejorada en 1996 por de Bruin & Roffel [22]. Los autores proponen modificar el algoritmo CCM añadiendo una etapa donde se calculan los grados de compatibilidad entre los conjuntos difusos, para luego determinar qué agrupaciones deben ser combinadas mediante la elección de un umbral que actúe sobre dichos valores. Posteriormente, debido a que se puede producir pérdida de información, se incluye una rotación del espacio de entrada al momento de proyectar las relaciones difusas, definiendo nuevas variables de proyección que permitan reducir dicha pérdida y definir las funciones de pertenencia de manera más clara. A su vez, la etapa de identificación de los parámetros de las consecuencias es abordada por un método más directo que el propuesto en [121], ya que aprovecha el ajuste realizado por el algoritmo GK, rescatando los centros de las agrupaciones y los vectores propios de la matriz de covarianza, prescindiendo de la necesidad de reajustar a través del método de mínimos cuadrados. Esta metodología tiene la ventaja de ser más precisa y contiene menos pérdida de información que el enfoque en [121], además de que la identificación directa de los parámetros de las consecuencias puede resultar más eficiente computacionalmente para grandes volúmenes de datos.

Babuska [10] hace referencia a ambos métodos expuestos anteriormente, detallando la metodología de identificación de modelos difusos TS con consecuencias lineales y los diversos algoritmos existentes para la realización de *clustering* difuso, donde destaca el uso para identificación del algoritmo Gustafson-Kessel, dado que puede detectar agrupaciones con distintas formas, y es más robusto al escalamiento de los datos e

inicialización del algoritmo, debido a que está basado en una medida de distancia adaptiva. A su vez, GK ha sido utilizado en estrategias de identificación de sistemas no lineales en conjunto con otros tipos de herramientas, como redes neuronales, demostrando que el modelo obtenido logra un buen desempeño como predictor [111].

Ayday & Eksin [9], y Eksin & Erol [25] proponen otro método para la identificación de modelos difusos basada en la división del espacio de estados. El diagrama de fase del sistema obtenido a partir de datos es dividido en subregiones difusas y a cada subregión se le asigna un modelo lineal (ya sea en variables de estado o en ecuaciones de entrada-salida). Luego, mediante el método de mínimos cuadrados se identifican los parámetros de dichos modelos. La ventaja de este enfoque es que permite aprovechar el conocimiento a priori que se tenga de la planta a modelar, identificando la dinámica lineal en diversos puntos de operación. Por otro lado, este procedimiento se vuelve complejo cuando no se tiene dicho conocimiento, ya que para dinámicas altamente no lineales el diagrama de fase puede resultar caótico, y en consecuencia la división del espacio de estado puede no representar fielmente el comportamiento de la planta.

En [90] se propone una metodología para identificación difusa general donde se considera la elección de variables relevantes, optimización de la estructura e identificación de los parámetros del modelo. Este enfoque de identificación se fundamenta en un proceso iterativo en el cual la elección del número óptimo de *clusters* y la estructura óptima en general del modelo se basa en minimizar un índice de desempeño, el cual en este trabajo corresponde a la raíz del error cuadrático medio (*Root Mean Squared Error*, RMSE).

La mayoría de las estrategias para obtener un modelo difuso de un sistema dividen el problema en dos categorías: identificar los parámetros de las premisas, e identificar los parámetros de las consecuencias. Esto se debe a que de esta manera la resolución del problema es más eficiente, permitiendo agregar funciones más complejas en los consecuentes, mejorando a su vez la precisión del modelo obtenido [56]. Por otro lado, si bien la mayoría de los algoritmos utilizados para identificar los parámetros de las premisas del modelo difuso se basan en GK o Fuzzy c-Means (FCM), hay trabajos donde se proponen otro tipo de *clustering* difuso. Por ejemplo, en [2] se propone el uso del algoritmo Gath-Geva, un método similar a GK, pero que modifica la definición de distancia, el cual permite obtener una buena aproximación del comportamiento de procesos. Otro ejemplo es el presentado en [51], donde se utiliza una red neuronal para realizar la tarea del *clustering* de los datos, buscando con ello reducir el error de predicción entre el modelo y la planta real. Por otro lado, en [55] se reporta una nueva estrategia de *clustering* denominada *Gravitational Search-Based Hyperplane*, la cual modifica la métrica de distancia en la función objetivo del algoritmo de *clustering*, y resuelve el problema de optimización utilizando el algoritmo de búsqueda gravitacional (*Gravitational Search Algorithm*, GSA), algoritmo inspirado por PSO y algoritmos genéticos (*Genetic Algorithms*, GA), el cual busca emular el comportamiento de la fuerza gravitacional y la atracción que induce en individuos con distinta masa.

Las referencias reportadas anteriormente abordan el problema de identificación desde su base, otorgando distintas alternativas para modelar un proceso, y no consideran casos particulares para mejorar los resultados frente a escenarios adversos. En particular, Soltani et al. [101] abordan el problema de identificación cuando hay ruido en la medición. Para ello utilizan el llamado *Fuzzy Noise Clustering* (FNC), el cual asume al ruido como una clase separada, apartada a una distancia  $\delta$  de todos los datos, en otras palabras, el algoritmo crea un *cluster* ficticio correspondiente a los datos de ruido. Para ello, se modifica la función objetivo del algoritmo de *clustering*, adicionando un término asociado a  $\delta$ , lo cual conlleva a mejores resultados en la identificación del sistema cuando se tienen niveles de ruido considerables. A su vez, siguiendo la misma línea de investigación, Soltani et al. [102] proponen el uso de una modificación del algoritmo Fuzzy c-Regression Model (FCRM), la cual al complementarla con el planteamiento presentado en [101] genera un algoritmo más robusto a datos con ruido. Por otro lado, otra línea de investigación en cuanto a identificación de sistemas difusos que presenta robustez a perturbaciones es la exhibida en [97], donde se plantea el uso de intervalos difusos que capturen las incertidumbres de los datos y la planta mediante funciones que abarquen un límite superior e inferior.

Todas las estrategias de identificación reportadas anteriormente son realizadas fuera de línea, es decir, se entrena y valida un modelo antes de ser empleado con fines de predicción, análisis, diseño de controladores, etc. Sin embargo, existen otros métodos que plantean realizar la identificación de manera *online*, de tal forma que el modelo se adapte a los nuevos datos que se obtengan del proceso.

En 2004, Angelov & Filev [6] proponen un nuevo enfoque de identificación de modelos difusos: identificación en tiempo real, *Evolving TS Model* (ETS). Este algoritmo actualiza la estructura y parámetros del modelo mediante aprendizaje supervisado y no supervisado, donde el primero se aplica para la estimación de los parámetros y el segundo para la construcción de la base de reglas. Como Fuzzy c-Means y Gustafson-Kessel son algoritmos de *clustering* difuso utilizados para identificación *offline*, se propone un nuevo método de *clustering*: *Online Clustering*. Esta técnica consiste en definir una función que calcula el potencial de cada dato recursivamente, y el que posea mayor valor pasa a ser el centro del conjunto. Posteriormente se define un potencial de referencia para determinar futuros centros, y en consecuencia nuevas reglas, o bien reemplazar a un centro anterior si el nuevo dato posee alto potencial y está cerca. Para la estimación de los parámetros de las consecuencias de cada regla se puede utilizar el método de mínimos cuadrados recursivo (*Recursive Least Square*, RLS) o el RLS ponderado (*weight RLS*, wRLS), los cuales hacen uso de la matriz de covarianza del error de predicción para actualizar los valores en cada instante de tiempo. Sin embargo, debido a que la base de reglas va evolucionando, se propone definir los parámetros de la nueva regla como una suma ponderada de los parámetros de reglas establecidas con anterioridad. A su vez, se incluye un factor de corrección a la matriz de covarianza del error que emula el hecho de que la nueva regla hubiese estado desde el inicio. Las ventajas de este enfoque es que puede adaptarse a cambios en el comportamiento del proceso a identificar. Además puede comenzar el entrenamiento con sólo un dato e ir mejorando el desempeño con el tiempo. Computacionalmente es más efectivo que la



identificación *offline* dada la recursividad para realizar los cálculos correspondientes. Posteriormente, este mismo enfoque es aplicado para sistemas MIMO [7].

Por otro lado, en [115] se propone un método de identificación *online* haciendo uso de redes neuro-difusas, permitiendo la identificación de modelos difusos tanto TS como Mamdani. Para ello se utiliza una red neuronal de 4 capas que permite emular los cálculos de grados de pertenencia a los conjuntos difusos, los cálculos de grados de activación y la ponderación de las salidas locales del set de reglas. El uso del algoritmo de gradiente descendente con una tasa de aprendizaje variable en el tiempo permite entrenar la red asegurando que el error entre el modelo y la salida real sea acotado. A diferencia del enfoque de Angelov & Filev en [6], en este caso se considera una estructura fija, lo cual puede resultar en una pérdida de desempeño cuando la planta cambia su comportamiento drásticamente.

En 2011, Salahshoor & Ahangari [91] se basan en el enfoque de Angelov & Filev para identificación *online*, es decir, en el algoritmo ETS. En este trabajo se aplica dicha técnica en la identificación de sistemas no lineales con dinámica híbrida, es decir, existen estados continuos y discretos, y además el proceso posee comportamientos alternantes. La contribución de este desarrollo radica en la modificación del cálculo de potencial de los datos, cuya principal repercusión es ralentizar el crecimiento de la base de reglas. Lo que proponen los autores es utilizar un coeficiente que pondere el umbral que determina si un nuevo punto corresponde al centro de un *cluster* o no, donde dicho ponderador va aumentando desde 1 a valores mayores, lo que en consecuencia permite que el umbral aumente, reduciendo la cantidad de nuevas reglas en el modelo. Luego es posible obtener modelos más sencillos y con un buen nivel de precisión.

Otro tipo de estrategias de identificación, a diferencia del caso básico y el enfoque *online*, se basan en incluir restricciones en el procedimiento que permitan manipular el conocimiento que se obtiene de la planta y de esta forma obtener un mejor modelo de predicción. Por ejemplo, en [120] se plantea un método de identificación de sistemas lineales en donde se restringe el rango del modelo lineal mediante intervalos difusos, planteando un problema de mínimos cuadrados con restricciones, demostrando que el modelo obtenido se acopla mejor a los datos que un modelo lineal obtenido sin utilizar este tipo de restricciones. Por otro lado, en [1] se plantea un algoritmo de identificación difusa donde se incluyen restricciones para identificar los parámetros de las consecuencias del modelo difuso, acotando el rango de dichos parámetros, o bien acotando el rango de variación de la salida del sistema; donde ambos tipos de restricciones se deben conocer a priori.

Si bien la identificación de sistemas difusos ha sido bastante abordada en la literatura, los resultados reportados en las referencias citadas sólo se enfocan en minimizar el error de modelación y/o predicción, y no consideran un estudio más profundo de los modelos obtenidos, tales como la estabilidad, observabilidad o controlabilidad; ni mucho menos cómo incorporar restricciones para obtener un modelo difuso que cumpla con estas propiedades, por lo que realizar dicho análisis aparece como un campo abierto para la investigación.

Dada las propuestas de identificación difusa reportadas en la literatura especializada, en las secciones a continuación se detalla el procedimiento paso a paso a utilizar en este trabajo para identificar un sistema, describiendo tanto los modelos difusos a emplear como también cada etapa de la metodología.

## 2.1. Modelos Difusos de Takagi-Sugeno

Los modelos difusos de Takagi-Sugeno son estructuras basadas en la lógica difusa que permiten representar procesos con dinámicas no lineales mediante la combinación de información otorgada por modelos locales. Estos tipos de modelos pueden ser expresados a partir de una base de reglas del tipo “Si-Entonces”, de la forma

$$R_r: \text{ Si } z_1(k) \text{ es } MF_1^r \text{ y } \dots \text{ y } z_p(k) \text{ es } MF_p^r \text{ entonces } y_r(z(k)) = f_r(z(k)), \quad (2.1)$$

donde  $R_r$  denota la  $r$ -ésima regla del modelo difuso, con  $r \in \{1, \dots, N_r\}$  y  $N_r$  el número total de reglas;  $y_r(z(k))$  es su consecuencia o modelo local;  $z(k) = [z_1(k), \dots, z_p(k)]$  es el vector de premisas en el tiempo  $k$ , las cuales por lo general son regresores de la entrada y/o salida del sistema;  $f_r(z(k))$  es una función de las premisas del modelo; y  $MF_i^r$  es el conjunto difuso (función de pertenencia) de la  $i$ -ésima premisa correspondientes a la  $r$ -ésima regla.

Sea  $\mu_r(z_i(k))$  el grado de pertenencia de la  $i$ -ésima premisa  $z_i(k)$  al conjunto difuso  $MF_i^r$ , donde  $\mu_r(z_i(k)) \in [0,1]$ , siendo 0 cuando la premisa no pertenece en ningún grado al conjunto  $MF_i^r$ , y siendo 1 si pertenece completamente a dicho conjunto. Luego, se define el grado de activación de la  $r$ -ésima regla,  $w_r(z(k))$ , como

$$w_r(z(k)) = \text{oper} \left( \mu_r(z_1(k)), \dots, \mu_r(z_p(k)) \right), \quad (2.2)$$

donde  $\text{oper}(\cdot)$  puede ser el operador mínimo o el producto. Se denota  $h_r(z(k))$  al grado de activación normalizado de la  $r$ -ésima regla, es decir,

$$h_r(z(k)) = \frac{w_r(z(k))}{\sum_{l=1}^{N_r} w_l(z(k))}. \quad (2.3)$$

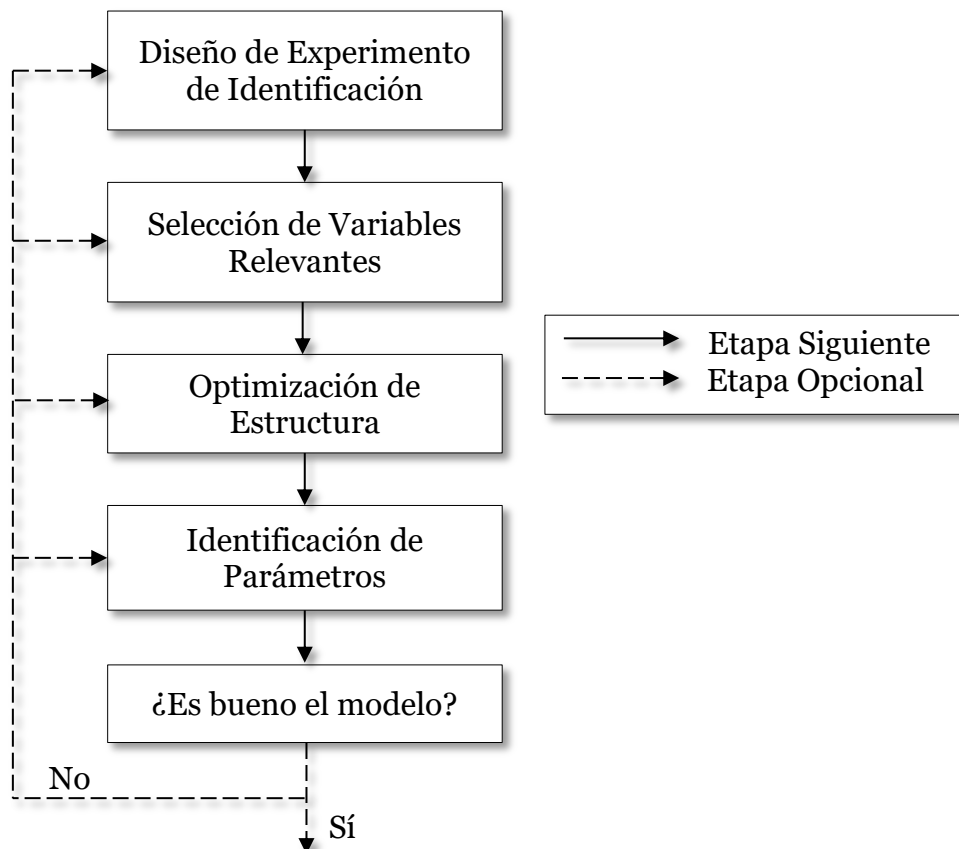
De acuerdo a (2.3), el grado de activación normalizado satisface que  $h_r(z(k)) > 0$ ,  $r = 1, \dots, N_r$  y  $\sum_{r=1}^{N_r} h_r(z(k)) = 1$ .

Ya definido el grado de activación de cada regla, la salida del modelo difuso,  $y_{fuzzy}(k)$ , está dada por una suma ponderada de cada modelo local por su grado de activación normalizado, de la forma

$$y_{fuzzy}(k) = \sum_{r=1}^{N_r} h_r(z(k)) \cdot y_r(z(k)). \quad (2.4)$$

De acuerdo a (2.1)-(2.4), los modelos difusos TS son una clase de sistemas no lineales, cuya formulación requiere definir una serie de variables que no se conocen a priori, por lo que una estructura adecuada para representar un sistema es desconocida y, como consecuencia, un proceso de identificación debe ser llevado a cabo para determinar la estructura y cada uno de los parámetros del modelo.

El proceso de identificación de un sistema difuso se basa en datos obtenidos de la planta real, y es descompuesto en varios pasos. Un diagrama de las etapas a tener en cuenta se puede observar en la Figura 2.2 [90]. Cabe destacar que este procedimiento consta de varias fases, y en consecuencia cada una de ellas puede ser abordada en mayor profundidad para mejorar los resultados obtenidos en caso de ser necesario. En particular, la identificación de sistemas es un proceso iterativo, por lo que en caso de obtener un modelo que no cumpla con el desempeño esperado se puede repetir alguna de sus etapas tal de alcanzar un estándar deseado. En las secciones siguientes se describe cada una de las etapas del procedimiento empleado.



**Figura 2.2: Procedimiento de Identificación de Sistemas Difusos.**

## 2.2. Diseño de Experimento de Identificación

El primer paso del procedimiento de identificación es diseñar un experimento informativo, es decir, un experimento que permita generar datos que representen la dinámica del sistema en la mayor cantidad de rangos de operación posibles, esto con el fin de entrenar un modelo difuso que represente fielmente (casi) todas las condiciones de operación de la planta.

Un experimento de identificación se basa en la obtención de datos de entrada-salida del sistema a modelar, y por lo tanto es necesario definir las dinámicas de interés y/o el objetivo del modelo que se desea plantear. A partir de este análisis inicial, se debe diseñar una entrada  $u(k)$  que excite a la planta en el rango de frecuencias en que se encuentran los fenómenos de interés. Cabe destacar que en la práctica existen limitaciones debido a los actuadores y/o modos de operación del sistema. Luego, el diseño de la señal de excitación  $u(k)$  debe considerar todos estos elementos. A su vez, debido a que los fenómenos de interés son finitos, es decir, los rangos de operación que se desean modelar son limitados, el modelo obtenido tiene un ancho de banda acotado, y por lo tanto las dinámicas definidas por fuera de dicha banda podrían no ser representadas adecuadamente [58][74].

El objetivo del experimento es minimizar la matriz de covarianza del error de estimación de parámetros, la cual es inversamente proporcional a la potencia de la señal de excitación [58]. Dicha propiedad de la señal es abordada en función del *Crest Factor*,  $C_r$ , el cual para señales de media cero se define como

$$C_r^2 = \frac{\max_k u^2(k)}{\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N u^2(k)}. \quad (2.5)$$

Luego, el objetivo es diseñar  $u(k)$  tal que minimice  $C_r$ . De (2.5) se tiene que el límite óptimo se alcanza cuando  $C_r = 1$ , lo que en la práctica se logra cuando se utilizan señales binarias simétricas  $u(k) = \pm \bar{u}$ . Sin embargo, debido a las limitaciones físicas que puede tener la planta, puede que no sea factible emplear una señal binaria, no obstante, una señal de entrada que varíe entre  $u_{min}$  y  $u_{max}$  permite mejorar el *Crest Factor* en comparación a otro tipo de señales [74]. Además, se debe tener en cuenta que la señal diseñada debe ser informativa, y para ello es necesario que  $u(k)$  sea de excitación persistente, es decir, su espectro debe ser positivo para casi todas las frecuencias; o en su defecto debe ser distinto de cero para al menos  $n$  frecuencias, en cuyo caso se denomina como señal de excitación persistente de orden  $n$ . Cabe destacar que se necesita una señal de excitación persistente de al menos orden  $n$  para identificar  $n$  parámetros [58].

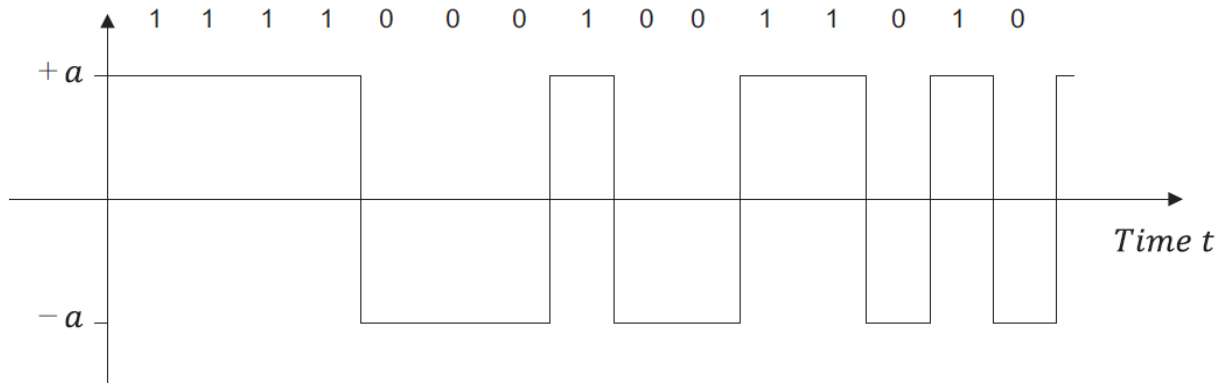
Dado lo anterior, existen distintos tipos de señales de excitación: suma de funciones sinusoidales, señales binarias, ruido blanco gaussiano filtrado, entre otras [58]. En particular, en este trabajo se propone el uso de señales binarias pseudo aleatorias (*Pseudo Random Binary Signal*, PRBS), debido a que las señales binarias mejoran  $C_r$  y

el espectro de una PRBS se aproxima al espectro del ruido blanco de banda limitada [26].

Una PRBS de orden  $n$  es una señal periódica y determinística generada a partir de la expresión

$$u(k) = \text{rem}(a_1u(k-1) + \dots + a_nu(k-n), 2). \quad (2.6)$$

donde  $\text{rem}(x, 2)$  es el residuo cuando  $x$  es dividido en 2, es decir, la operación se ejecuta calculando el módulo 2 de la expresión en (2.6). Luego, como el residuo de un número al ser dividido por 2 puede ser cero o uno solamente, se tiene que  $u(k) \in \{0,1\}$ . Posteriormente se puede modificar el rango de la señal de excitación cambiando los valores límites anteriores a otros dos cualquiera. El vector  $[u(k-1), \dots, u(k-n)]$  máximo puede tomar  $2^n$  valores distintos, luego la PRBS tiene un período máximo de  $2^n$ . Sin embargo, si se tiene el estado cero, la señal permanece en cero, por lo que se elimina dicho estado, obteniendo como período máximo  $N = 2^n - 1$  [58]. En la Figura 2.3 se observa la forma de una PRBS desplazada a los valores  $\pm a$ .



**Figura 2.3: PRBS [26].**

Cabe destacar que sólo las PRBS de período máximo poseen propiedades de ruido blanco [26]. Se ha mostrado que para cada orden  $n$  de la PRBS, existen valores  $a_i$  en la expresión (2.6) tales que la señal posee largo máximo (período igual a  $N$ ). Dichos valores se resumen en la Tabla 2.1, donde la tercera columna indica los valores de los coeficientes  $a_i$  tal que la PRBS sea de período máximo [58].

**Tabla 2.1: Coeficientes para Generar PRBS de Largo Máximo.**

Orden $n$	$N = 2^n - 1$	Coeficientes $a_i$
2	3	$a_1 = a_2 = 1$
3	7	$a_1 = 0$ $a_2 = a_3 = 1$
4	15	$a_1 = a_4 = 1$ $a_2 = a_3 = 0$
5	31	$a_2 = a_5 = 1$ $a_i = 0, \forall i \neq \{2,5\}$
6	63	$a_1 = a_6 = 1$ $a_i = 0, \forall i \neq \{1,6\}$
7	127	$a_3 = a_7 = 1$ $a_i = 0, \forall i \neq \{3,7\}$
8	255	$a_1 = a_2 = a_7 = a_8 = 1$ $a_i = 0, \forall i \neq \{1,2,7,8\}$
9	511	$a_4 = a_9 = 1$ $a_i = 0, \forall i \neq \{4,9\}$
10	1023	$a_7 = a_{10} = 1$ $a_i = 0, \forall i \neq \{7,10\}$
11	2047	$a_9 = a_{11} = 1$ $a_i = 0, \forall i \neq \{9,11\}$

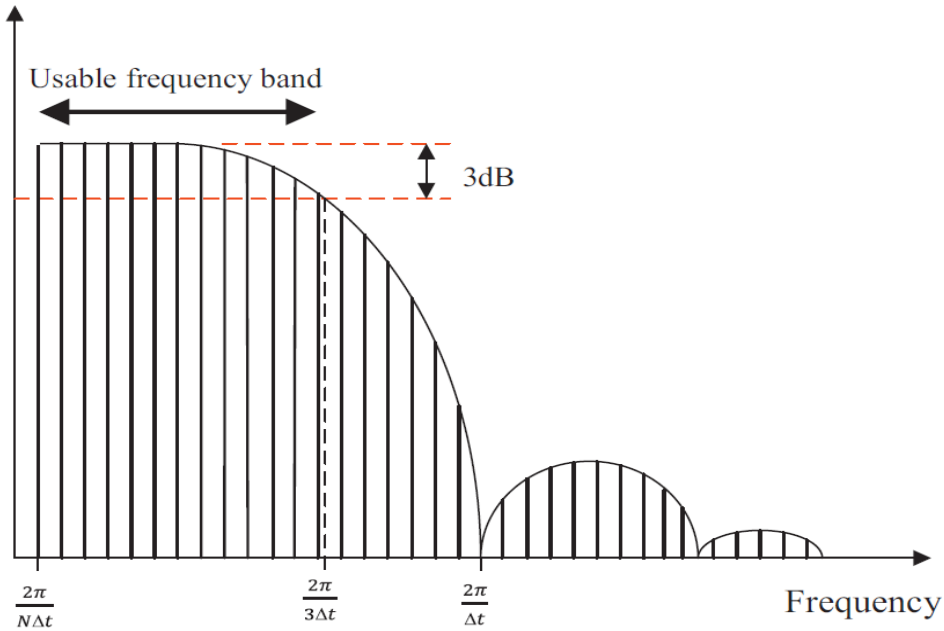
Luego de diseñar la PRBS, cada bit puede ser muestreado acorde al tiempo de muestreo utilizado en el proceso. Por otro lado, la máxima frecuencia de interés para un proceso de identificación utilizando PRBS ocurre cuando la ganancia del espectro de la señal disminuye en 3 [dB] [26][74], es decir,  $f_{max} = f_{3dB}$ . En la Figura 2.4 se presenta el espectro de frecuencias de una señal PRBS. En dicha figura se observa que el ancho de banda efectivo de la señal está dado por el rango  $f = [\frac{1}{N\Delta t}, \frac{1}{3\Delta t}]$ , donde  $\Delta t$  es el período del bit de la PRBS.

En [14] se demuestra analíticamente que la relación entre la frecuencia al disminuir la ganancia en 3 [dB] y la frecuencia de cambio de bit de la PRBS,  $f_c = 1/\Delta t$ , esta dada por

$$f_{3dB} = 0.44f_c \rightarrow f_c \approx 2.27f_{3dB} = 2.27f_{max}. \quad (2.7)$$

Sin embargo, algunos textos aproximan dicho valor y utilizan la relación [26][58]

$$f_c = 2.5f_{max}. \quad (2.8)$$



**Figura 2.4: Espectro en Frecuencia de PRBS [26].**

Luego, y como se explica al inicio de esta sección, para diseñar una señal de excitación PRBS se requiere conocer el rango de frecuencias en que se encuentran las dinámicas de interés, es decir,  $f_{min}$  y  $f_{max}$ . A su vez, se necesita conocer el tiempo de muestreo,  $T_s$ , para el experimento, el cual se puede deducir a partir del límite superior de la banda de frecuencia a modelar, o bien utilizar un valor entre la décima y la vigésima parte del tiempo de estabilización de la planta [74], teniendo siempre en consideración el teorema de Shanon (o teorema de muestreo) [84]. De acuerdo a este teorema, el tiempo de muestreo debe seleccionarse de tal forma que la desigualdad (2.9) se satisfaga.

$$f_s = \frac{1}{T_s} > 2f_{max}. \quad (2.9)$$

Una vez seleccionado el tiempo de muestreo e identificado el ancho de banda de interés, se calcula la frecuencia de cambio de bits de la PRBS,  $f_c$ , a partir de (2.8). Por otro lado, el período de cambio de bits de la PRBS,  $\Delta t$ , puede ser escrito en función de la cantidad de muestras por bit,  $N_s$ , y el tiempo de muestreo,  $T_s$ , de la forma  $\Delta t = N_s \cdot T_s$ . Así se tiene que

$$N_s = \frac{\Delta t}{T_s} = \frac{f_s}{f_c}. \quad (2.10)$$

Vale la pena resaltar que  $N_s$  debe ser un número entero, por tanto sólo la parte entera del resultado en (2.10) debe ser tomada en cuenta.

A partir del espectro de la PRBS, se tiene que

$$f_{min} = \frac{1}{N\Delta t} = \frac{f_c}{N} = \frac{f_c}{2^n - 1}. \quad (2.11)$$

Luego, de (2.11) se obtiene la expresión para el orden de la PRBS,  $n$ , dada por

$$n = \frac{\log\left(\frac{f_c}{f_{min}} + 1\right)}{\log 2}. \quad (2.12)$$

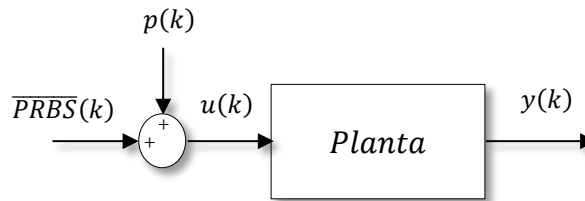
Al igual que  $N_s$ , el orden de la PRBS también debe ser truncado a un valor entero.

Ya definidos  $n$  y  $N_s$  es posible generar la PRBS de período máximo utilizando la expresión (2.6) y los coeficientes presentes en la Tabla 2.1 correspondientes al orden calculado.

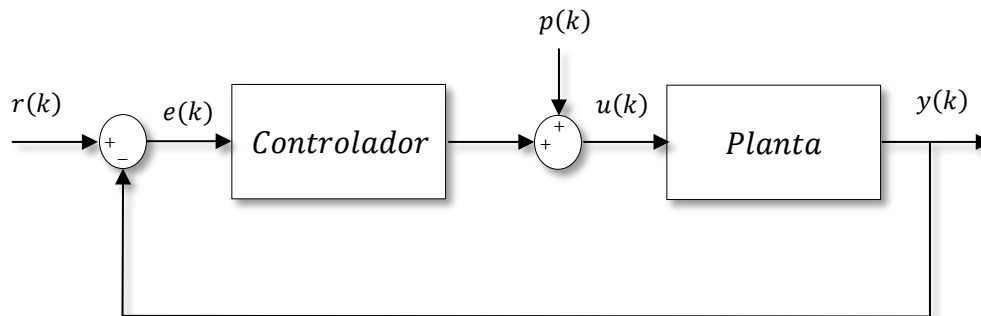
Finalmente se puede llevar la señal a dos valores deseados,  $[\alpha, \beta]$ , utilizando la transformación

$$\overline{PRBS}(k) = (\beta - \alpha)PRBS(k) + \alpha. \quad (2.13)$$

Una vez diseñada la señal de excitación es posible llevar a cabo el experimento de identificación. Dicha prueba puede ser realizada en lazo abierto (Figura 2.5), o en lazo cerrado (Figura 2.6). En ambos casos se agrega una señal de ruido aleatoria  $p(k)$  con el fin de evitar correlaciones entre los datos de entrada y los datos de salida del sistema.



**Figura 2.5: Esquema de Identificación en Lazo Abierto.**



**Figura 2.6: Esquema de Identificación en Lazo Cerrado.**



En el caso del esquema en lazo cerrado, el experimento puede no ser informativo incluso si la entrada  $u(k)$  es persistentemente excitante [58], por lo que se debe asegurar de utilizar las señales  $r(k)$  y  $p(k)$  para excitar adecuadamente el sistema.

Dependiendo del propósito del modelo, una estrategia puede generar mayor información que la otra. En particular, un experimento de identificación en lazo cerrado se realiza cuando la planta es inestable o existen ciertas restricciones que no permiten operar en lazo abierto (condiciones de seguridad, por ejemplo) [58][74]. A su vez, se muestra en [113] que experimentos en lazo cerrado pueden otorgar mejores resultados para el diseño de controladores basados en el modelo obtenido a partir del proceso de identificación.

Si bien una PRBS es adecuada para la identificación de sistemas lineales, dicha señal de excitación puede ser inapropiada para identificar un sistema no lineal (y en particular para los modelos difusos TS, una clase de modelo no lineal), esto debido a que no hay información acerca del comportamiento del sistema para entradas con amplitudes distintas a los dos niveles excitados por la PRBS [10][74]. Para afrontar esto se propone el uso de una señal PRBS de amplitud modulada (APRBS) [73], la cual varía tanto en amplitud como en frecuencia, como se observa en la Figura 2.7, lo que permite capturar información en distintos puntos de operación del proceso. Para el diseño de una APRBS primero se genera una PRBS estándar, luego se divide el rango entre la amplitud mínima y máxima en un número de niveles definido a priori, y posteriormente se asocia aleatoriamente cada amplitud de la PRBS a uno de los niveles definidos anteriormente.

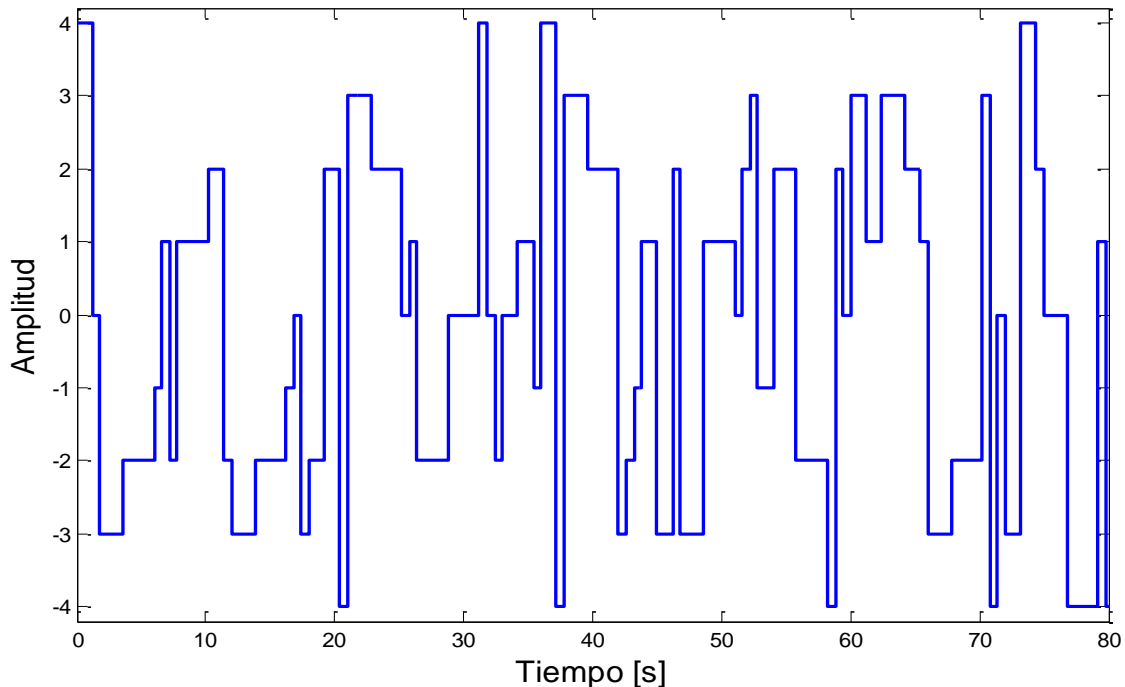


Figura 2.7: Ejemplo de APRBS con Amplitud entre -4 y +4.

Una vez obtenidos los datos experimentales de entrada-salida, éstos deben ser clasificados en tres conjuntos con distinta información: datos de entrenamiento, datos de validación y datos de prueba; esto con el fin de evaluar adecuadamente los modelos generados. El conjunto de entrenamiento se utiliza para determinar los parámetros del modelo. El conjunto de validación permite comparar distintas estructuras de los modelos generados. Finalmente, el conjunto de prueba permite verificar el sobreajuste del modelo óptimo obtenido, evaluándolo en un nuevo conjunto de datos (distintos a los datos del conjunto de entrenamiento y validación), analizando su capacidad de generalización [35]. Cabe destacar que algunos textos en la literatura se refieren al conjunto de prueba como conjunto de validación y viceversa (según la descripción anterior) [90], sin embargo en este trabajo se emplean las definiciones dadas.

No hay una regla general que permita determinar el número de observaciones en cada conjunto, ya que depende de la aplicación y tipo de pruebas que se desean realizar. Sin embargo, en [35] se plantea que una división típica es utilizar 50% de los datos para entrenamiento, 25% para validación y 25% prueba, mientras que en [67] se plantea el uso del 70% de los datos para entrenamiento, 15% de los datos para validación y 15% de los datos para prueba. Luego, se recomienda utilizar algunos de estas propuestas o algún valor intermedio. Adicionalmente, dado que se desean modelar sistemas dinámicos, la selección de datos debe realizarse de tal manera que se respete la causalidad del proceso, además de considerar datos de las distintas condiciones de operación en cada conjunto.

Una vez obtenidos los datos de entrada-salida y dividirlos en los conjuntos antes mencionados, se procede con la siguiente etapa del procedimiento de identificación: seleccionar las variables relevantes del modelo.

## 2.3. Selección de Variables Relevantes

En esta fase se deben seleccionar las variables que actúan como entrada al sistema difuso, es decir, las premisas  $z(k)$  del modelo (2.1)-(2.4). Para ello se debe estudiar qué variables tienen una mayor relación con la salida del sistema.

Esta tarea se puede abordar de diversas maneras. Una de ellas consiste en utilizar el conocimiento dado por la experiencia que se tiene de la planta para hacer la selección. Otra alternativa se basa en evaluar el impacto que tienen las variables candidatas en la salida del modelo. Para ello se puede emplear un análisis de sensibilidad [90], un método heurístico [106], o bien realizar un estudio de correlación/autocorrelación de las variables candidatas [58].

### 2.3.1. Análisis de Sensibilidad

El análisis de sensibilidad consiste en estudiar la influencia de cada premisa sobre la salida del modelo difuso [90]. Para ello se deben calcular las derivadas

$$\xi_i(z(k)) = \frac{dy_{fuzzy}(k)}{dz_i(k)}, \quad i = 1, \dots, p. \quad (2.14)$$

Con  $\xi_i(z(k))$  la sensibilidad de la  $i$ -ésima premisa  $z_i(k)$ ,  $p$  la cantidad total de premisas e  $y_{fuzzy}(k)$  la salida del modelo difuso. Cabe destacar que la derivada en (2.14) se puede calcular explícitamente, dependiendo de la clase de modelo difuso a utilizar. Luego, dichas sensibilidades deben ser evaluadas utilizando el conjunto de datos de entrenamiento, esto para obtener representatividad en distintos puntos de operación, con lo cual se obtiene un vector de índices  $\bar{\xi}_i(z(k))$  para cada entrada al modelo difuso.

Posteriormente, se propone el uso del indicador  $I_i = \mu^2(\bar{\xi}_i(z(k))) + \sigma^2(\bar{\xi}_i(z(k)))$ , donde el primer término de  $I_i$  indica la media al cuadrado de las sensibilidades de la  $i$ -ésima premisa, mientras que el segundo término de  $I_i$  representa la varianza de los datos de sensibilidad de la premisa correspondiente. De esta manera,  $I_i$  evita algún valor sesgado de representatividad de las variables candidatas al utilizar valores estadísticos de las sensibilidades dadas en distintas condiciones de operación. Finalmente, se trabaja con las premisas que mayor índice  $I_i$  posean, descartando las con peor rendimiento. Este procedimiento se puede realizar iterativamente, descartando la variable candidata con menor índice de sensibilidad en cada iteración, y comparando paso a paso el desempeño del modelo difuso hasta encontrar una configuración óptima acorde a un índice de desempeño; o bien se puede definir un umbral a priori y descartar todas las variables candidatas que posean un índice  $I_i$  menor a dicho umbral.

### 2.3.2. Método Heurístico

En general, un modelo difuso puede ser definido a partir de un total de  $p$  variables candidatas:  $z_1(k), \dots, z_p(k)$ ; como se presenta en (2.1)-(2.4). Luego, se tiene un total de  $2^p - 1$  modelos posibles, considerando todas las combinaciones posibles de entradas (y sin considerar el subconjunto vacío de entradas, es decir, se tiene el cuidado de tener al menos una premisa en el modelo).

El método heurístico consiste en seleccionar premisas a partir del conjunto de variables candidatas e incrementar el número de entradas uno a uno, considerando algún criterio. En [106] se utiliza el criterio de regularidad (*Regularity Criterion*, RC), por lo que primero se deben dividir los datos de entrenamiento en dos conjuntos:  $A$  y  $B$ . Luego, el índice RC se define como

$$RC = \frac{1}{2} \left[ \sum_{k=1}^{N_A} \frac{1}{N_A} \left( y^A(k) - y_{fuzzy}^{AB}(k) \right)^2 + \sum_{k=1}^{N_B} \frac{1}{N_B} \left( y^B(k) - y_{fuzzy}^{BA}(k) \right)^2 \right], \quad (2.15)$$

donde  $N_A, N_B$  es el número de datos en los conjuntos  $A$  y  $B$  respectivamente;  $y^A(k), y^B(k)$  son los datos de salida de los conjuntos  $A$  y  $B$ ;  $y_{fuzzy}^{AB}(k)$  es la salida del modelo utilizando los datos del grupo  $A$  estimada por el modelo identificado con los datos del grupo  $B$ ; y  $y_{fuzzy}^{BA}(k)$  es la salida del modelo utilizando los datos del grupo  $B$  estimada por el modelo identificado con los datos del grupo  $A$ .

Luego, los pasos del algoritmo heurístico son los siguientes:

- 1) Identificar  $p$  modelos difusos con una variable de entrada.
- 2) Calcular el  $RC$  para cada modelo y seleccionar el de menor  $RC$ .
- 3) Fijar la variable seleccionada en 2) y agregar otra variable del conjunto de candidatas restante.
- 4) Continuar hasta que  $RC$  aumente.

### 2.3.3. Estudio de Correlación/Autocorrelación

Otra forma de analizar la influencia de una premisa sobre la salida del modelo difuso es estudiar el nivel de correlación que puede existir entre una variable candidata y los datos de salida correspondientes, donde un bajo índice de correlación indica que la variable tiene poca o nula influencia en la salida, por lo que dicha variable debiese ser descartada como premisa del modelo difuso.

Por otro lado, en caso de utilizar modelos autorregresivos, un estudio de correlación entre la salida  $y(k)$  y sus regresores puede permitir identificar los retardos necesarios a considerar en el modelo final. Sin embargo, utilizar sólo este indicador puede sesgar el estudio debido a la dependencia entre un regresor y otro, en otras palabras, la autocorrelación de  $y(k)$  e  $y(k-j)$  se ve afectada por la dependencia existente entre  $y(k-1), y(k-2), \dots, y(k-j+1)$ . Por ello se utiliza la función de autocorrelación parcial (*Partial Autocorrelation Function*, PACF), la cual permite calcular la relación entre  $y(k)$  e  $y(k-j)$  eliminando el efecto de los retardos intermedios [15][65].

Cabe destacar que el uso de muchos regresores en la identificación de sistemas (y en particular para sistemas difusos) puede provocar un sobreajuste en el modelo final y una estimación condicionada a muchos datos. Por otro lado, utilizar pocos regresores puede generar modelos imprecisos para representar las dinámicas no lineales de la planta original [10]. Por lo tanto se sugiere utilizar una cantidad de regresores máxima, escogida a priori o a partir de un estudio de correlación/autocorrelación parcial; ir disminuyendo su valor y comparar el desempeño de los modelos obtenidos mediante

validación cruzada [58], es decir, se entrenan dos modelos con estructuras distintas utilizando el conjunto de datos de entrenamiento, y luego se compara el desempeño de ambas representaciones mediante la evaluación de los modelos utilizando los datos en el conjunto de validación.

Posterior a la selección de variables relevantes es necesario determinar la estructura del modelo difuso a utilizar, por lo que se requiere analizar cual de ellas otorga mejores resultados acorde a algún criterio de desempeño.

## 2.4. Optimización de la Estructura

La estructura del modelo difuso TS descrito por (2.1)-(2.4) está determinada por el tipo de funciones  $f_r(z(k))$ , la forma de las funciones de pertenencia  $MF_i^r$ , y también por el número de reglas  $N_r$  que lo compone. Por lo general, el tipo de modelo local dado por  $f_r(z(k))$  y la forma de las funciones de pertenencia  $MF_i^r$  se escogen con anterioridad, por lo que su influencia recae principalmente en la etapa de identificación de parámetros del modelo. Luego, el objetivo principal de esta fase es determinar el número óptimo de reglas del modelo difuso.

Una opción para determinar el número óptimo de reglas es proponer diversas arquitecturas, entrenarlas para obtener sus parámetros y comparar sus desempeños con el conjunto de datos de validación utilizando algún índice de evaluación, como el RMSE, definido como

$$RMSE = \sqrt{\frac{1}{N} \sum_{k=1}^N (y(k) - y_{fuzzy}(k))^2}. \quad (2.16)$$

donde  $N$  es la cantidad total de datos,  $y(k)$  es la salida de la planta real en el instante  $k$ , e  $y_{fuzzy}(k)$  es la predicción realizada por el modelo difuso en el instante  $k$ .

La optimalidad de los modelos entrenados depende netamente del índice de desempeño utilizado. Luego, si no importa la complejidad, el mejor modelo es aquel que tiene menor RMSE. Sin embargo, es posible que un modelo con peor índice de desempeño, pero menos complejo que el modelo óptimo, pueda obtener resultados aceptables bajo un estándar de rendimiento definido preliminarmente. Dado lo anterior, se podría escoger el modelo menos complejo como el modelo resultante del proceso de identificación, aplicando el llamado principio de parsimonia [99], el cual afirma que no se deben utilizar parámetros adicionales para describir fenómenos dinámicos si es que no es necesario. En la Figura 2.8 se presenta un ejemplo del principio de parsimonia, donde  $W(\theta)$  es un índice de desempeño que mide el ajuste de un modelo al utilizar una cantidad de parámetros  $\theta$ . En la figura se observa que si se define con anterioridad un

rendimiento esperado  $\bar{W} > W(\hat{\theta}_N) > W(\theta_0) = \min_{\theta} W(\theta)$ , se deberían considerar los parámetros  $\hat{\theta}_N$  en vez de los parámetros óptimos  $\theta_0$ , debido a que un modelo más simple cumple con el estándar de rendimiento deseado.

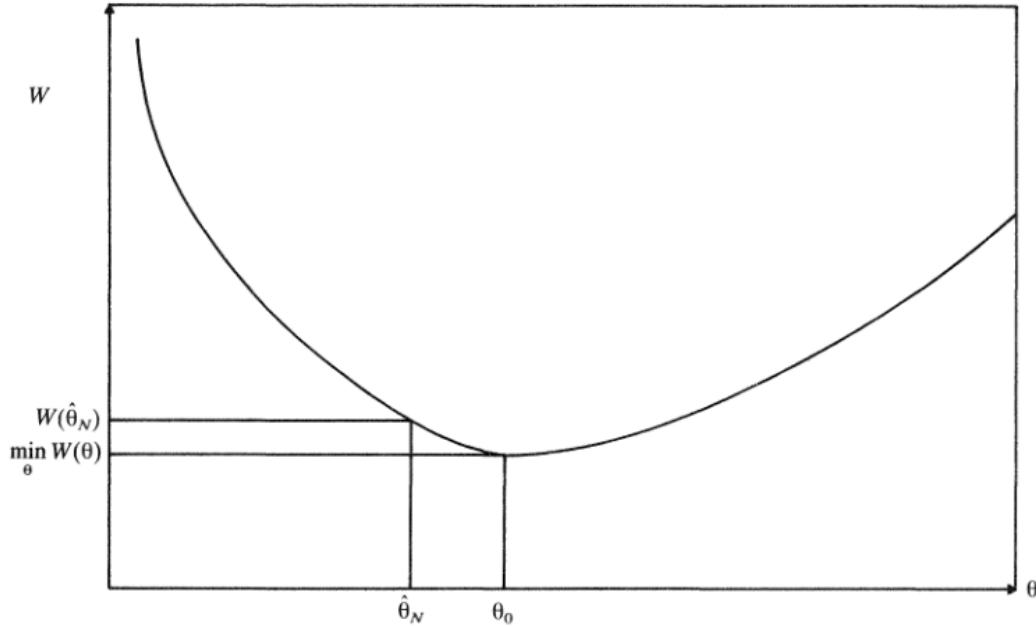


Figura 2.8: Principio de Parsimonia [99].

Otra forma de determinar el número óptimo de reglas de un modelo difuso es realizando un análisis basado en los valores y vectores propios de las matrices de covarianza difusa de los *clusters* obtenidos al emplear algoritmos de *clustering* basados en dichas matrices (un ejemplo de estos algoritmos se ve en más detalle en la Sección 2.5.1). En [10] y [121] se propone la técnica *Compatible Cluster Merging*, CCM, la cual se basa en la idea de combinar dos agrupaciones que estén relativamente cerca entre sí y apunten en la misma dirección. En la Figura 2.9 se ilustra la idea detrás del método CCM.

Para lograr la combinación de *clusters* compatibles se verifican las siguientes condiciones:

$$\begin{aligned}
 |\phi_{in}^T \phi_{jn}| &\geq C_1 \\
 \left| \frac{\phi_{in}^T + \phi_{jn}^T}{2} \frac{v_i - v_j}{\|v_i - v_j\|} \right| &\leq C_2, \\
 \frac{\|v_i - v_j\|}{\sqrt{\lambda_{i1}} + \sqrt{\lambda_{j1}}} &\leq C_3
 \end{aligned} \tag{2.17}$$

donde  $\phi_{in}$  representa el vector propio asociado al menor valor propio de la matriz de covarianza difusa del  $i$ -ésimo *cluster*  $F_i$ ,  $\lambda_{i1}$  es el mayor valor propio de  $F_i$ , y  $v_j$  es el centro del *cluster*.  $C_1, C_2, C_3$  son valores umbrales escogidos por el usuario, donde  $C_1$  debe ser cercano a 1,  $C_2$  debe ser cercano a 0, y  $C_3 \in [2,4]$ , rango que se deriva a partir de una interpretación geométrica en  $R^2$ . La primera condición en (2.17) indica que los *clusters* deben ser aproximadamente paralelos; la segunda condición establece que la recta normal a los hiperplanos debe ser ortogonal a la línea que conecta los centros de los *clusters*; y la tercera condición indica que los centros de los *clusters* deben estar relativamente cerca. Luego, si dos agrupaciones cumplen con las condiciones en (2.17), éstas pueden ser combinadas en una sola, reduciendo el número total de *clusters*, y en consecuencia el número de reglas del modelo.

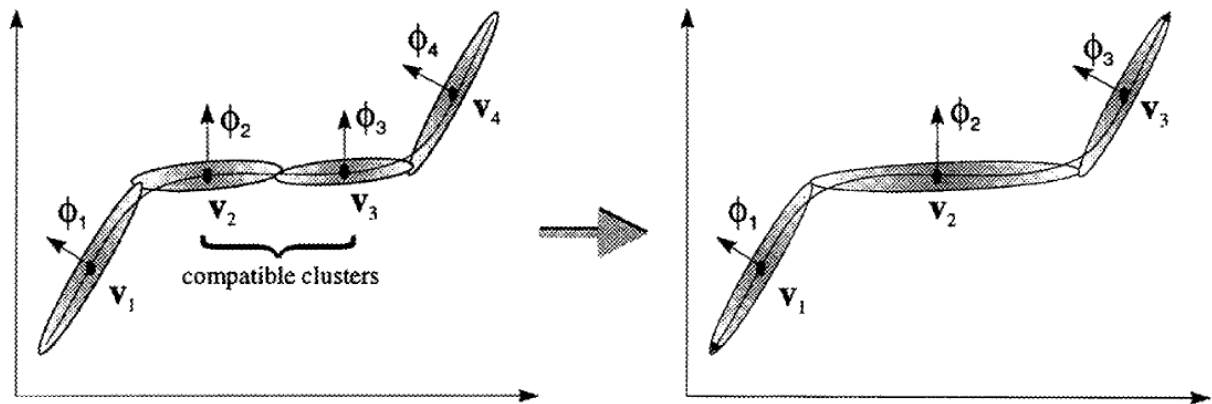


Figura 2.9: Combinación de *Clusters* Compatibles [10].

Cabe destacar que un proceso debería ser representado por una jerarquía de modelos, yendo desde los más simples a los más complejos [8]. Los modelos simples son utilizados para estudiar a grosso modo el comportamiento del sistema, mientras que los modelos complejos son usados para una corroboración detallada del desempeño del sistema de control. El objetivo entonces es escoger el modelo correcto para cada propósito en específico.

Una vez seleccionada la estructura del modelo difuso a utilizar, se requiere determinar sus parámetros asociados, los cuales se conforman por dos grupos: los parámetros de las premisas, y los parámetros de las consecuencias.

## 2.5. Identificación de Parámetros

En esta etapa se deben identificar los parámetros de las premisas del modelo difuso, correspondientes a las funciones de pertenencia  $MF_i^r$ , e identificar los parámetros de las consecuencias, correspondientes a la función  $f_r(z(k))$ . En algunos enfoques reportados

en la literatura se plantea la identificación de los parámetros de las premisas y consecuencias como un único problema de optimización, donde se busca ajustar todos los parámetros con el fin de minimizar una función de costos [4][82]. Sin embargo, identificar los parámetros de las premisas y consecuencias por separado simplifica la construcción del modelo, permitiendo resolver problemas más simples, pero cayendo en el riesgo de obtener una solución subóptima. No obstante, simplificar el problema de identificación de parámetros permite la inclusión de funciones más complejas en las consecuencias sin aumentar en demasía la complejidad del problema [56]. Dado lo anterior, se opta por describir la identificación de parámetros como problemas separados. A continuación se detallan ambos procedimientos.

### 2.5.1. Parámetros de las Premisas

Para identificar los parámetros de las premisas es necesario emplear técnicas de *clustering* difuso. Existen diversos algoritmos para realizar dicha tarea, como Fuzzy c-Means, *clustering* con prototipos lineales y *clustering* con matrices de covarianza difusa [10]. Cada uno de ellos busca encontrar los centros de los *clusters* y los grados de pertenencia de los datos mediante la minimización del siguiente funcional de costos (o una variación de éste):

$$J(Z, U, V) = \sum_{i=1}^c \sum_{k=1}^N (\mu_{ik})^m D_{ikA}^2, \quad (2.18)$$

$$D_{ikA}^2 = (z_k - v_i)^T A (z_k - v_i), \quad (2.19)$$

donde  $Z$  representa el conjunto de datos  $z_k \in R^n$ ;  $U$  es la matriz de partición de los datos, la cual contiene los grados de pertenencia  $\mu_{ik}$  del  $k$ -ésimo dato al  $i$ -ésimo *cluster*;  $V$  contiene los centros de los *clusters*  $v_i$ ;  $m \in [1, \infty[$  es el factor de difusidad, el cual determina el grado de borrosidad de los clusters, es decir, a medida que  $m \rightarrow 1$  la partición se vuelve menos difusa ( $\mu_{ik} \in \{0,1\}$ ), mientras que si  $m \rightarrow \infty$  la partición se vuelve completamente difusa, por lo general se escoge  $m = 2$  [10];  $c$  y  $N$  indican la cantidad de *clusters* y datos, respectivamente;  $D_{ikA}^2$  es el cuadrado de la desviación del dato respecto al centro; y  $A$  es una matriz positiva definida que induce la norma de distancia a utilizar. La medida de disimilitud en la función  $J(Z, U, V)$  puede ser vista como una medida de la varianza total de los datos  $z_k$  desde los centros  $v_i$ .

En particular, dentro de la categoría de *clustering* con matrices de covarianza difusa, se tiene el algoritmo Gustafson-Kessel (GK) [32]. Este algoritmo destaca debido a que incluye en su formulación distancias adaptivas, lo cual permite encontrar agrupaciones con distinta forma. Para lo anterior es necesario modificar la métrica de distancia en (2.19) a la forma:

$$D_{ikA_i}^2 = (z_k - v_i)^T A_i (z_k - v_i). \quad (2.20)$$



Las matrices de normas inducidas  $A = (A_1, \dots, A_c)$  pasan a ser variables del problema de optimización. Como  $J(Z, U, V, A)$  es lineal respecto a  $A$ , bastaría que la norma de  $A$  tienda a cero para minimizar la función objetivo, lo cual invalida la influencia de las demás variables de optimización. Luego, para evitar el problema descrito anteriormente, se incluyen restricciones sobre el determinante de las matrices  $A_i$ , dadas por:

$$\det(A_i) = \rho_i, \quad \rho_i > 0, \quad \forall i, \quad (2.21)$$

lo que permite a  $A_i$  optimizar la forma del *cluster*, mientras su volumen o tamaño permanece constante. Usualmente  $\rho_i = 1, \forall i$ .

Aplicando el método de multiplicadores de Lagrange se puede obtener soluciones para minimizar el funcional en (2.18). Luego, dada alguna inicialización del espacio de partición  $U$ , el algoritmo para la  $l$ -ésima iteración queda definido como:

**PASO 1: Calcular los centros.**

$$v_i^{(l)} = \frac{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^m z_k}{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^m}, \quad 1 \leq i \leq c. \quad (2.22)$$

**PASO 2: Calcular matrices de covarianza difusa  $F_i$ .**

$$F_i^{(l)} = \frac{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^m (z_k - v_i^{(l)})(z_k - v_i^{(l)})^T}{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^m}, \quad 1 \leq i \leq c. \quad (2.23)$$

**PASO 3: Calcular distancias  $D_{ikA_i}^2$ .**

$$D_{ikA_i}^2 = (z_k - v_i^{(l)})^T A_i (z_k - v_i^{(l)}), \quad 1 \leq i \leq c. \quad (2.24)$$

$$A_i = (\rho_i \det(F_i))^{1/n} F_i^{-1}$$

**PASO 4: Actualizar  $U$ .**

$$\mu_{ik}^{(l)} = \frac{(D_{ikA_i}^2)^{-\frac{1}{m-1}}}{\sum_{j=1}^c (D_{jkA_i}^2)^{-\frac{1}{m-1}}}, \quad 1 \leq i \leq c, \quad 1 \leq k \leq N. \quad (2.25)$$

El algoritmo se detiene cuando se cumple que  $\|U^{(l)} - U^{(l-1)}\| < \epsilon$ , con  $\epsilon$  un valor pequeño (usualmente  $\epsilon = 10^{-3}$  o menor).

Cabe destacar que las matrices de covarianza difusas,  $F_i$ , y también las matrices de normas inducidas  $A_i$ , debido a la relación en (2.24), están asociadas al tamaño y forma de los *clusters* a través de sus vectores propios  $\phi_{ij}$ , y sus valores propios  $\lambda_{ij}$ , como se puede observar en la Figura 2.10.

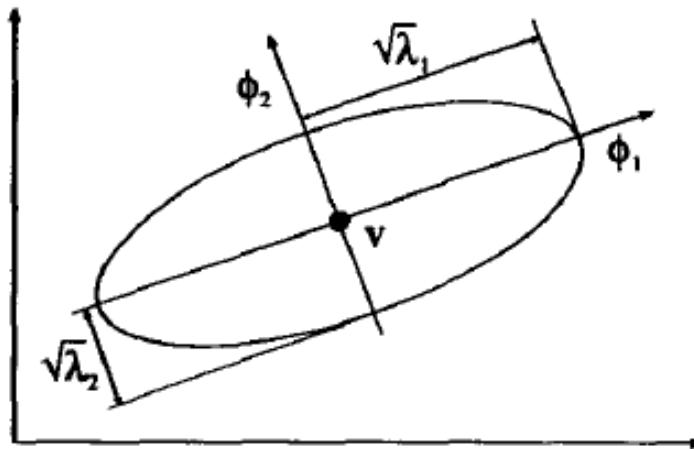


Figura 2.10: Curva Descrita por  $(z - v)^T F^{-1}(z - v) = 1$  [10].

GK es adecuado para identificación debido a que es menos sensible al escalamiento (normalización) de los datos e inicialización de sus variables, además posee la capacidad para detectar *clusters* de distinta forma y orientación. Por otro lado, el algoritmo puede resultar lento computacionalmente al tener que invertir matrices en cada iteración, además no puede detectar *clusters* que difieran mucho de tamaño (dada la restricción sobre las matrices  $A_i$ ), y puede tener problemas numéricos al manejar datos que estén linealmente correlacionados. Para esto último, en [11] se plantean un par de modificaciones del algoritmo original.

Para lidiar con problemas de singularidad de  $F_i$  se plantea el estudio del *condition number* de la matriz de covarianza difusa, correspondiente a la razón entre el mayor y el menor valor propio de  $F_i$ . Un valor muy alto de dicha razón indica que la matriz es (casi) singular, por lo que se propone reconstruir  $F_i$  de tal forma de limitar sus valores propios.

Para ello primero se extraen los valores y vectores propios de  $F_i$ :  $\lambda_{ij}$  y  $\phi_{ij}$  respectivamente. Sea  $\lambda_{i,max}$  el mayor valor propio, luego se establece

$$\lambda_{ij} = \frac{\lambda_{i,max}}{\beta}, \quad \text{si } \frac{\lambda_{i,max}}{\lambda_{ij}} > \beta, \quad \forall j, \quad (2.26)$$

donde usualmente  $\beta = 10^{15}$ . Posteriormente se reconstruye  $F_i$  usando

$$F_i = [\phi_{i1}, \dots, \phi_{in}] \text{diag}(\lambda_{i1}, \dots, \lambda_{in}) [\phi_{i1}, \dots, \phi_{in}]^{-1}. \quad (2.27)$$

Dado lo anterior, puede darse el caso donde los *clusters* resultantes sean muy alargados en la dirección del mayor valor propio, teniendo poca relación con la distribución real de los datos. Para esto se propone añadir una “matriz de identidad escalada” a la formulación de la matriz de covarianza, resultando

$$F_i^{\text{new}} = (1 - \gamma)F_i + \gamma \det(F_0)^{1/n} I, \quad (2.28)$$

donde  $\gamma \in [0,1]$  es un ponderador y  $F_0$  es la matriz de covarianza de los datos. El efecto de  $\gamma$  debe ser estudiado dependiendo de cada caso, por lo que se debe ajustar acorde a cada aplicación.

Una vez realizado el *clustering* es posible proyectar las agrupaciones en el espacio de entrada y ajustar funciones paramétricas para describir los conjuntos difusos  $MF_r^i$ . En particular, considérese funciones de pertenencia gaussianas dadas por

$$MF_r^i(z_i(k)) = \exp\left(-0.5 \cdot \left(a_{r,i} \cdot (z_i(k) - b_{r,i})\right)^2\right), \quad (2.29)$$

donde  $MF_r^i(z_i(k))$  es la función de pertenencia de la  $i$ -ésima premisa  $z_i(k)$  al  $r$ -ésimo *cluster*,  $a_{r,i}$  es el inverso de la desviación estándar de los datos ajustados por la gaussiana, y  $b_{r,i}$  representa su media. Luego, es posible utilizar los centros de los *clusters* otorgados por el algoritmo GK como la media de la función de pertenencia, y emplear los valores propios de las matrices de normas inducidas,  $A_i$ , para determinar la desviación estándar; esto debido a la forma de las agrupaciones difusas, como se observa en la Figura 2.10. En el caso de la desviación estándar se tiene que las distancias dentro de cada *cluster* están dadas por (2.20), luego la ecuación del hiperelipsoide que define al conjunto es

$$(z_k - v_i)^T A_i (z_k - v_i) = 1. \quad (2.30)$$

De (2.23) y (2.24) se tiene que  $A_i$  es simétrica por definición, por lo que es posible descomponer dicha matriz como  $A_i = \Phi_i \Lambda_i \Phi_i^{-1}$ , donde  $\Phi_i$  contiene los vectores propios de  $A_i$  y  $\Lambda_i$  es una matriz diagonal con sus valores propios  $\lambda_{A_{ij}}$ ,  $j = 1, \dots, n$ . Además,  $\Phi_i^{-1} = \Phi_i^T$  debido a que es una matriz ortonormal. Luego, utilizando el cambio de base  $\bar{z}_{kl} = \Phi_i^T (z_k - v_i)$  se obtiene

$$\bar{z}_{kl}^T \Lambda_i \bar{z}_{kl} = \lambda_{A_{i1}} \bar{z}_{kl1}^2 + \dots + \lambda_{A_{in}} \bar{z}_{kln}^2 = 1, \quad (2.31)$$

ecuación que describe un hiperelipsoide en  $R^n$  centrado en  $v_i$ , orientado por los vectores propios de  $A_i$ , y con semiejes  $1/\sqrt{\lambda_{A_{ij}}}$ ,  $j = 1, \dots, n$ . Por lo tanto, para determinar la

desviación estándar se calcula el inverso de la raíz de los valores propios de  $A_i$ ,  $1/\sqrt{\lambda_{A_{ij}}}$ ,  $j = 1, \dots, n$ ; se proyectan en el espacio cartesiano utilizando la matriz de vectores propios,  $\Phi_i$ , y se toma el valor absoluto de dicha proyección (para no obtener valores negativos en caso de que el vector propio de  $A_i$  apunte en sentido contrario a los ejes cartesianos). Como existen  $n$  valores propios de la matriz  $A_i$ , se considera como desviación estándar la proyección con mayor valor, de esta forma se logra representar la mayor cantidad de datos pertenecientes al *cluster*.

En resumen, es posible determinar los parámetros de las funciones de pertenencia gaussianas a partir de los centros de los *clusters* y los valores propios de las matrices de normas inducidas  $A_i$ . Una vez determinados dichos valores se procede con determinar los parámetros de los consecuentes del modelo difuso.

## 2.5.2. Parámetros de las Consecuencias

La identificación de los parámetros de las consecuencias (los parámetros asociados a los modelos locales) se realiza mediante el método de mínimos cuadrados (*Least Square*, LS). Para ello se busca resolver el siguiente problema de optimización:

$$\min_{\theta} J(\theta) = \sum_{k=1}^N \left( y(k) - y_{fuzzy}(k, \theta) \right)^2, \quad (2.32)$$

donde  $\theta$  son los parámetros de las consecuencias, es decir, los parámetros de las funciones  $f_r(z(k))$ ;  $y(k)$  es la salida medida del proceso en el instante  $k$ ;  $y_{fuzzy}(k, \theta)$  representa la predicción realizada por el modelo difuso utilizando la expresión (2.4), usando  $\theta$  como parámetros de los modelos locales; y  $N$  es la cantidad de datos.

Cabe destacar que la identificación de los parámetros de las consecuencias usualmente se realiza a un paso, sin embargo, dado a que este trabajo pretende validar la metodología propuesta mediante el diseño de estrategias de control predictivo, identificar un modelo a más pasos puede traer ventajas al obtener mejores predicciones en horizontes mayores [94]. Por lo tanto, en dicho caso se propone calcular los parámetros de las consecuencias a través del problema de optimización:

$$\min_{\theta} J(\theta) = \sum_{k=1}^N \left( y(k_0 + k) - y_{fuzzy}(k_0 + k | k_0, \theta) \right)^2, \quad (2.33)$$

con  $y_{fuzzy}(k_0 + k | k_0, \theta)$  la predicción del modelo difuso a  $k$  pasos, a partir de condiciones iniciales dadas en un instante  $k_0$ . La diferencia con el enfoque a un paso radica en que las premisas se actualizan utilizando la predicción del modelo difuso en lugar del dato medido de la salida del sistema. Cabe destacar que el método propuesto

anteriormente difiere del concepto de predicción a  $n$  pasos, donde se emplea un horizonte deslizante para predecir el comportamiento del sistema a una cantidad de  $n$  pasos fija. En este caso se predice el comportamiento del sistema utilizando un horizonte dado por la cantidad de datos presentes. Dicho lo anterior, en este trabajo se opta por denominar **Identificación a  $N$  pasos** al método propuesto.

Finalmente, para obtener los parámetros de las consecuencias se puede emplear cualquier algoritmo de optimización para resolver (2.32) o (2.33). Sin embargo, de los planteamientos para obtener los parámetros de las consecuencias expuestos anteriormente, es posible obtener una solución analítica del problema de optimización cuando se realizan predicciones a un paso y las consecuencias del modelo difuso son lineales en los parámetros, es decir, los modelos locales pueden ser escritos como  $f_r(z(k)) = \theta_r^T z(k)$ , con  $\theta_r^T = [\theta_{r,1}, \dots, \theta_{r,n}]$ . Dado lo anterior, la salida del modelo difuso puede ser reescrita como

$$y_{fuzzy}(\theta) = [y_{fuzzy}(1, \theta), \dots, y_{fuzzy}(N, \theta)]^T = Z_{fuzzy}\theta, \quad (2.34)$$

donde

$$Z_{fuzzy} = \begin{bmatrix} h_1(z(1))z_1(1) \dots h_{N_r}(z(1))z_1(1) & \dots & h_1(z(1))z_n(1) \dots h_{N_r}(z(1))z_n(1) \\ \vdots & \ddots & \vdots \\ h_1(z(N))z_1(N) \dots h_{N_r}(z(N))z_1(N) & \dots & h_1(z(N))z_n(N) \dots h_{N_r}(z(N))z_n(N) \end{bmatrix}, \quad (2.35)$$

y  $\theta^T = [\theta_{1,1}, \dots, \theta_{N_r,1}, \dots, \theta_{1,n}, \dots, \theta_{N_r,n}]$  contiene los parámetros de todas las reglas. Dado lo anterior, se puede reescribir la función de costos en (2.32) de manera matricial, resultando

$$J(\theta) = (y - y_{fuzzy}(\theta))^T (y - y_{fuzzy}(\theta)) = (y - Z_{fuzzy} \cdot \theta)^T (y - Z_{fuzzy} \cdot \theta), \quad (2.36)$$

con  $y = [y(1), \dots, y(N)]^T$ .

Luego, la solución del problema de optimización dado por la minimización de (2.36) viene dada por

$$\theta^* = (Z_{fuzzy}^T \cdot Z_{fuzzy})^{-1} Z_{fuzzy}^T \cdot y. \quad (2.37)$$

Después de abordadas todas las etapas del proceso de identificación se prueba el modelo obtenido, si éste no cumple con los requisitos deseados se puede volver a iterar en alguna de las fases anteriores hasta obtener un modelo satisfactorio.

---

Luego de detallado el proceso de identificación de sistemas difusos, en el siguiente capítulo se presenta el estudio de la controlabilidad de los sistemas dinámicos, para luego proceder a describir las propuestas para la obtención de modelos difusos TS que posean la propiedad de ser controlables, o en su defecto, que posean un rango de controlabilidad mayor que los modelos de predicción lineales.

### 3. Controlabilidad

Para introducir el concepto de controlabilidad, considérese un modelo discreto de la forma

$$x(k+1) = f(x(k), u(k)), \quad x(k) \in M, \quad (3.1)$$

con  $M$  el espacio de estado,  $x(k)$  el estado y  $u(k)$  la entrada. Luego, la controlabilidad del sistema en (3.1) se define a partir de su capacidad de alcanzar un estado deseado  $x(k_1) = x_1$  cualquiera a partir de cualquier estado inicial  $x(k_0) = x_0$ , mediante una secuencia de acciones de control  $u(k_0), \dots, u(k_0 + k_1)$  admisibles, es decir, que dichas acciones de control cumplan con la ecuación de estado en (3.1) [46][95]. No obstante, existen sistemas que no necesariamente satisfacen esta definición en su totalidad, por lo que aparecen conceptos que relajan estas condiciones.

Sea  $R(x_0, T)$  el conjunto de todos los puntos que pueden ser alcanzados desde  $x_0$  en un tiempo finito  $T$  utilizando alguna secuencia de acciones de control admisibles. Además, sea

$$R(x_0) = \bigcup_{0 \leq k} R(x_0, k) \quad (3.2)$$

el conjunto de todos los estados que pueden ser alcanzados a partir de  $x_0$  en un tiempo  $k$  dado. Luego se dice que el sistema en (3.1) es **accesible desde  $x_0$**  si el interior del conjunto  $R(x_0) \neq \emptyset$  [54]. En otras palabras, de la definición de accesibilidad se puede inferir que este concepto aparece como una condición que relaja la definición de controlabilidad, en la que sólo se exige que el sistema sea controlable para un subconjunto de puntos.

Por otro lado, se dice que (3.1) es **controlable en  $x_0$**  si  $R(x_0) = M$ , es decir, si el conjunto de estados alcanzables es igual al espacio de estados completo. Se dice también que el sistema es **localmente controlable en  $x_0$**  si  $R(x_0)$  es una vecindad cercana a  $x_0$  [38]. Las definiciones anteriores pueden ser extendidas al caso en que se cumplen  $\forall x_0 \in M$ , donde se dice que el sistema es **(completamente) controlable**, o **localmente controlable**, respectivamente. La diferencia entre las dos definiciones anteriores se puede apreciar gráficamente en la Figura 3.1.

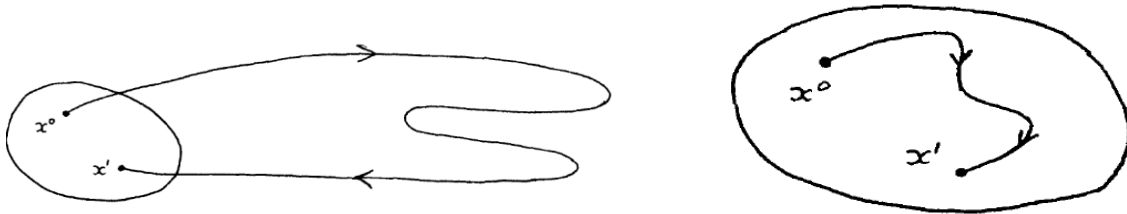


Figura 3.1: Izquierda: Sistema Controlable. Derecha: Sistema Localmente Controlable [43].

Cabe destacar que si un sistema no lineal es no accesible, entonces es no controlable; a su vez, si el sistema es accesible, no necesariamente es controlable.

Dadas las definiciones anteriores, se busca encontrar condiciones matemáticas que permitan determinar la propiedad de controlabilidad/accesibilidad de los sistemas difusos para poder aplicarlas en el proceso de identificación de sistemas. A continuación se enuncian dichas condiciones tanto para sistemas lineales como sistemas difusos.

### 3.1. Controlabilidad de Sistemas Lineales

Se dice que un sistema lineal del tipo

$$x(k+1) = Ax(k) + Bu(k), \quad x(k) \in R^n, \quad (3.3)$$

es (completamente) controlable si se cumple con la condición de rango de Kalman para controlabilidad dada por [46]

$$\text{rank}(\mathbb{C}) = n, \quad (3.4)$$

$$\mathbb{C} = [B \quad AB \quad A^2B \quad \dots \quad A^{n-1}B], \quad (3.5)$$

donde  $\mathbb{C}$  se denomina matriz de controlabilidad y  $n$  es el orden del sistema. La condición anterior nace de la solución de la ecuación de diferencias (3.3), la cual está dada por

$$x(n) = A^n x(0) + \sum_{i=0}^{n-1} A^{k-1-i} B u(i). \quad (3.6)$$

Luego,  $x(n)$  es una combinación lineal entre las acciones de control pasadas y los componentes  $B, AB, A^2B$ , etc. Por lo tanto, si el conjunto de vectores  $\{A^{k-1-i}B, i = 0, 1, \dots, n\}$  son linealmente independientes, es posible formar una base del espacio de estados, y de esta forma generar cualquier punto deseado mediante la elección adecuada de los coeficientes de la combinación lineal, los cuales en este caso son las acciones de control  $u(i)$ .



En particular, si el sistema estudiado es SISO, la matriz de controlabilidad  $\mathbb{C}$  es cuadrada, por lo que la condición de rango puede ser abordada verificando el determinante de la matriz, es decir, si  $\det(\mathbb{C}) \neq 0$ , entonces el sistema es controlable.

Cabe destacar que la controlabilidad de un sistema lineal también puede ser analizada desde el punto de vista de la función de transferencia  $G(z)$  del sistema. Para ello, considérese

$$G(z) = \frac{Y(z)}{U(z)} = \frac{\theta_{n_y+1}z^{-1} + \dots + \theta_{n_y+n_u}z^{-n_u}}{1 - \theta_1z^{-1} - \dots - \theta_{n_y}z^{-n_y}} = \frac{\theta_{n_y+1}z^{n_y-1} + \dots + \theta_{n_y+n_u}z^{n_y-n_u}}{z^{n_y} - \theta_1z^{n_y-1} - \dots - \theta_{n_y}}, \quad (3.7)$$

con  $n_y > n_u$ . Se puede estudiar la controlabilidad de (3.7) analizando si el numerador y el denominador de  $G(z)$  son coprimos, ya que en caso de existir un factor común se puede perder la propiedad de controlabilidad.

A modo de ilustrar la aseveración anterior, considérese el sistema

$$x(k+1) = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} x(k) + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} u(k), \quad y(k) = [c_1 \quad c_2]x(k), \quad (3.8)$$

donde se identifica  $A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$ ,  $B = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$  y  $C = [c_1 \quad c_2]$ . La matriz de controlabilidad del sistema anterior está dada por

$$\mathbb{C} = \begin{bmatrix} b_1 & a_{11}b_1 + a_{12}b_2 \\ b_2 & a_{21}b_1 + a_{22}b_2 \end{bmatrix} \rightarrow \det(\mathbb{C}) = a_{21}b_1^2 + a_{22}b_1b_2 - a_{11}b_1b_2 - a_{12}b_2^2, \quad (3.9)$$

Luego, para que el sistema sea no controlable se debe cumplir que  $\det(\mathbb{C}) = 0$ . Si se impone  $b_2 = 0$ , entonces  $a_{21}b_1^2 = 0 \rightarrow a_{21} = 0$  (note que si  $b_1 = 0$  entonces el sistema sería autorregresivo solamente). Dado lo anterior, si se lleva el sistema a su forma de función de transferencia se tiene

$$G(z) = C(zI - A)^{-1}B = C \frac{1}{(z - a_{11})(z - a_{22})} \begin{pmatrix} z - a_{22} & a_{12} \\ 0 & z - a_{11} \end{pmatrix} \begin{pmatrix} b_1 \\ 0 \end{pmatrix} = \frac{c_1b_1(z - a_{22})}{(z - a_{11})(z - a_{22})} = \frac{c_1b_1}{(z - a_{11})}. \quad (3.10)$$

En otras palabras, la pérdida de controlabilidad se debe a una anulación entre ceros y polos del sistema, lo que provoca que el sistema en estudio en realidad se comporte como uno de orden menor [27]. No obstante lo anterior, la existencia de factores comunes entre el numerador y denominador de una función de transferencia también puede deberse a que el sistema es no observable, por lo que la asegurar que ambos

polinomios de una función de transferencia sean coprimos es una condición suficiente para corroborar la controlabilidad, mas no necesaria.

Dado lo anterior, una forma de estudiar si los polinomios que conforman la función de transferencia son coprimos es mediante la matriz de Sylvester [16]. Dada la función de transferencia en (3.7), la matriz de Sylvester entre  $Y(z)$  y  $U(z)$ ,  $S(Y(z), U(z))$ , se formula como

$$S(Y(z), U(z)) = \begin{bmatrix} 1 & 0 & \dots & 0 & \theta_{n_y+1} & 0 & \dots & 0 \\ -\theta_1 & 1 & & & \vdots & \theta_{n_y+1} & & \vdots \\ \vdots & -\theta_1 & \ddots & \vdots & \theta_{n_y+n_u} & \vdots & \ddots & 0 \\ -\theta_{n_y} & \vdots & & & 0 & \theta_{n_y+n_u} & & \theta_{n_y+1} \\ 0 & -\theta_{n_y} & \ddots & 0 & \vdots & 0 & \ddots & \vdots \\ & 0 & & 1 & 0 & \vdots & & \theta_{n_y+n_u} \\ \vdots & \vdots & \ddots & -\theta_1 & 0 & 0 & \ddots & 0 \\ & & & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & -\theta_{n_y} & 0 & 0 & \dots & 0 \end{bmatrix}. \quad (3.11)$$

Cabe destacar que los ceros que aparecen luego del término  $\theta_{n_y+n_u}$  son los coeficientes asociados a los factores  $z^{n_y-n_u-1}, \dots, z^0$  del numerador en (3.7), es decir, es un vector de  $(n_y - n_u)$  ceros. Otro aspecto a tener en cuenta es que  $S(Y(z), U(z))$  es de dimensión  $(2n_y - 1) \times (2n_y - 1)$ . Luego, los polinomios  $Y(z)$  y  $U(z)$  poseen algún factor en común si  $\det(S(Y(z), U(z))) = 0$ , es decir, si la matriz de Sylvester es singular entonces el sistema lineal puede ser no controlable.

Por otra parte, en [89] se plantean condiciones de controlabilidad para sistemas lineales discretos con parámetros variables en el tiempo, es decir, el modelo lineal se rige por la ecuación  $x(k+1) = A(k)x(k) + B(k)u(k)$ , por lo que los enfoques anteriores dejan de ser válidos. En este caso el análisis se basa en la matriz de transición de estados,  $\Phi(k, k_0) = A(k-1)A(k-2) \dots A(k_0+1)A(k_0)$ , donde se dice que el sistema lineal discreto es controlable si y sólo si existe un  $s > k_0$  tal que  $\Phi(s, k_0) = 0$ , es decir, el estudio requiere evaluar la matriz de transición de estados y verificar si se cumple la condición anterior, derivando en un método poco apto para un algoritmo de identificación, debido a que en caso de que si un modelo es no controlable, entonces nunca se podría encontrar un  $s$  tal que lo anterior se cumpla.

Aunque la condición de rango de Kalman para controlabilidad o el método de la matriz de Sylvester son directos para determinar si un sistema lineal discreto de parámetros constantes es controlable o no, dicha condición no puede ser aplicada directamente a un modelo difuso TS, debido a que éste es una clase de sistemas no lineales, y por lo tanto se requiere otro tipo de análisis.

## 3.2. Controlabilidad de Sistemas Difusos

Si bien se ha demostrado que los modelos difusos logran representar de buena manera el comportamiento de plantas con fuertes no linealidades, siendo una potente herramienta para la identificación de sistemas, dichos modelos pueden resultar limitados desde el punto de vista de controlabilidad, de tal manera que pueden existir puntos de operación en donde la representación utilizada no permita aplicar una acción de control adecuada y en consecuencia esto se traduzca en una pérdida de desempeño en lazo cerrado. Por esto el estudio de esta propiedad en los modelos difusos TS puede resultar beneficioso al momento de diseñar estrategias de control.

En [18] se aborda el tema de controlabilidad de una clase de modelos difusos dinámicos para representar tanto plantas lineales como no lineales. El modelo difuso consiste en un modelo de Mamdani con conjuntos difusos de tipo triangular, donde los conjuntos de salidas quedan determinados por los parámetros de los conjuntos difusos de las premisas correspondientes a cada regla. El análisis de controlabilidad propuesto, tanto para sistemas lineales como no lineales, se basa en el conocimiento del modelo original de la planta, es decir, se deben conocer a priori las funciones que rigen la dinámica del sistema real. Dado lo anterior, las condiciones para garantizar la controlabilidad planteadas son inutilizables desde el punto de vista de identificación, donde no se conoce la planta real, y en consecuencia no sería posible evaluar dichas condiciones.

En [45] se opta por modificar el enfoque difuso clásico impuesto por Takagi & Sugeno con el fin de analizar las características del modelo, como estabilidad, controlabilidad, observabilidad, etc., de una manera más simple. Lo que propone el autor es disponer de una base de reglas donde cada consecuencia sea un modelo lineal obtenido en distintos puntos de operación. Luego de calculados los grados de activación de cada regla se escoge como salida del modelo difuso el modelo local que mayor grado de activación posea, es decir, no existe una salida dada por la suma ponderada de los modelos locales como el enfoque Takagi-Sugeno, sino que se utiliza el modelo más representativo solamente. Por lo tanto, dado que los modelos locales son lineales, basta que cada una de las consecuencias sea diseñada de tal forma que sean estables, completamente controlables, completamente observables, etc. Si bien el estudio de controlabilidad se simplifica significativamente, el modelo difuso propuesto pierde representatividad en los tramos intermedios entre un punto de operación y otro.

A su vez, la controlabilidad de los sistemas difusos TS ha sido estudiada desde diferentes enfoques, sin embargo, sólo se han reportado resultados para modelos difusos TS de tiempo continuo. Por ejemplo, en [69] se plantea el estudio del modelo difuso como una clase de sistema lineal con parámetros variables en el tiempo, de la forma

$$\begin{aligned} \dot{x}(t) &= A(t)x(t) + B(t)u(t), & x(t) &\in R^n \\ A(t) &= \sum_{r=1}^{N_r} h_r(z(t))A_r, & B(t) &= \sum_{r=1}^{N_r} h_r(z(t))B_r, \end{aligned} \quad (3.12)$$

donde cada modelo local del sistema difuso está dado por  $\dot{x}_r(t) = A_r x(t) + B_r u(t)$ ,  $z(t)$  son las premisas del modelo,  $N_r$  la cantidad de reglas y  $h_r(z(t))$  el grado de activación normalizado de la  $r$ -ésima regla. Luego, es posible aplicar condiciones de controlabilidad similares a las expuestas en la Sección 3.1, pero en este caso, dado que los parámetros son variantes en el tiempo, se reformula la matriz de controlabilidad  $\mathbb{C}(t)$  como [95]

$$\begin{aligned} \mathbb{C}(t) &= [\xi_0(t) \quad \xi_1(t) \quad \xi_2(t) \quad \dots \quad \xi_{n-1}(t)] \\ \xi_0(t) &= B(t), \quad \xi_j(t) = A(t)\xi_{j-1}(t) - \dot{\xi}_{j-1}(t), \quad j = 1, \dots, n-1. \end{aligned} \quad (3.13)$$

Luego, el sistema (3.12) es controlable si se cumple que  $\text{rank}(\mathbb{C}(t)) = n, \forall t$ . Es importante destacar que este enfoque no puede ser aplicado cuando el grado relativo entre la premisa  $z(t)$  y la entrada  $u(t)$  es menor que  $n$ , ya que en caso contrario la señal de entrada aparece en (3.13), dificultando el cálculo de la controlabilidad por este medio. Sin embargo, es posible analizar la matriz de controlabilidad y limitar el espacio de acciones de control admisibles tal que se cumpla la condición de rango de Kalman. A su vez, cabe destacar que esta formulación se basa en un abuso de notación debido a que se considera que el grado de activación normalizado depende sólo del tiempo, y no considera que las premisas del sistema puedan ser parte del estado o la entrada, con lo cual se obtendría un sistema no lineal y el método propuesto deja de ser válido. Lo mismo ocurre en el planteamiento presentado en [57], donde se presenta el estudio de la controlabilidad utilizando el álgebra de Lie, sin embargo, no tienen en cuenta las variaciones impuestas por los grados de activación, considerando la suma de matrices como un factor independiente del estado, y en consecuencia se derivan condiciones similares a las presentadas en la Sección 3.1 para sistemas lineales.

Debido a lo anterior se plantea que la controlabilidad puede ser estudiada a partir de la teoría general de sistemas no lineales. En [29] abordan el análisis considerando el concepto de estabilidad asintótica, donde se dice que el sistema es controlable si existe una acción de control  $u(t)$  tal que lleva al sistema desde el punto  $x(t_0) = x_0$  a  $x(t_f) = x_f = 0$ . Luego, el teorema de controlabilidad planteado se basa en encontrar una función  $V(x(t), t)$  y una acción de control  $u(t)$  tal que el sistema  $\dot{x}(t) = f(x(t), u(t), t)$  cumpla las siguientes condiciones:

- $\dot{V} = \frac{dV}{dt} + \frac{dV}{dx} f(x, u^*, t) \leq M < \infty, \forall t$  y  $\frac{dV}{dx}, \frac{dV}{dt}$  existen para todo  $x, t$ .
- Para todo vector  $c(t)$  se cumple que  $\lim_{t \rightarrow t_f} c(t) \neq 0 \Rightarrow \lim_{t \rightarrow t_f} V(c(t), t) = \infty$
- La solución de  $\dot{x}(t) = f(x(t), u^*(t), t), x(t_0) = x_0$  existe y es única.

Si bien lo anterior permite un estudio analítico de la controlabilidad, esto no permite determinar a priori condiciones explícitas para asegurarla, por lo que para implementar

un algoritmo de identificación sería necesario programar algún método que permita encontrar funciones candidatas para satisfacer las condiciones anteriores.

Por otro lado, en [13] se aplican conceptos de la teoría de sistemas no lineales afines para estudiar la accesibilidad de una clase de sistemas difusos cuyas premisas se restringen a ser alguno de los estados del sistema solamente. De esta manera, se tiene un sistema difuso de la forma

$$\begin{aligned} \dot{x}(t) &= f(x(t)) + g(x(t))u(t), & x(t) &\in \mathbb{R}^n \\ f(x(t)) &= \sum_{r=1}^{N_r} h_r(x(t))A_r x(t), & g(x(t)) &= \sum_{r=1}^{N_r} h_r(x(t))B_r. \end{aligned} \quad (3.14)$$

Luego, el estudio de la controlabilidad de (3.14) puede ser abordado mediante el análisis de la **matriz de accesibilidad** [36]. En dicho caso, se dice que un sistema no lineal afín con respecto a la entrada es **accesible desde  $x_0$**  si

$$\text{rank}(\mathbb{A}(x_0)) = n, \quad (3.15)$$

$$\mathbb{A}(x_0) = [g(x_0), [f, g](x_0), \dots, ad_f^{n-1}g(x_0)], \quad (3.16)$$

donde  $\mathbb{A}(x_0)$  se denomina matriz de accesibilidad,  $n$  es la dimensión del espacio de estado y  $[f, g](x_0)$  es el producto (o *bracket*) de Lie entre  $f(x(t))$  y  $g(x(t))$  evaluado en  $x_0$ , el cual se define como

$$[f, g](x_0) = \left. \frac{dg(x(t))}{dx} f(x(t)) - \frac{df(x(t))}{dx} g(x(t)) \right|_{x(t)=x_0} = ad_f g(x_0). \quad (3.17)$$

Además se denota  $ad_f^j g(x_0) = [f, ad_f^{j-1}g](x_0)$  y  $ad_f^0 g(x_0) = f(x_0)$ .

También si la matriz de accesibilidad en (3.16) posee rango completo y el sistema no lineal en (3.14) está libre de *drift*, es decir,  $\dot{x}(t) = f(x(t)) + g(x(t))u(t)$ , y  $f(x(t)) = 0, \forall t$ , entonces el sistema no lineal es controlable [28].

A su vez, en [13] plantean la opción de linealizar el sistema difuso en torno al origen y estudiar la controlabilidad aplicando la condición de rango de Kalman. En particular, se dice que un sistema no lineal continuo es **localmente controlable en  $x_0$**  si su linealización en torno a un punto de operación  $(x_0, u_0)$  cumple con la condición de rango de Kalman en (3.4)-(3.5) [42][75].

Emplear técnicas de aproximación de primer orden mediante series de Taylor permite simplificar el estudio de controlabilidad, esto a cambio de limitar la región de validez de las conclusiones obtenidas. Luego, sea un sistema difuso representado como

un sistema no lineal del tipo  $\dot{x}(t) = f(x(t), u(t))$  y  $(x_0, u_0)$  un punto de equilibrio, entonces se puede formular un modelo linealizado de la forma

$$\dot{x}(t) = A(x(t) - x_0) + B(u(t) - u_0) \quad (3.18)$$

$$A = \left. \frac{df(x(t), u(t))}{dx} \right|_{(x_0, u_0)}, B = \left. \frac{df(x(t), u(t))}{du} \right|_{(x_0, u_0)}$$

Por lo tanto, si el modelo en (3.18) es controlable, entonces el sistema original es localmente controlable en  $x_0$ . A su vez, si se asegura que la aproximación lineal de un sistema difuso es controlable para todos los puntos de operación, entonces el modelo difuso no lineal es localmente controlable, acorde a las definiciones presentes al inicio de este capítulo. Adicionalmente, si el modelo linealizado en el origen de un sistema no lineal  $\dot{x}(t) = f(x(t), u(t))$  es controlable, y además el sistema no forzado  $\dot{x}(t) = f(x(t), 0)$  es globalmente asintóticamente estable, entonces se dice que el sistema original es completamente controlable [85].

Cabe recordar que las definiciones anteriores son aplicables para sistemas de tiempo continuo, sin embargo, también se han reportado resultados para sistemas no lineales de tiempo discreto, los cuales podrían ser extrapolados al caso difuso. En particular, en [50] se plantea la linealización y evaluación de controlabilidad, demostrando resultados similares al caso continuo. Por otro lado, la definición del álgebra de Lie es diferente para sistemas de tiempo discreto, más engorrosa, y en consecuencia no es directo aplicar las definiciones anteriores. En [41][104] se define el álgebra de Lie para sistemas de tiempo discreto, donde se obtienen teoremas similares a las condiciones en (3.15)-(3.16) para corroborar la controlabilidad de este tipo de sistemas.

Si bien se han expuesto alternativas para analizar la controlabilidad tanto de sistemas lineales como no lineales, puede que el modelo sea no controlable y en consecuencia afecte el diseño de una estrategia de control. Para abordar la problemática de modelos dinámicos que presenten deficiencias en cuanto a controlabilidad, en la literatura se propone el uso de aproximaciones de orden superior de los procesos de tal forma que permitan ampliar el rango en que el modelo es controlable, en comparación a utilizar aproximaciones de primer orden.

### 3.3. Controlabilidad de Aproximaciones de Orden Superior

En particular, Patwardhan & Madhavan [79] proponen el uso de aproximaciones de segundo orden obtenidas a partir del método de perturbaciones regulares, y su metodología se fundamenta en el Lema de Morse, el cual caracteriza el comportamiento de las aproximaciones de segundo orden en puntos de operación donde los términos de primer orden se anulan (puntos críticos), quedando los términos de segundo orden

como respaldo para poder controlar el sistema. De esta manera se demuestra mediante simulaciones que el rango de controlabilidad del sistema es ampliado, permitiendo alcanzar los objetivos de control planteados. Cabe destacar que esta propuesta requiere el cálculo de gradientes y Hessianos en cada iteración, por lo que eventualmente el esfuerzo computacional puede ser bastante alto.

Siguiendo la línea de estudio de Patwardhan & Madhavan, en [80] plantean una solución al problema de la eficiencia computacional para el cálculo de los coeficientes de la aproximación de segundo orden. Para ello proponen el uso de ecuaciones de sensibilidad de primer y segundo orden que rijan el comportamiento de los parámetros del modelo y que permitan el desarrollo de un modelo de perturbación cuadrático recursivo.

En [81], Patwardhan & Madhavan extienden los resultados obtenidos en los dos trabajos anteriores a sistemas que posean un mapeo no lineal en la salida. En este trabajo se redefine el modelo de perturbación de segundo orden con el fin de retratar mejor los efectos de los términos cuadráticos. Se realiza una comparación entre las aproximaciones de segundo orden obtenidas mediante el método de perturbaciones regulares utilizado en [79], y el modelo obtenido a través de ecuaciones de sensibilidad de primer y segundo orden utilizado en [80], demostrando que este último aproxima mejor la dinámica del sistema debido a que no necesita los términos de primer orden para obtener la aproximación de segundo orden.

Cabe destacar que las aproximaciones de mayor orden derivan en expresiones multidimensionales llamadas tensores, por lo que dichos términos quedan sujetos al álgebra definida para operar estos elementos. En particular, Patwardhan & Madhavan centran sus procedimientos en las definiciones realizadas por Economou [24], la cual también ha sido abordada por Gomme & Klein [30], en donde caracterizan los términos de segundo orden mediante un sistema de ecuaciones lineales derivado de la definición de matriz Hessiana presente en [60], concatenando las derivadas de segundo orden formando una matriz rectangular y realizando aproximaciones de segundo orden basadas en el producto de Kronecker, el cual se define como:

$$A_{m \times n} \otimes B_{p \times q} = \begin{bmatrix} a_{11}B & \dots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \dots & a_{mn}B \end{bmatrix}_{mp \times nq} \quad (3.19)$$

Luego, a partir de esta definición resulta más fácil la programación de las aproximaciones de segundo orden debido a que se remonta a operaciones matriciales.

Una vez que se tiene una representación matemática del proceso es posible diseñar diversas estrategias de control para llevar al sistema a puntos de referencia deseados. A lo largo del tiempo se han desarrollado varias metodologías, como el clásico control PID, o bien estrategias más avanzadas como control óptimo, control inteligente o control adaptable. En particular, se destaca el uso del control predictivo basado en modelos, una estrategia de control óptimo que utiliza un modelo de predicción para un horizonte

dado, y a partir de dicha predicción y la minimización de un funcional de costos se determina la acción de control a tomar en el tiempo actual [17]. En la Sección 8.2 de anexos se enuncian las principales referencias del uso de esta estrategia en la literatura especializada, como también se detalla su formulación y la manera de sintonizar este tipo de controlador.

Teniendo en cuenta la revisión bibliográfica realizada, es posible tomar las herramientas presentes en la literatura para proponer nuevos métodos de identificación difusa que permitan obtener buenos resultados al momento de plantear estrategias de control. En los capítulos siguientes se describe en detalle las propuestas diseñadas.



## 4. Identificación Difusa con Restricciones de Controlabilidad

En este capítulo se describe la propuesta de identificación de sistemas difusos teniendo en cuenta el procedimiento detallado en el Capítulo 2 y los métodos para estudiar la controlabilidad expuestos en el Capítulo 3. En particular, se plantea el algoritmo que se describe a continuación para obtener modelos difusos que posean la propiedad de ser controlables en el rango de operación en que trabajan, imponiendo restricciones a partir del estudio de la controlabilidad de la aproximación de primer orden del sistema difuso.

### 4.1. Algoritmo de Identificación

La propuesta para el algoritmo de identificación se basa en el algoritmo GK presentado en la Sección 2.5.1, y por ello se plantea el uso de funciones de pertenencia del tipo gaussianas, aprovechando que el algoritmo encuentra agrupaciones hiperelipsoidales, por lo que resulta simple ajustar este tipo de funciones. Luego, el grado de pertenencia de la  $i$ -ésima premisa al  $r$ -ésimo *cluster* está dado por

$$\mu_r(z_i(k)) = \exp\left(-0.5 \cdot \left(a_{r,i} \cdot (z_i(k) - b_{r,i})\right)^2\right), \quad (4.1)$$

donde  $a_{r,i}$  es el inverso de la desviación estándar de los datos ajustados por la gaussiana, y  $b_{r,i}$  representa su media.

A su vez, se consideran modelos locales lineales autorregresivos con variable exógena, por lo que el vector de premisas es de la forma  $z(k) = [y(k-1), \dots, y(k-n_y), u(k-1), \dots, u(k-n_u)]^T$ , donde  $n_y$  y  $n_u$  representan el número de retardos a considerar de la salida y la entrada respectivamente, además se impone  $n_y \geq n_u$  para asegurar que el orden del sistema esté dado por  $n_y$ . Luego, el  $r$ -ésimo modelo local queda representado de la forma

$$f_r(z(k)) = \theta_r^T z(k) = \theta_{r,1}y(k-1) + \dots + \theta_{r,n_y}y(k-n_y) + \theta_{r,n_y+1}u(k-1) + \dots + \theta_{r,n_y+n_u}u(k-n_u). \quad (4.2)$$

Una vez definidas las premisas, el tipo de funciones de pertenencia y los modelos locales del sistema difuso, se procede con la metodología de identificación expuesta en detalle en el Capítulo 2. Luego del diseño del experimento de identificación, se propone utilizar una división del 60% de los datos para entrenamiento, 20% para validación y 20% para prueba, teniendo en cuenta la causalidad temporal de los datos, esto para utilizar un término medio entre los valores consultados en la literatura especializada.

La selección de variables de interés en este caso se limita a escoger el número de retardos  $n_y$  y  $n_u$ . Se propone identificar un número máximo de retardos de la salida,  $n_{y,max}$ , mediante la función de autocorrelación parcial de  $y(k)$ , y luego variar  $n_y$  en el rango  $[1, n_{y,max}]$ , y  $n_u$  en el rango  $[1, n_y]$ , de tal forma que el orden del sistema este determinado por la cantidad de retardos de la salida,  $n_y$ , y de esta manera poder corroborar la controlabilidad del sistema mediante el método de la matriz de Sylvester. Posteriormente se compara el desempeño de los distintos modelos planteados.

De manera similar, para escoger la cantidad óptima de reglas del modelo se propone variar el número de *clusters* y comparar el desempeño de los modelos obtenidos [10]. Para ello se debe definir a priori un número máximo de *clusters* a evaluar, el cual depende de la aplicación en sí. Si se tienen datos de un modelo no lineal con varios puntos de operación, lo más probable es que se necesite una mayor cantidad de reglas para representar de buena manera la dinámica del sistema. Si en una primera instancia se obtiene un modelo óptimo con un número de reglas igual al máximo indicado, y el modelo obtenido no tiene un buen desempeño acorde a su propósito, se aconseja aumentar el valor del número máximo de reglas y volver a realizar el procedimiento de identificación.

Las condiciones de controlabilidad expuestas en el Capítulo 3 indican que esta propiedad depende principalmente de los parámetros del sistema. En este caso se realiza una aproximación de primer orden del modelo difuso utilizando series de Taylor para encontrar condiciones que permitan inferir restricciones de controlabilidad. La idea del algoritmo se basa en realizar un análisis simplificado de la controlabilidad del sistema, donde, si bien no se garantiza la controlabilidad global teóricamente, se analiza la controlabilidad local en todos los puntos de operación presentes en los datos de identificación, de tal manera de imponer una controlabilidad local extendida, la cual permita obtener un modelo controlable en los puntos de interés. A su vez, para simplificar el análisis de la controlabilidad local del sistema, se asume que la aproximación de primer orden en un punto de operación tiene sus parámetros constantes, para luego repetir el mismo análisis al actualizar sus valores en otro punto de operación.

Luego, dada la clase de sistema difuso con la que se trabaja, se tiene que

$$y_{fuzzy}(k) = \sum_{r=1}^{N_r} h_r(z(k)) \cdot \theta_r^T z(k), \quad (4.3)$$

$$w_r(z(k)) = \prod_{i=1}^{n_y+n_u} \exp\left(-0.5 \cdot (a_{r,i} \cdot (z_i(k) - b_{r,i}))^2\right), \quad (4.4)$$

donde  $h_r(z(k))$  se define como en (2.3). Si se deriva (4.3) con respecto a la  $i$ -ésima premisa,  $z_i(k)$ , se obtiene

$$\frac{dy_{fuzzy}(k)}{dz_i(k)} = \sum_{r=1}^{N_r} \left\{ \frac{dh_r(z(k))}{dz_i(k)} \theta_r^T z(k) + h_r(z(k)) \cdot \theta_{r,i} \right\}, \quad (4.5)$$

$$\begin{aligned} \frac{dh_r(z(k))}{dz_i(k)} = & -(a_{r,i})^2 (z_i(k) - b_{r,i}) \cdot h_r(z(k)) \\ & - h_r(z(k)) \sum_{l=1}^{N_r} -(a_{l,i})^2 (z_i(k) - b_{l,i}) \cdot h_l(z(k)). \end{aligned} \quad (4.6)$$

Luego, se formula el modelo linealizado evaluado en un punto de operación  $z_0 = [z_{0,1}, \dots, z_{0,n_y+n_u}]$  como

$$y(k) = \sum_{i=1}^{n_y+n_u} \{ \hat{\theta}_i(\theta, a_{\cdot,i}, b_{\cdot,i}, z_0) \cdot (z_i(k) - z_{0,i}) \} + y(z_0), \quad (4.7)$$

donde  $\hat{\theta}_i(\theta, a_{\cdot,i}, b_{\cdot,i}, z_0) = \left. \frac{dy_{fuzzy}(k)}{dz_i(k)} \right|_{z(k)=z_0}$ .  $\theta$  son los parámetros de las consecuencias,  $a_{\cdot,i} = [a_{1,i}, \dots, a_{N_r,i}]$  y  $b_{\cdot,i} = [b_{1,i}, \dots, b_{N_r,i}]$  son los parámetros de la  $i$ -ésima premisa para cada regla. Para simplificar la notación se considera  $\hat{\theta}_i = \hat{\theta}_i(\theta, a_{\cdot,i}, b_{\cdot,i}, z_0)$ .

El modelo en (4.7) puede ser reescrito como

$$y(k) = \sum_{i=1}^{n_y+n_u} \{ \hat{\theta}_i \cdot z_i(k) \} + \zeta(z_0), \quad (4.8)$$

donde  $\zeta(z_0) = y(z_0) - \sum_{i=1}^{n_y+n_u} \{ \hat{\theta}_i \cdot z_{0,i} \}$ . Luego, (4.8) puede ser interpretado como un modelo lineal con una perturbación constante  $\zeta(z_0)$  [76], el cual puede ser reescrito en variables de estado de manera general de la forma

$$x(k+1) = Ax(k) + Bu(k) + w \quad (4.9)$$

$$y(k) = Cx(k) + Du(k) + v \quad (4.10)$$

con  $A = A(z_0)$ ,  $B = B(z_0)$ ,  $C = C(z_0)$ ,  $D = D(z_0)$ ,  $w = w(z_0)$  y  $v = v(z_0)$  las constantes del sistema dadas por el punto de operación  $z_0$ . La solución de la ecuación de diferencias (4.9) está dada por

$$x(k) = A^k x(0) + \sum_{i=0}^{k-1} A^{k-1-i} B u(i) + \sum_{i=0}^{k-1} A^{k-1-i} w, \quad (4.11)$$

de donde se tiene que la perturbación constante genera un desplazamiento al añadir un término extra, afectando la dinámica del sistema, mas no afecta en la relación entre la entrada  $u(k)$  y el estado  $x(k)$ , debido a que persiste la combinación lineal entre los vectores  $\{A^{k-1-i} B, i = 0, 1, \dots, n\}$  y las acciones de control  $u(i)$ , conservando la condición de controlabilidad. Luego, si dichos vectores son linealmente independientes, entonces el sistema linealizado es controlable. Por lo tanto, el estudio de controlabilidad de un sistema lineal con perturbación constante como en (4.8) puede ser abordado sin tomar en cuenta dicha perturbación.

En particular, si  $z_0 = z(k-1)$  y se corrobora la controlabilidad del sistema linealizado para todos los puntos en el conjunto de entrenamiento (asumiendo que dicho conjunto posee información de todo el espacio de estado, o en su defecto de los puntos relevantes en que opera el sistema), entonces el sistema difuso es localmente controlable, acorde a la definición expuesta en la Sección 3.2. Cabe recordar que en este contexto se habla de una controlabilidad local extendida, ya que teóricamente no se comprueba que el sistema se completamente controlable.

El sistema linealizado podría ser transformado directamente desde la ecuación de entrada-salida a la ecuación en variables de estado utilizando la forma canónica controlable (ver Sección 8.4.1 de anexos) o la forma canónica observable (ver Sección 8.4.2 de anexos). Sin embargo, utilizando una u otra representación se debe asumir que el sistema es controlable u observable (según corresponda), restando generalidad al método propuesto. Es por esto que para el estudio de controlabilidad se aplica el método de la matriz de Sylvester utilizando los parámetros del sistema linealizado (4.8), ya que para ello no se requiere considerar alguno de los supuestos anteriores. Cabe recordar que las premisas del modelo son regresores de la salida y entrada del sistema, por lo que el sistema en (4.8) puede ser expresado en forma de función de transferencia, permitiendo utilizar la matriz de Sylvester para evaluar la controlabilidad del sistema aproximado. Por otro lado, debido a que el estudio de controlabilidad mediante la matriz de Sylvester requiere conocer si dicha matriz es singular o no, se plantea reemplazar la condición del determinante por el cálculo del *condition number* de la matriz, debido a que este último es una mejor medida de cercanía a singularidad que el determinante, debido a sus propiedades [70].

A partir de (4.5) y (4.6) se puede observar que la controlabilidad depende tanto de los parámetros de las consecuencias como de los parámetros de las premisas, por lo que se debe modificar la metodología expuesta en la Sección 2.5 con el fin de incluir restricciones que permitan obtener un modelo controlable. En el caso de los parámetros de las premisas, se elige no alterar esta etapa de la identificación debido a que variar los parámetros de las funciones de pertenencia puede generar una base de reglas que no necesariamente se ajusten al rango de operación del sistema en estudio, alterando

completamente el modelo, por lo que se opta modificar la etapa de identificación de los parámetros de las consecuencias solamente.

Para incluir condiciones de controlabilidad se replantea el problema de optimización descrito en la Sección 2.5.2, utilizando un método de mínimos cuadrados con restricciones de la forma [83]

$$\min_{\theta} J(\theta) = \sum_{k=1}^N \left( y(k) - y_{fuzzy}(k, \theta) \right)^2 + \sum_{k=1}^N \varphi \left( S_{fuzzy}(k, \theta) \right) \quad (4.12)$$

$$\varphi \left( S_{fuzzy}(k, \theta) \right) = \begin{cases} \lambda & \text{si } S_{fuzzy}(k, \theta) \text{ es singular} \\ 0 & \text{si } S_{fuzzy}(k, \theta) \text{ es no singular,} \end{cases} \quad (4.13)$$

donde  $\lambda \gg \sum_{k=1}^N \left( y(k) - y_{fuzzy}(k, \theta) \right)^2$  penaliza el caso en que el sistema es no controlable;  $N$  es el total de datos del conjunto de entrenamiento;  $y(k)$  es la salida medida del proceso;  $y_{fuzzy}(k, \theta)$  es la salida del modelo difuso al considerar los parámetros de las consecuencias  $\theta$ ; y  $S_{fuzzy}(z(k), \theta)$  es la matriz de Sylvester del modelo linealizado en el instante  $k$  dado los parámetros  $\theta$ . Como la idea detrás del planteamiento radica en asegurar la controlabilidad para cada dato del conjunto de entrenamiento, es necesario linealizar el sistema en cada punto de operación  $z_0(k) = z(k-1)$ , calcular su matriz de Sylvester y verificar si ésta es no singular, de tal manera de asegurar la controlabilidad local extendida del modelo difuso no lineal.

Notar que el problema de optimización (4.12)-(4.13) está planteado para un procedimiento de identificación a un paso. En caso de querer identificar un sistema a más pasos (identificación a  $N$  pasos, según la nomenclatura adoptada en la Sección 2.5.2) se debe resolver

$$\min_{\theta} J(\theta) = \sum_{k=1}^N \left( y(k_0 + k) - y_{fuzzy}(k_0 + k | k_0, \theta) \right)^2 + \sum_{k=1}^N \varphi \left( S_{fuzzy}(k_0 + k | k_0, \theta) \right), \quad (4.14)$$

donde las premisas se actualizan utilizando las predicciones del modelo difuso, es decir, en este caso se tiene  $z(k_0 + k) = [y_{fuzzy}(k_0 + k - 1 | k_0), \dots, y_{fuzzy}(k_0 + k - n_y | k_0), u(k_0 + k - 1), \dots, u(k_0 + k - n_u)]^T$ .

A diferencia del caso expuesto en la Sección 2.5.2 donde se tienen modelos locales lineales en los parámetros y una función de costos convexa sin restricciones, en esta oportunidad no se tiene una solución analítica del problema de optimización en (4.12)/(4.14), debido a la discontinuidad de la función objetivo. Luego, es necesario

emplear algoritmos de optimización para encontrar una solución aproximada. Para ello se consideran tres métodos de optimización con distintas características: uno basado en gradiente (Quasi-Newton), uno que no requiere el gradiente de la función (Nelder-Mead), y un algoritmo evolutivo (PSO); esto con el fin de comparar sus rendimientos. En la Sección 8.1 de anexos se detallan estos algoritmos.

Cabe destacar que los tres algoritmos de optimización a utilizar requieren de una condición inicial para iniciar a iterar. Luego, para identificar los parámetros de las consecuencias del modelo se plantean dos alternativas: inicializar los parámetros aleatoriamente, o bien utilizar los parámetros obtenidos a partir de la solución del problema de identificación de los parámetros de las consecuencias sin imponer restricciones de controlabilidad.

Finalmente, el algoritmo de identificación de modelos difusos con restricciones de controlabilidad se puede resumir en el siguiente pseudo-código:

- 1) Obtener datos de entrada-salida e identificar el número de retardos máximo de la salida a considerar,  $n_{y,max}$ .
- 2) Definir parámetros como el número máximo de *clusters* o reglas ( $reglas_{max}$ ),  $\lambda$ , el tipo de algoritmo de optimización a utilizar y sus condiciones iniciales; el tipo de identificación: a un paso o a  $N$  pasos.
- 3) Dividir los datos de entrada-salida en conjuntos de entrenamiento (60%), validación (20%) y prueba (20%).
- 4) Para  $n_y = 1$  a  $n_{y,max}$ 
  - a. Para  $n_u = 1$  a  $n_y$ 
    - i. Para  $\#reglas = 1$  a  $reglas_{max}$ 
      1. Entrenar modelo difuso utilizando los datos de entrenamiento, **considerando las restricciones de controlabilidad.**
      2. Calcular RMSE de entrenamiento y validación utilizando los datos del conjunto correspondiente.
      3. Comparar RMSE de validación y actualizar modelo óptimo.
    - ii. Siguiendo  $\#reglas$
  - b. Siguiendo  $n_u$
- 5) Siguiendo  $n_y$
- 6) El modelo óptimo es el que posee menor RMSE de validación entre todos los modelos generados por las combinaciones de  $n_y, n_u$  y  $\#reglas$ .

Luego de obtener un modelo difuso controlable es posible utilizarlo para el diseño de controladores. En particular, en este trabajo se abordan estrategias de control predictivo difuso, donde se emplea el modelo obtenido del proceso de identificación anterior como modelo de predicción. En la siguiente sección, la identificación difusa propuesta y la

estrategia de control predictivo difuso diseñada es aplicada a distintos casos para estudiar su desempeño.

## 4.2. Casos de Estudio

Para validar la estrategia propuesta en las secciones anteriores se plantean dos casos de estudio: la aplicación del algoritmo al modelo del reactor de estanque continuamente agitado presentado en el Capítulo 1 como motivación, y el uso de un sistema no lineal teóricamente controlable.

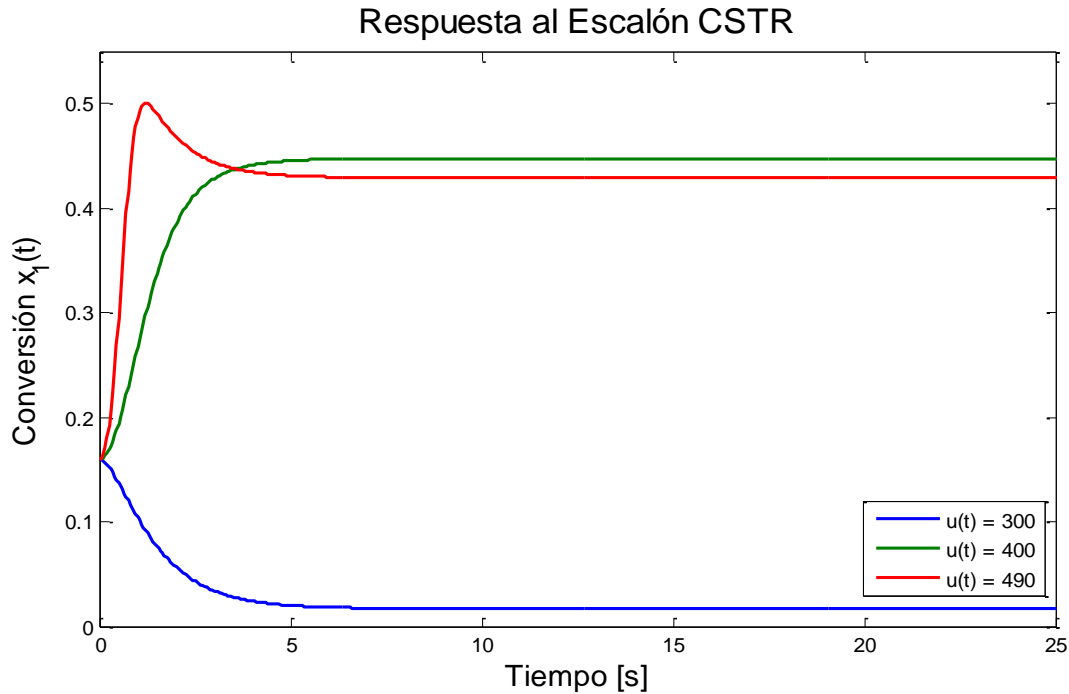
### 4.2.1. Reactor de Estanque Continuosamente Agitado

#### 4.2.1.1. Identificación del CSTR

El primer paso para identificar al CSTR es obtener datos de entrada-salida que representen su dinámica en distintos puntos de operación. Por lo tanto, se debe diseñar la señal de excitación que permita obtener dicha información del proceso. Para ello se estudia la respuesta al escalón de la planta para determinar las frecuencias de interés.

En la Figura 4.1 se puede observar la respuesta del CSTR para distintos escalones. De la gráfica se puede apreciar que el sistema comienza a estabilizarse alrededor de los 5 [s], mientras que superado los 10 [s] la planta está completamente en estado permanente. Luego, la máxima frecuencia de interés tiene un valor de  $f_{max} = 1/5$  [Hz], esto para lograr capturar información en estado transiente de la planta. En cuanto a la frecuencia mínima, cualquier valor menor a  $f_{min} = 1/10$  [Hz] permite capturar información en estado permanente. En este caso se considera utilizar un quinto de la frecuencia máxima, es decir,  $f_{min} = f_{max}/5$  [Hz]. Por otra parte, para el tiempo de muestreo se considera  $T_s = 0.05$  [s], emulando el muestreo utilizado en algunos procesos reales.

Con los parámetros anteriores, y utilizando las ecuaciones expuestas en la Sección 2.2, se genera una PRBS de orden  $n = 4$ , por lo que el largo máximo corresponde a  $N = 2^n - 1 = 15$ . A su vez, la cantidad de muestras por bit son  $N_s = 40$ . Luego,  $\Delta t = N_s \cdot T_s = 2$  [s], por lo que la PRBS dura en total 30 [s]. Debido a que se considera poco tiempo (y en consecuencia pocos datos) para una prueba de identificación de un sistema no lineal, se opta por generar otras PRBS con los mismos parámetros, pero con distintas condiciones iniciales, concatenándolas para obtener una señal de mayor duración. Utilizar distintas condiciones iniciales genera un desplazamiento de la señal, por lo que no necesariamente se repiten los datos.



**Figura 4.1: Respuesta al Escalón del CSTR.**

Finalmente se genera la APRBS variando la amplitud aleatoriamente de la PRBS generada. Para ello se cuantifica la cantidad de valores posibles para obtener diferencias en amplitud relevantes, y también se evita que la amplitud entre un cambio y otro sea la misma, para evitar modificar el comportamiento en frecuencia de la señal.

Cabe destacar que se realizan pruebas de identificación en lazo abierto y en lazo cerrado, por lo que se traslada una APRBS a valores en el rango [300,490] en el caso de lazo abierto, mientras que en lazo cerrado la APRBS toma valores entre 0.02 y 0.5088, donde dichos valores se obtienen a partir del rango de operación de las variables manipulada y controlada, respectivamente. En el caso de la identificación en lazo cerrado se emplea un controlador predictivo para la obtención de los datos, utilizando como modelo de predicción el modelo fenomenológico del CSTR dado en (1.1)-(1.3), el cual es discretizado utilizando la aproximación de Euler

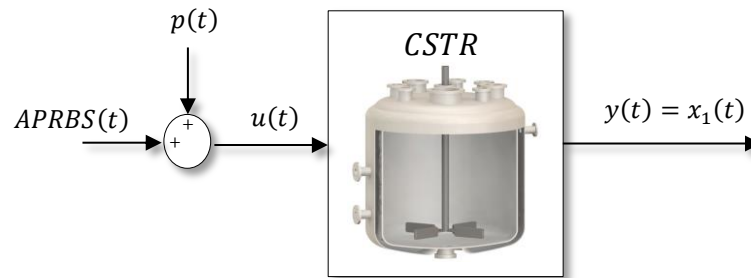
$$\frac{dx(t)}{dt} = \frac{x(k+1) - x(k)}{h}, \quad (4.15)$$

donde se considera  $h = 0.3$  para sintonizar el controlador acorde al desempeño deseado.

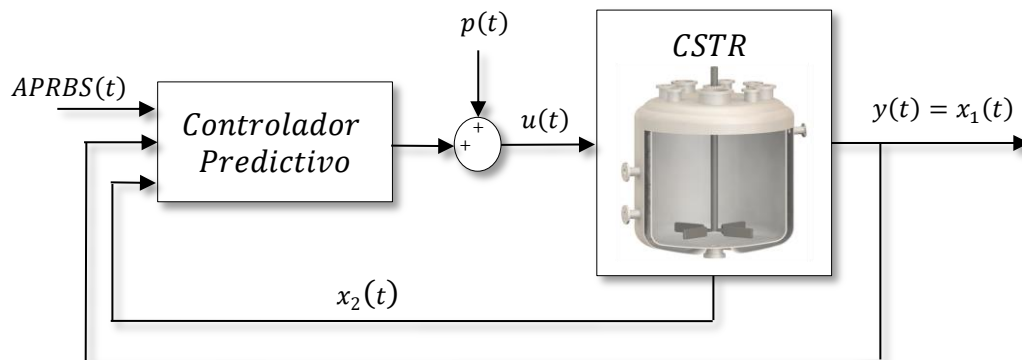
En ambas pruebas se considera una componente aleatoria  $p(t)$  uniformemente distribuida en el rango  $[-2,2]$ . Se escoge ese rango para obtener una variación aproximadamente del 1% respecto a la diferencia entre el mínimo y máximo de  $u(t)$ .



El diagrama de bloques utilizado para la prueba de identificación en lazo abierto se presenta en la Figura 4.2, mientras que para la prueba de identificación en lazo cerrado se utiliza el esquema de la Figura 4.3.

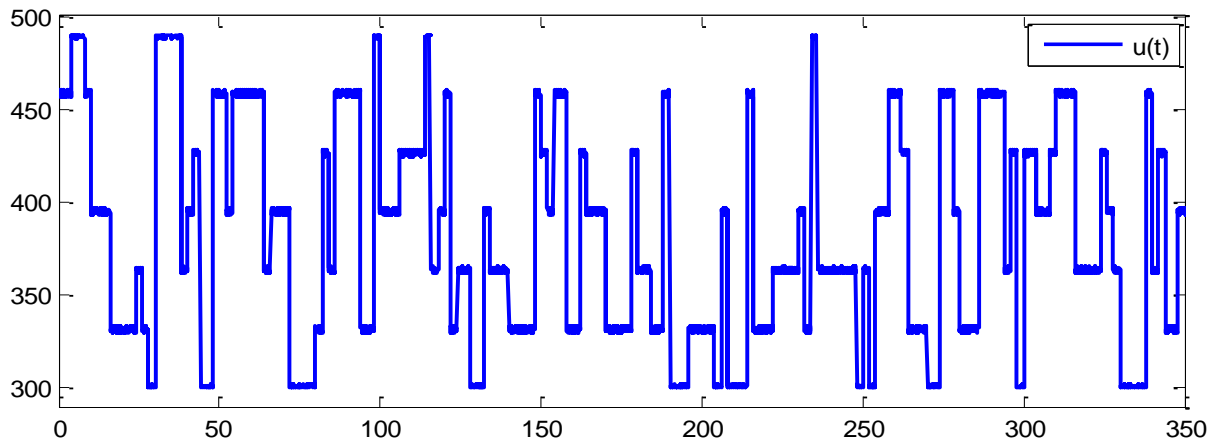


**Figura 4.2: Diagrama de Bloques Identificación CSTR en Lazo Abierto.**

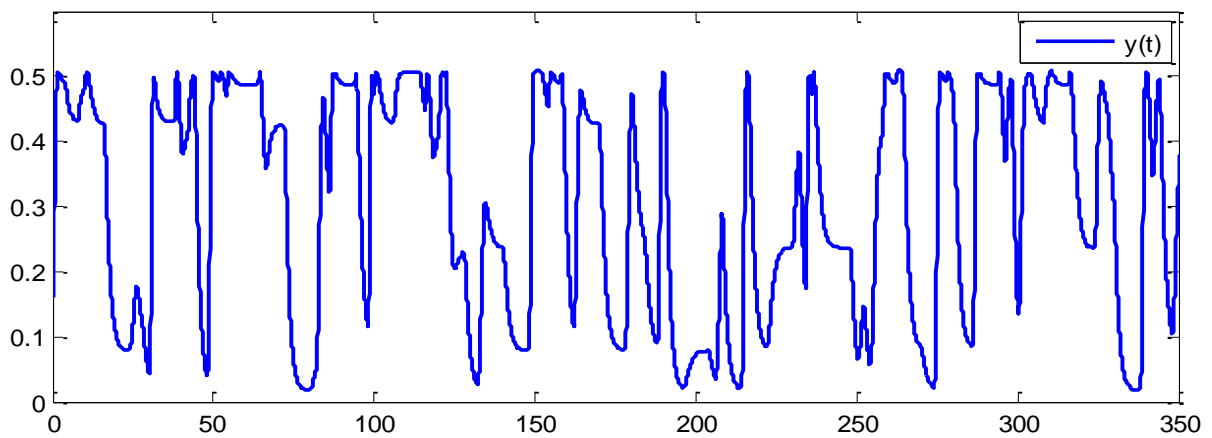


**Figura 4.3: Diagrama de Bloques Identificación CSTR en Lazo Cerrado.**

Parte de los datos recolectados en lazo abierto se pueden observar en la Figura 4.4, mientras que parte de los datos en lazo cerrado se aprecian en la Figura 4.5.

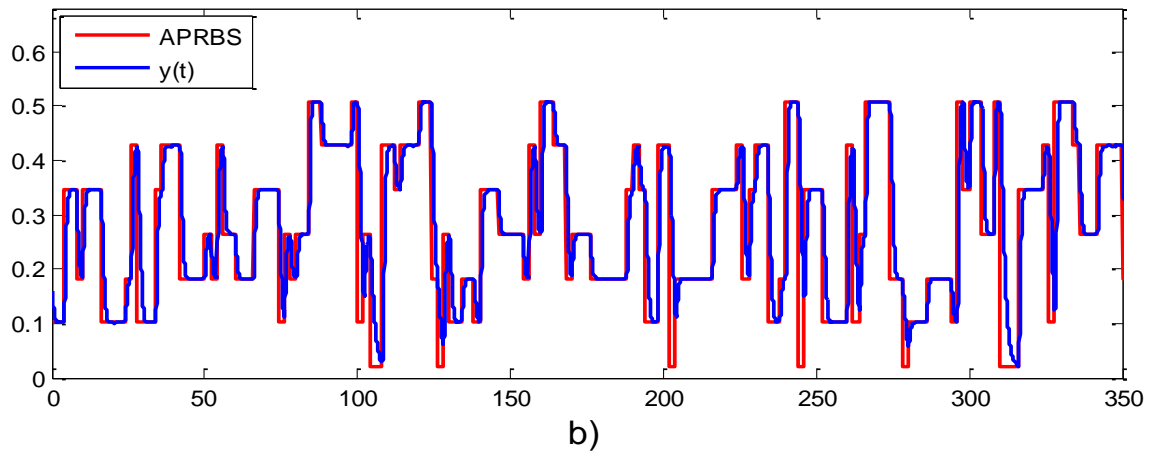
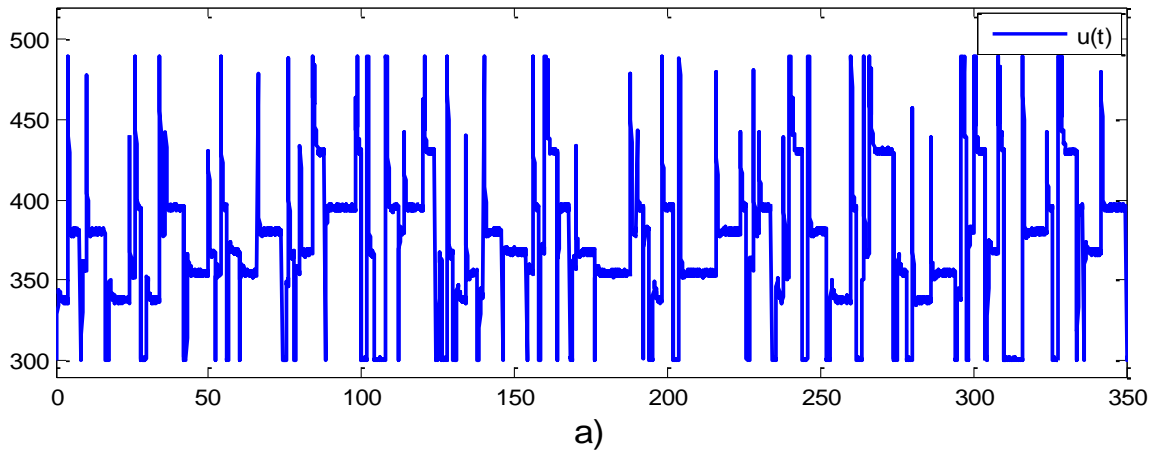


a)



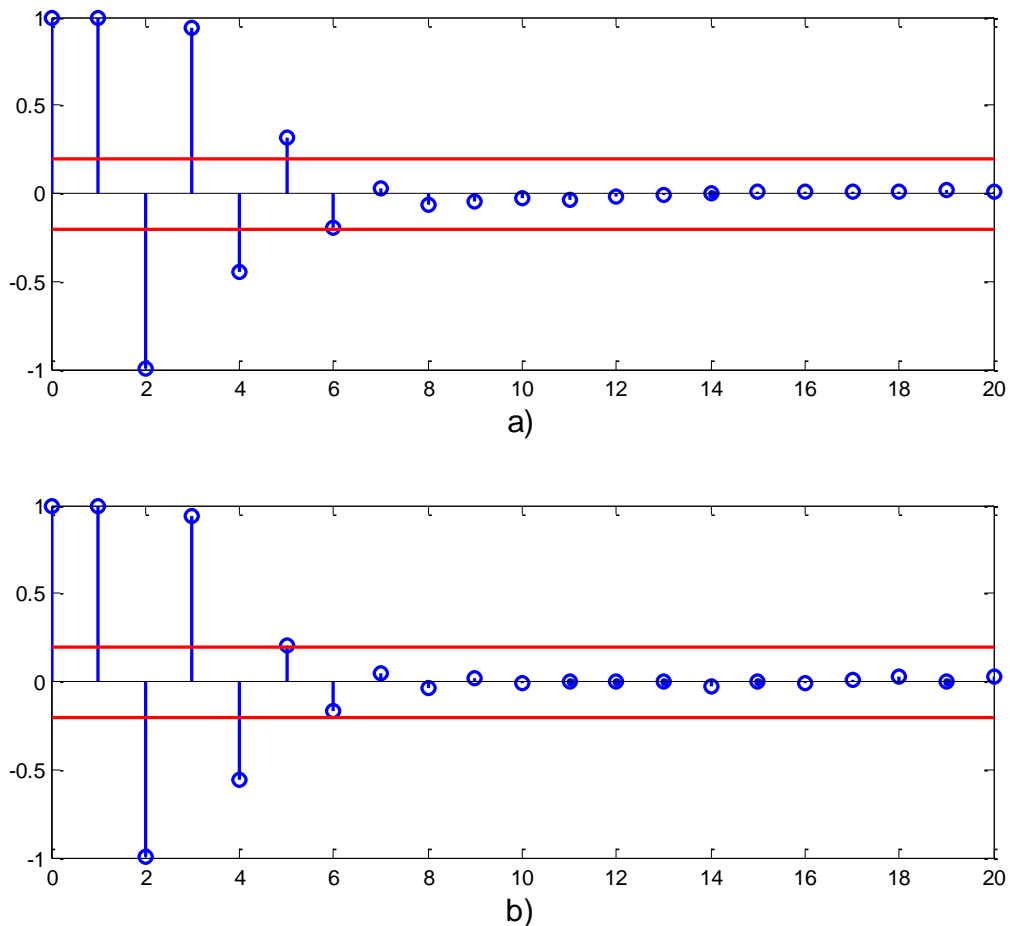
b)

**Figura 4.4: Datos de Identificación del CSTR en Lazo Abierto. a) Entrada  $u(t)$  [ $^{\circ}$  K] (APRBS) vs Tiempo [s]. b) Datos de Salida  $y(t)$  vs Tiempo [s].**



**Figura 4.5: Datos de Identificación del CSTR en Lazo Cerrado. a) Entrada  $u(t)$  [ $^{\circ}$  K] vs Tiempo [s]. b) APRBS (Referencia) y Datos de Salida  $y(t)$  vs Tiempo [s].**

Una vez obtenidos los datos se procede a determinar el retardo máximo de la salida,  $n_{y,max}$  mediante el estudio de su función de autocorrelación parcial. En particular, para determinar  $n_{y,max}$  se consideran sólo los retardos que tengan un índice de correlación fuera del rango  $[-0.2,0.2]$ , debido a que bajo ese umbral los retardos tienen poca relación con  $y(k)$ , por lo que ya no aportan información relevante. En la Figura 4.6 se observan las funciones de autocorrelación parcial de la salida  $y(k)$  para los datos de identificación en lazo abierto y en lazo cerrado. Se puede apreciar que ambas gráficas presentan un comportamiento similar, donde se puede identificar un retardo máximo a considerar de  $n_{y,max} = 5$ .



**Figura 4.6: Gráfico Función de Autocorrelación Parcial CSTR vs Número de Retardos. a) Datos de Identificación en Lazo Abierto. b) Datos de Identificación en Lazo Cerrado.**

Una vez obtenidos los datos para la identificación del modelo difuso y determinado el número de retardos máximos a considerar,  $n_{y,max}$ , se procede a realizar las pruebas de identificación dadas por el algoritmo presentado en la Sección 4.1, donde se considera un valor  $\lambda = 10^{10}$  para penalizar a los modelos que sean no controlables, y a priori un número máximo de *clusters* o reglas de  $reglas_{max} = 5$ .

Cabe recordar que se realizan pruebas utilizando distintas configuraciones, donde se consideran tres aspectos:

- **Tipo de algoritmo de optimización (Solver):** Quasi-Newton (QN), Nelder-Mead (NM), y PSO.
- **Condiciones Iniciales (CI's):** Aleatorias, y dadas por la solución del problema de identificación de parámetros de las consecuencias sin considerar restricciones de controlabilidad (CI's conocidas).
- **Tipo de Identificación:** Identificación a uno y  $N$  pasos.

A continuación se detallan los resultados obtenidos en las pruebas de identificación en lazo abierto y en lazo cerrado.

- **IDENTIFICACIÓN EN LAZO ABIERTO**

Los resultados obtenidos al utilizar los datos en lazo abierto se detallan en la Tabla 4.2 y en la Tabla 4.3, correspondientes a la identificación a uno y  $N$  pasos respectivamente, en donde se pueden observar los índices RMSE de la predicción a uno y  $N$  pasos del modelo obtenido.  $[n_y^*, n_u^*, reglas^*]$  indica el número óptimo de retardos en la salida, el número óptimo de retardos en la entrada y el número óptimo de reglas respectivamente, representando la estructura del modelo difuso obtenido del proceso de identificación. Cabe recordar que la elección del modelo óptimo recae en el RMSE obtenido en el conjunto de validación utilizando la función objetivo a uno o  $N$  pasos, según corresponda.

En el caso de utilizar PSO como método de optimización se incorpora una pequeña modificación al algoritmo presentado en la Sección 8.1.3, ya que se altera la condición inicial de una de las partículas del enjambre para reemplazarla por la solución óptima de la identificación de parámetros de las consecuencias sin utilizar restricciones de controlabilidad. A su vez, cabe destacar que los parámetros de PSO deben ser sintonizados dependiendo de la aplicación, principalmente los límites de las soluciones, el tamaño de la población y el número de iteraciones. En este caso se consideran los parámetros definidos en la Tabla 4.1 (ver Sección 8.1.3 de anexos para revisar la notación utilizada).

**Tabla 4.1: Parámetros PSO para Identificación del CSTR.**

<b>Parámetro</b>	<b>Valor</b>
Tamaño de la población	15
Número de iteraciones	30
$[p_{min}, p_{max}]$	$[-5,5]$
$V_{max}$	1
$[c_1, c_2]$	$[2,2]$
$[w_{inicial}, w_{final}]$	$[0.95,0.4]$
$[f_c, s_c]$	$[5,15]$

**Tabla 4.2: Resultados Identificación del CSTR en Lazo Abierto a un Paso.**

<i>Solver</i>	<i>CI's</i>	RMSE Entrenamiento		RMSE Validación		RMSE Prueba		$[n_y^*, n_u^*, reglas^*]$
		1 Paso	<i>N</i> Pasos	1 Paso	<i>N</i> Pasos	1 Paso	<i>N</i> Pasos	
QN	Aleatorias	4.3450 e-04	0.1328	3.4497 e-04	0.1190	4.4034 e-04	0.1236	[2,1,1]
	Conocidas	7.2159 e-05	0.0400	<b>6.0736 e-05</b>	0.0410	9.0838 e-05	0.0416	[5,4,4]
NM	Aleatorias	0.0012	673.6326	9.5659 e-04	860.9673	0.0014	717.5477	[5,1,1]
	Conocidas	7.2159 e-05	0.0400	<b>6.0736 e-05</b>	0.0410	9.0838 e-05	0.0416	[5,4,4]
PSO	Conocidas	7.2159 e-05	0.0400	<b>6.0736 e-05</b>	0.0410	9.0838 e-05	0.0416	[5,4,4]

**Tabla 4.3: Resultados Identificación del CSTR en Lazo Abierto a *N* Pasos.**

<i>Solver</i>	<i>CI's</i>	RMSE Entrenamiento		RMSE Validación		RMSE Prueba		$[n_y^*, n_u^*, reglas^*]$
		1 Paso	<i>N</i> Pasos	1 Paso	<i>N</i> Pasos	1 Paso	<i>N</i> Pasos	
QN	Aleatorias	0.0362	0.1300	0.0327	0.1167	0.0341	0.1220	[3,3,1]
	Conocidas	7.7876 e-05	0.0308	6.5168 e-05	0.0222	9.6099 e-05	0.0290	[5,4,3]
NM	Aleatorias	39.2556	2.4020	49.5700	2.8441	42.4798	2.4234	[5,1,2]
	Conocidas	7.7877 e-05	0.0307	6.5162 e-05	<b>0.0220</b>	9.6092 e-05	0.0289	[5,4,3]
PSO	Conocidas	7.7876 e-05	0.0308	6.5168 e-05	0.0222	9.6099 e-05	0.0290	[5,4,3]

Lo primero a destacar es que todos los algoritmos de optimización empleados logran llevar las variables de optimización a la región factible, es decir, los modelos óptimos obtenidos cumplen con las restricciones de controlabilidad impuestas en la función objetivo, lo cual se puede comprobar al corroborar que el determinante de las matrices de Sylvester formuladas a partir de la linealización del modelo difuso en distintos puntos de operación es distinto de cero, demostrando que la matriz es no singular, y en consecuencia la linealización es controlable. Sin embargo, no todas las estructuras analizadas logran converger a un modelo difuso controlable, por lo que son penalizadas por el factor  $\lambda$  en la función objetivo (4.12)-(4.13), descartando dichas estructuras inmediatamente como una solución, a pesar de que eventualmente puedan poseer un mejor RMSE.

De los resultados se puede observar que los mejores rendimientos se obtienen al utilizar condiciones iniciales para las variables de optimización dadas por la solución del problema de identificación de los parámetros de las consecuencias sin considerar restricciones de controlabilidad, esto debido a que al utilizar condiciones iniciales aleatorias los algoritmos empleados no logran converger a una mejor solución, producto de cumplirse alguna condición de término del algoritmo empleado, entregando resultados con peor desempeño.

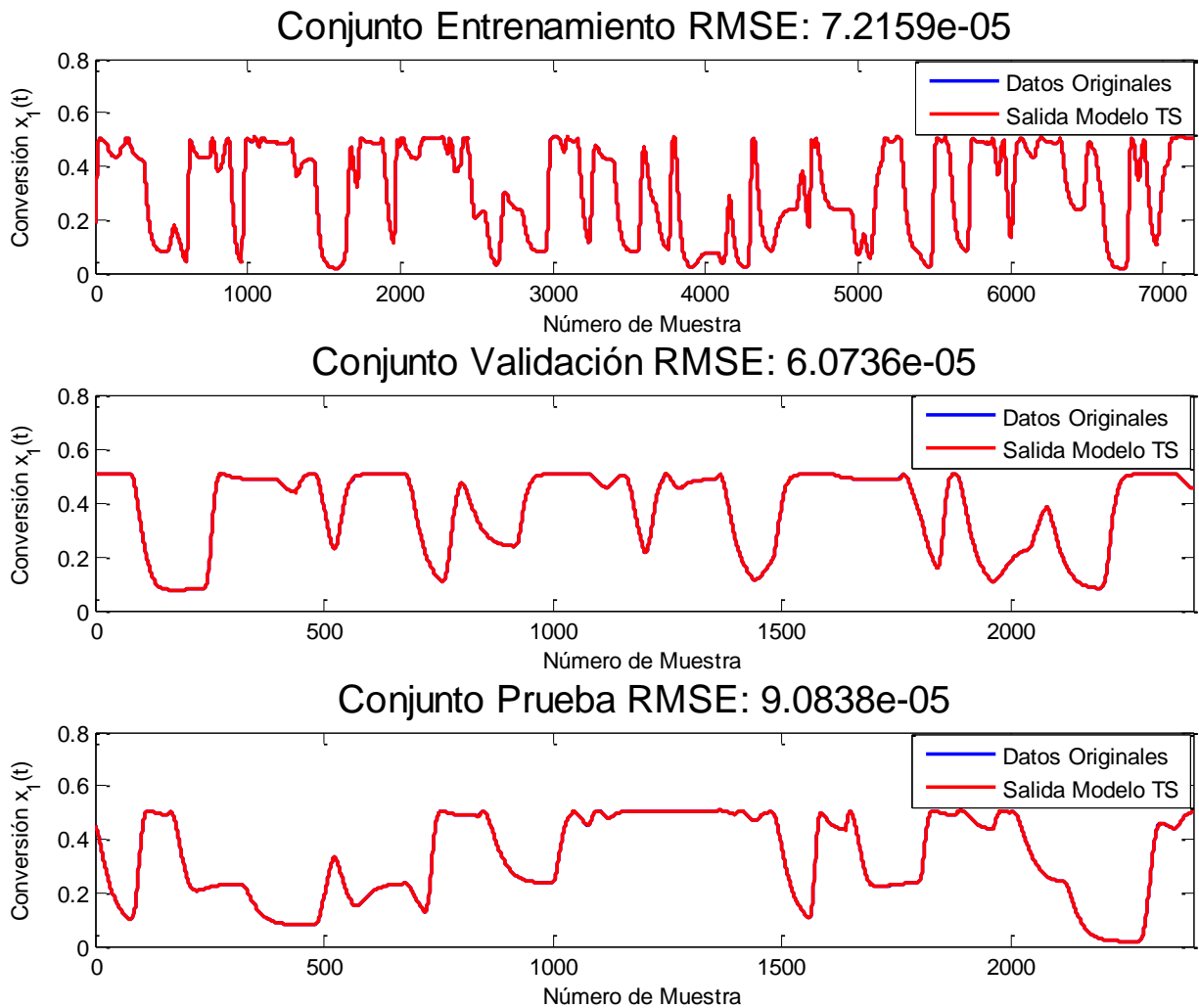
Por otro lado, se aprecia que al utilizar condiciones iniciales conocidas, tanto para identificación a uno como  $N$  pasos, los algoritmos de optimización logran converger a una misma estructura del modelo difuso, lo cual da a entender que la solución óptima está cerca de los parámetros obtenidos sin restricciones de controlabilidad. En particular para la identificación a  $N$  pasos y CI's conocidas (Tabla 4.3) el algoritmo Nelder-Mead demuestra un mejor rendimiento que PSO y Quasi-Newton, donde se aprecian pequeñas variaciones en los parámetros que repercuten en un mejor índice de desempeño, lo cual se debe principalmente a que Nelder-Mead posee un mejor manejo de funciones objetivos que presenten discontinuidades, producto de que no necesita información del gradiente de la función. En contraste, PSO y Quasi-Newton convergen exactamente al mismo modelo difuso, por lo que el algoritmo evolutivo no presenta grandes ventajas en comparación a algoritmos convencionales respecto a la solución del problema de optimización.

Cabe destacar que los errores de la predicción a un paso de los mejores modelos obtenidos por identificación a  $N$  pasos son similares a los logrados por los modelos generados utilizando identificación a un paso en la mayoría de los casos, por lo que utilizar sólo una identificación a  $N$  pasos generalmente otorga soluciones aceptables en ambos escenarios.

Note también que los mejores modelos de predicción obtenidos utilizando los datos de identificación en lazo abierto tienen premisas de mayor orden que el sistema original, esto debido a que la estructura del modelo difuso empleada no permite ajustar óptimamente las no linealidades de la planta original presentes en los datos de identificación utilizando la misma cantidad de estados. Luego, la inclusión de estados ficticios permite representar de mejor manera dichas dinámicas, corroborando que no se genere sobreajuste mediante la evaluación del modelo por validación cruzada.

Los resultados obtenidos demuestran un gran desempeño respecto al error de predicción, por lo que aumentar la complejidad de los modelos mediante la inclusión de más reglas ( $reglas_{max} > 5$ ) no mejoraría considerablemente los resultados.

En la Figura 4.7 se presentan los resultados del mejor modelo difuso para predicción a un paso (utilizando cualquier *solver* con CI's conocidas), donde se observa que el modelo obtenido es capaz de predecir (casi) perfectamente el comportamiento de la planta cuando se tiene información del instante anterior.



**Figura 4.7: Resultado Mejor Modelo Identificación a un Paso del CSTR con Datos en Lazo Abierto. Arriba: Conjunto de Entrenamiento. Medio: Conjunto de Validación. Abajo: Conjunto de Prueba.**

En el caso de la identificación a un paso, el modelo óptimo obtenido posee la estructura resumida en la Tabla 4.4. Note que los parámetros de las consecuencias asociados a los regresores de la entrada poseen valores bastante pequeños en comparación a los parámetros asociados a los regresores de la salida, esto debido al rango de operación de ambas variables.



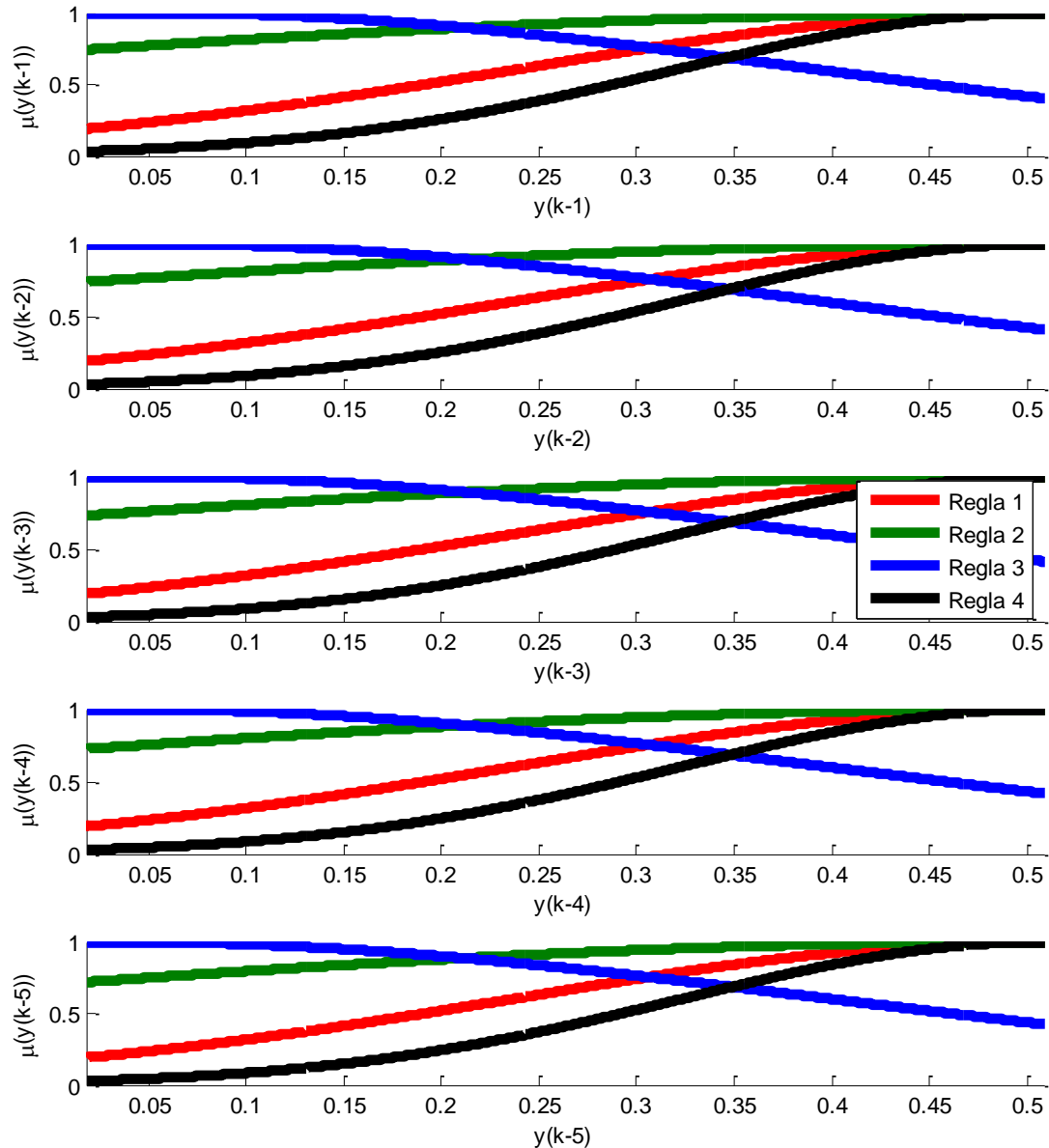
**Tabla 4.4: Estructura Mejor Modelo Identificación a un Paso del CSTR con Datos en Lazo Abierto.**

<b>Premisas</b>	$y(k-1), \dots, y(k-5), u(k-1), \dots, u(k-4)$		
<b>Cantidad de Reglas</b>	4		
<b>Parámetros Premisas</b>	$R_1$	$a_{1,i}$	[3.7108, 3.7101, 3.7099, 3.7110, 3.7096, 0.0172, 0.0171, 0.0171, 0.0203]
		$b_{1,i}$	[0.5087, 0.5087, 0.5087, 0.5087, 0.5087, 490.0, 490.0, 490.0 407.552]
	$R_2$	$a_{2,i}$	[1.5677, 1.5910, 1.6123, 1.6341, 1.6556, 0.0136, 0.0135, 0.0135, 0.0136]
		$b_{2,i}$	[0.5087, 0.5087, 0.5087, 0.5087, 0.5087, 395.9522, 395.9717, 395.9955, 395.9602]
	$R_3$	$a_{3,i}$	[2.9712, 2.9243, 2.8731, 2.8093, 2.7549, 0.0105, 0.0105, 0.0105, 0.0105]
		$b_{3,i}$	[0.0556, 0.0514, 0.0464, 0.0394, 0.0339, 307.3698, 307.4057, 307.4371, 307.4234]
	$R_4$	$a_{4,i}$	[5.3380, 5.3694, 5.3981, 5.4239, 5.4483, 0.0117, 0.0117, 0.0117, 0.0117]
		$b_{4,i}$	[0.5087, 0.5087, 0.5087, 0.5087, 0.5087, 481.6553, 482.4372, 481.7727, 481.6281]
<b>Parámetros Consecuencias</b>	$R_1$	$\theta_1^T = [4.5282, -7.7195, 5.5800, -1.1976, -0.2007, 4.2596 \cdot 10^{-5}, -8.169 \cdot 10^{-5}, 2.8367 \cdot 10^{-5}, 1.9654 \cdot 10^{-5}]$	
	$R_2$	$\theta_2^T = [3.2176, -3.9894, 2.4760, -0.8698, 0.1647, 1.6294 \cdot 10^{-5}, -1.4578 \cdot 10^{-5}, 1.3411 \cdot 10^{-7}, -9.4622 \cdot 10^{-7}]$	
	$R_3$	$\theta_3^T = [2.9893, -3.2370, 1.4596, -0.1937, -0.0191, 1.9909 \cdot 10^{-6}, -1.3091 \cdot 10^{-6}, -1.6136 \cdot 10^{-6}, 1.0384 \cdot 10^{-6}]$	
	$R_4$	$\theta_4^T = [2.5926, -1.8277, -0.1204, 0.3695, -0.0099, -3.3111 \cdot 10^{-5}, 5.5442 \cdot 10^{-5}, 3.7964 \cdot 10^{-7}, -2.6373 \cdot 10^{-5}]$	

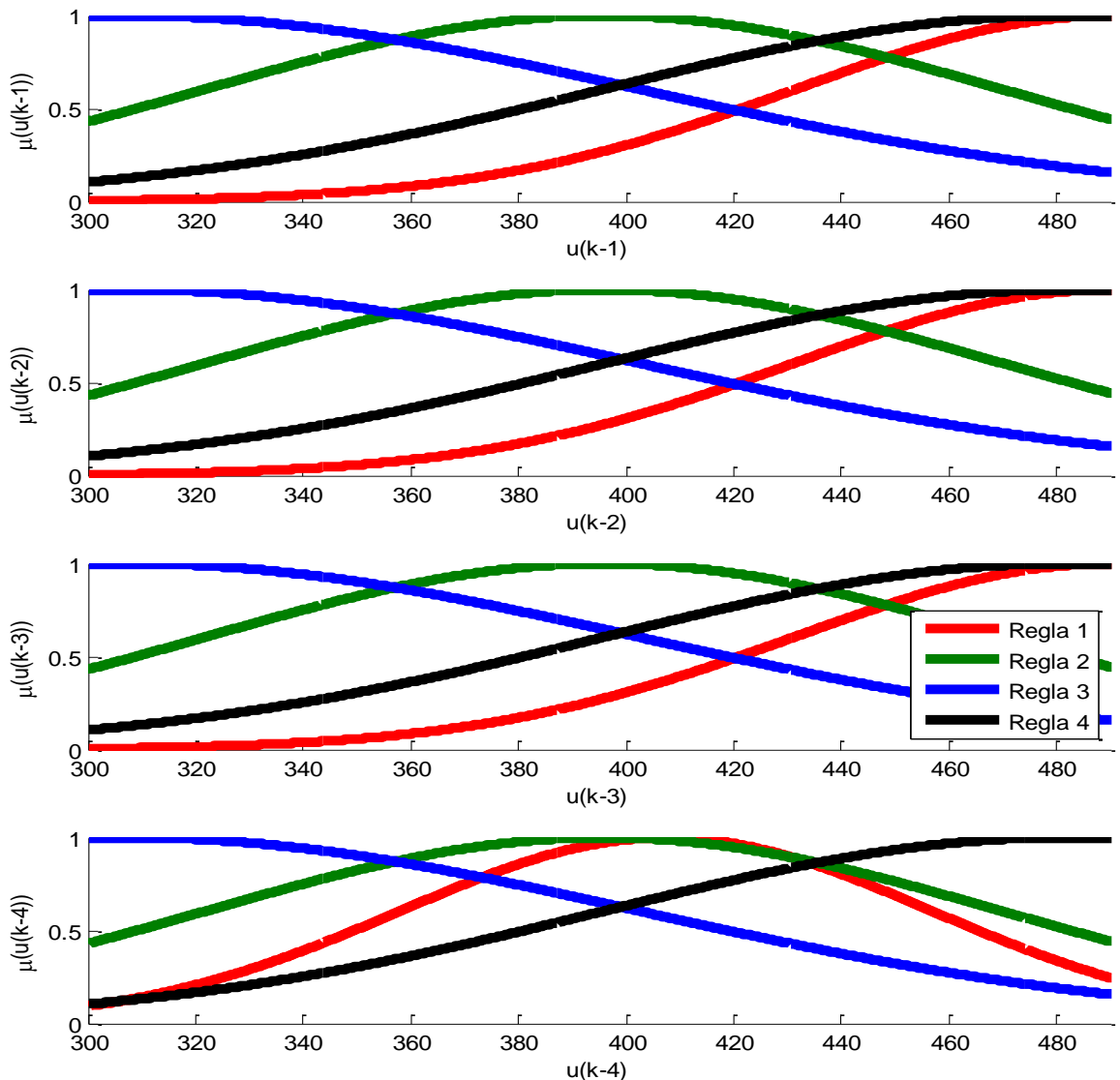
La Figura 4.8 y la Figura 4.9 ilustran los conjuntos difusos asociados a los regresores de la salida y a los regresores de la entrada respectivamente. En dichas gráficas se observa que existen conjuntos difusos que se sobreponen unos con otros, lo que da a entender que la estructura podría ser optimizada mejor si se considerasen 3 reglas en lugar de 4, es decir, el algoritmo CCM presentado en la Sección 2.4 podría ser utilizado para mezclar aquellos *clusters* que están relativamente cerca. De hecho, el modelo obtenido utilizando la misma cantidad de regresores para la salida y entrada, pero con 3 reglas en lugar de 4, tiene un RMSE a un paso en el conjunto de validación de  $6.5168e-05$ , bastante cercano al  $6.0736e-05$  obtenido por el modelo óptimo. Luego, sería posible emplear en este caso el principio de parsimonia introducido en la Sección 2.4, donde el modelo más simple y con desempeño similar podría ser considerado en lugar del modelo óptimo obtenido con el algoritmo de identificación.

Por otro lado, llama la atención las grandes varianzas que presentan los conjuntos difusos asociados a  $y(k-j)$ , lo cual da a entender que dichas variables no son tan

significantes para discriminar entre las distintas reglas del modelo. Lo anterior se puede deber a que el *toolbox* utilizado solamente permite utilizar las mismas variables tanto en las premisas como consecuencias del modelo. No obstante, a pesar de lo anterior, los conjuntos difusos encontrados permiten un ajuste óptimo (en términos del RMSE) de la salida del modelo difuso a las dinámicas del sistema.



**Figura 4.8: Conjuntos Difusos de los Regresores  $y(k - j)$  para el Mejor Modelo Identificación a un Paso del CSTR con Datos en Lazo Abierto.**

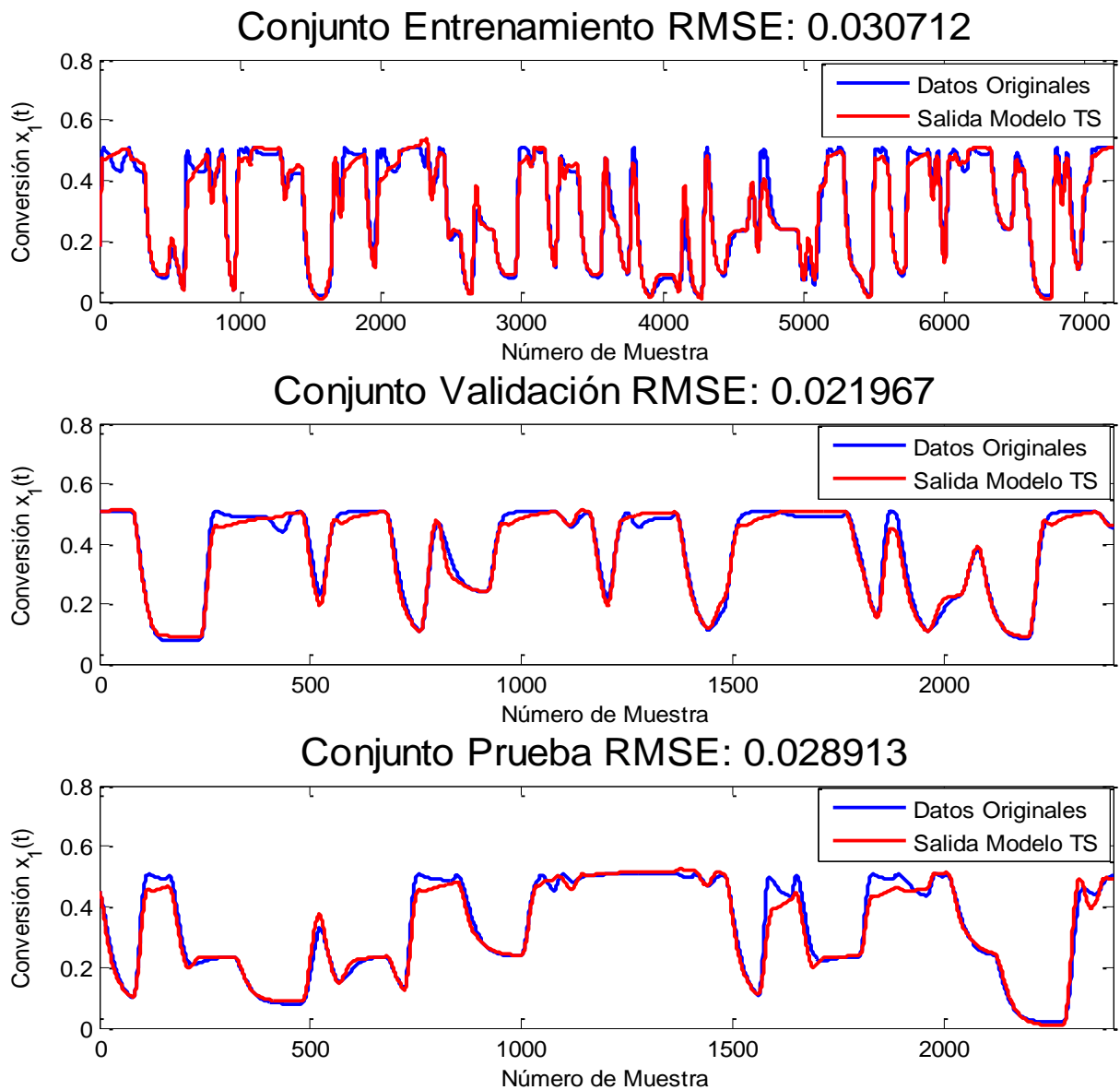


**Figura 4.9: Conjuntos Difusos de los Regresores  $u(k - j)$  para el Mejor Modelo Identificación a un Paso del CSTR con Datos en Lazo Abierto.**

A su vez, si se analizan los modelos locales se puede verificar, mediante el método de la matriz de Sylvester, que todos ellos son controlables, por lo que es posible corroborar dicha condición también a nivel local. A su vez, es posible analizar la estabilidad de cada consecuencia mediante el estudio de los polos de cada modelo. Luego, dado que los consecuentes en estudio son de tiempo discreto, se dice que un sistema lineal es estable si todos sus polos pertenecen al círculo unitario dentro del espacio complejo, es decir,  $|q_i| < 1, i = 1, \dots, n_y$ , con  $q_i$  el  $i$ -ésimo polo del sistema [8]. En este caso se tiene que sólo la segunda y tercera regla poseen consecuentes estables, mientras que la primera y cuarta regla poseen modelos que tienen al menos un polo fuera del círculo unitario, lo cual genera que todo el modelo difuso sea considerado como inestable [108], contradiciendo que la planta original es estable en la zona de operación en la que se

identifica. Lo anterior puede deberse a que la discretización del sistema puede producir una pérdida de información, y en consecuencia algunas propiedades de la planta original no se heredan al modelo identificado.

Por otro lado, en la Figura 4.10 se puede apreciar el desempeño del mejor modelo para identificación a  $N$  pasos (utilizando Nelder-Mead con CI's conocidas), donde se puede observar que el modelo difuso logra representar las tendencias de la planta con cierta exactitud, aun cuando posee información de la salida real del sistema de los primeros instantes solamente.



**Figura 4.10: Resultado Mejor Modelo Identificación a  $N$  Pasos del CSTR con Datos en Lazo Abierto. Arriba: Conjunto de Entrenamiento. Medio: Conjunto de Validación. Abajo: Conjunto de Prueba.**

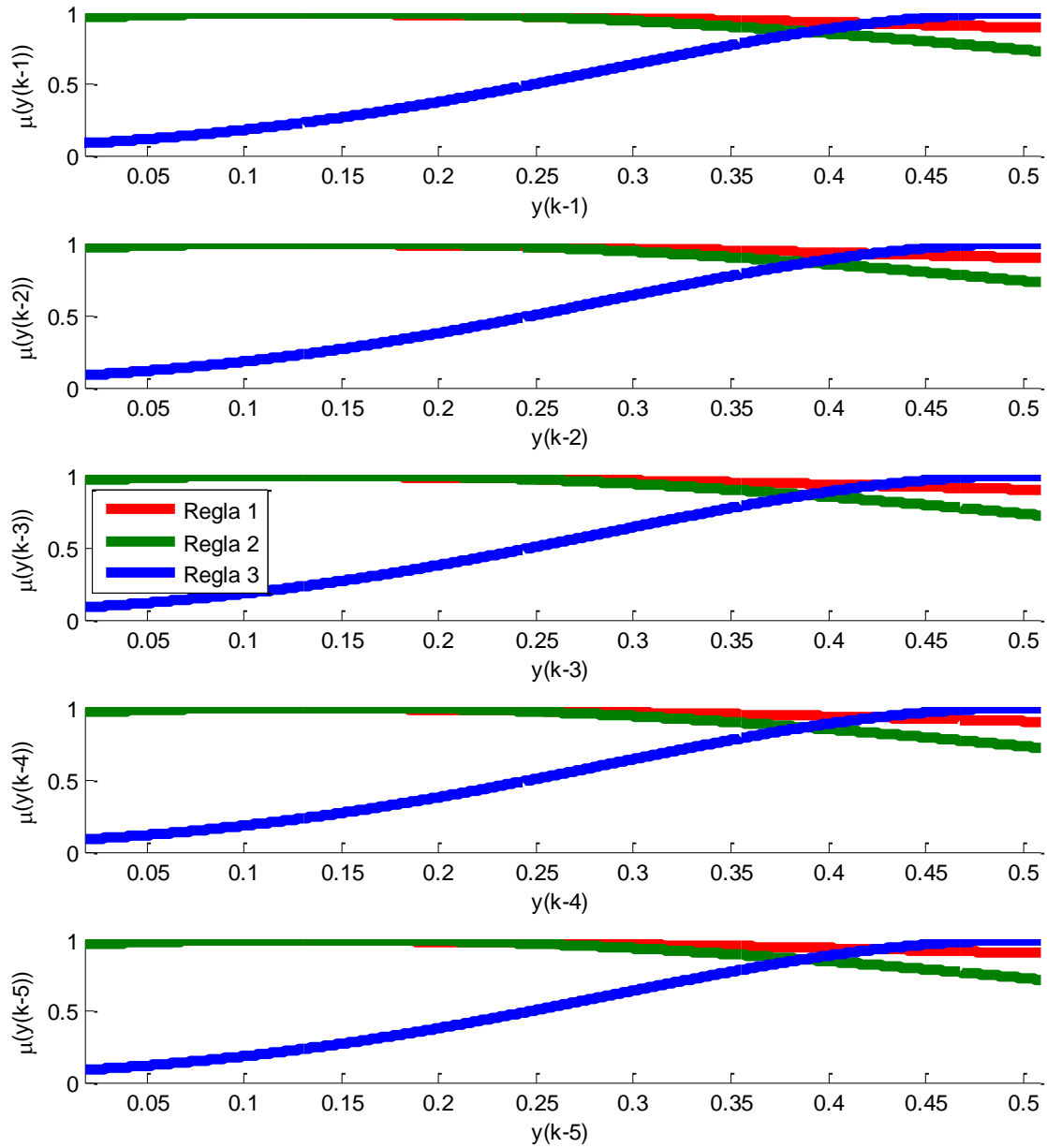
En el caso de la identificación a  $N$  pasos, el modelo óptimo obtenido posee la estructura resumida en la Tabla 4.5.

**Tabla 4.5: Estructura Mejor Modelo Identificación a  $N$  Pasos del CSTR con Datos en Lazo Abierto.**

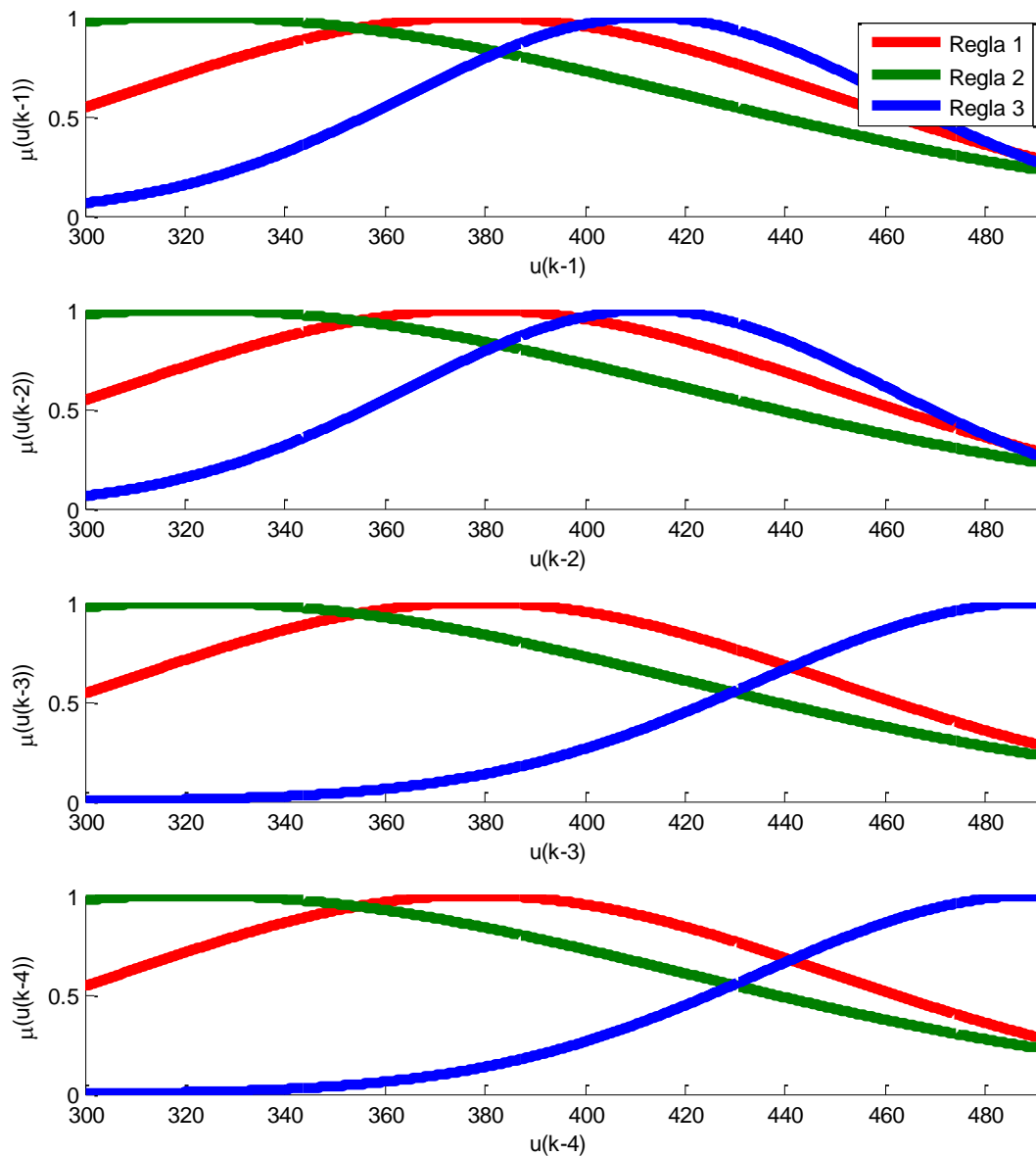
<b>Premisas</b>	$y(k-1), \dots, y(k-5), u(k-1), \dots, u(k-4)$		
<b>Cantidad de Reglas</b>	3		
<b>Parámetros Premisas</b>	$R_1$	$a_{1,i}$	[0.9737, 0.9572, 0.9421, 0.9278, 0.9147, 0.0141, 0.0141, 0.0141, 0.0141]
		$b_{1,i}$	[0.0179, 0.0179, 0.0179, 0.0179, 0.0179, 378.4306, 378.4083, 378.4626, 378.4777]
	$R_2$	$a_{2,i}$	[2.1632, 2.1656, 2.1871, 2.1944, 2.1908, 0.0101, 0.0101, 0.0101, 0.0101]
		$b_{2,i}$	[0.1416, 0.1414, 0.1412, 0.1404, 0.1392, 321.3822, 321.3470, 321.2516, 321.3817]
	$R_3$	$a_{3,i}$	[4.5380, 4.5289, 4.5222, 4.5152, 4.5091, 0.0208, 0.0209, 0.0181, 0.0182]
		$b_{3,i}$	[0.5087, 0.5087, 0.5087, 0.5087, 0.5087, 412.7435, 412.7085, 490.0, 490.0]
<b>Parámetros Consecuencias</b>	$R_1$	$\theta_1^T = [3.4464, -4.6569, 3.2455, -1.3138, 0.2784, 1.6596 \cdot 10^{-5}, -1.5283 \cdot 10^{-5}, 1.7599 \cdot 10^{-6}, -2.4446 \cdot 10^{-6}]$	
	$R_2$	$\theta_2^T = [2.9336, -3.0506, 1.2321, -0.0744, -0.0414, -1.07 \cdot 10^{-6}, 1.4859 \cdot 10^{-6}, -2.0908 \cdot 10^{-6}, 1.6135 \cdot 10^{-6}]$	
	$R_3$	$\theta_3^T = [2.9893, -3.2370, 1.4596, -0.1937, -0.0191, -1.861 \cdot 10^{-5}, 2.7991 \cdot 10^{-5}, -5.1232 \cdot 10^{-6}, -4.0207 \cdot 10^{-6}]$	

En la Figura 4.11 se observa los conjuntos difusos asociados a los regresores de la salida  $y(k-j)$ , donde se ve que la Regla 1 y la Regla 2 se sobreponen la una con la otra, sin embargo, en la Figura 4.12 se presentan los conjuntos difusos asociados a los regresores de la entrada, en donde se observa que ninguna de las reglas se sobrepone, por lo que en este caso las 3 reglas utilizadas por el modelo difuso óptimo se distribuyen de manera correcta, permitiendo una buena identificación en horizontes de predicción mayores, obteniendo bajos índices RMSE y una representación adecuada de los comportamientos de la planta, como se observa en la Figura 4.10.

A su vez, nuevamente los conjuntos difusos asociados a los regresores de la salida presentan grandes varianzas, por lo que no son una buena variable para ser utilizadas como premisas. Cabe destacar que el algoritmo GK ajusta los *clusters* en un espacio multidimensional, mientras que los conjuntos expuestos en las gráficas son las proyecciones de dichos *clusters* en el espacio de entrada, lo cual puede distorsionar la apreciación de variabilidad de los conjuntos. No obstante, se destaca el buen desempeño del modelo a pesar de estas consideraciones.



**Figura 4.11: Conjuntos Difusos de los Regresores  $y(k - j)$  para el Mejor Modelo Identificación a  $N$  Pasos del CSTR con Datos en Lazo Abierto.**



**Figura 4.12: Conjuntos Difusos de los Regresores  $u(k - j)$  para el Mejor Modelo Identificación a  $N$  Pasos del CSTR con Datos en Lazo Abierto.**

Por el lado de los modelos locales, nuevamente todos cumplen con ser controlables, sin embargo, la primera regla posee un consecuente inestable, afectando nuevamente a la estabilidad del modelo difuso completo, repaldando la hipótesis de que un modelo no necesariamente hereda las características del sistema original.

- **IDENTIFICACIÓN EN LAZO CERRADO**

Similar a lo realizado en la identificación en lazo abierto, en la Tabla 4.6 y en la Tabla 4.7 se presentan los resultados obtenidos del proceso de identificación en lazo cerrado a

uno y  $N$  pasos respectivamente. En el caso de PSO se utilizan los mismos parámetros presentes en la Tabla 4.1.

**Tabla 4.6: Resultados Identificación del CSTR en Lazo Cerrado a un Paso.**

<i>Solver</i>	<i>CI's</i>	RMSE Entrenamiento		RMSE Validación		RMSE Prueba		$[n_y^*, n_u^*, reglas^*]$
		1	$N$	1	$N$	1	$N$	
		Paso	Pasos	Paso	Pasos	Paso	Pasos	
QN	Aleatorias	4.3937 e-04	0.1092	4.3925 e-04	0.0823	4.1725 e-04	0.0992	[2,2,1]
	Conocidas	7.9479 e-05	0.0411	8.0902 e-05	0.0288	7.7913 e-05	0.0334	[4,4,4]
NM	Aleatorias	6.3110 e-04	0.1044	6.3613 e-04	0.0829	6.0249 e-04	0.0938	[5,1,1]
	Conocidas	7.6652 e-05	0.0297	<b>7.7969 e-05</b>	0.0224	7.4316 e-05	0.0269	[4,3,5]
PSO	Conocidas	7.9479 e-05	0.0411	8.0902 e-05	0.0288	7.7913 e-05	0.0334	[4,4,4]

**Tabla 4.7: Resultados Identificación del CSTR en Lazo Cerrado a  $N$  Pasos.**

<i>Solver</i>	<i>CI's</i>	RMSE Entrenamiento		RMSE Validación		RMSE Prueba		$[n_y^*, n_u^*, reglas^*]$
		1	$N$	1	$N$	1	$N$	
		Paso	Pasos	Paso	Pasos	Paso	Pasos	
QN	Aleatorias	0.0226	0.1042	0.0207	0.0848	0.0205	0.0932	[3,3,1]
	Conocidas	7.6620 e-05	0.0285	7.7915 e-05	<b>0.0231</b>	7.4306 e-05	0.0275	[4,3,5]
NM	Aleatorias	35.2638	2.5670	38.7360	2.8369	33.1325	2.5420	[5,1,2]
	Conocidas	7.6888 e-05	0.0206	7.8222 e-05	0.0262	7.4594 e-05	0.0188	[4,3,5]
PSO	Conocidas	7.6620 e-05	0.0285	7.7915 e-05	<b>0.0231</b>	7.4306 e-05	0.0275	[4,3,5]

Nuevamente los algoritmos de optimización empleados logran converger dentro de la región factible, por lo que los resultados presentes representan el desempeño de modelos controlables en la región en que operan, aunque cabe destacar nuevamente que no todas las estructuras estudiadas cumplen con las restricciones de controlabilidad, descartando dichos modelos como soluciones óptimas.

Otra tendencia que se repite en los resultados es que los mejores desempeños se obtienen al utilizar como condiciones iniciales la solución del problema de identificación de parámetros de las consecuencias sin incluir restricciones de controlabilidad, logrando



mejor convergencia de los parámetros que cuando se utilizan condiciones iniciales aleatorias, lo cual se refleja en un mejor índice de desempeño.

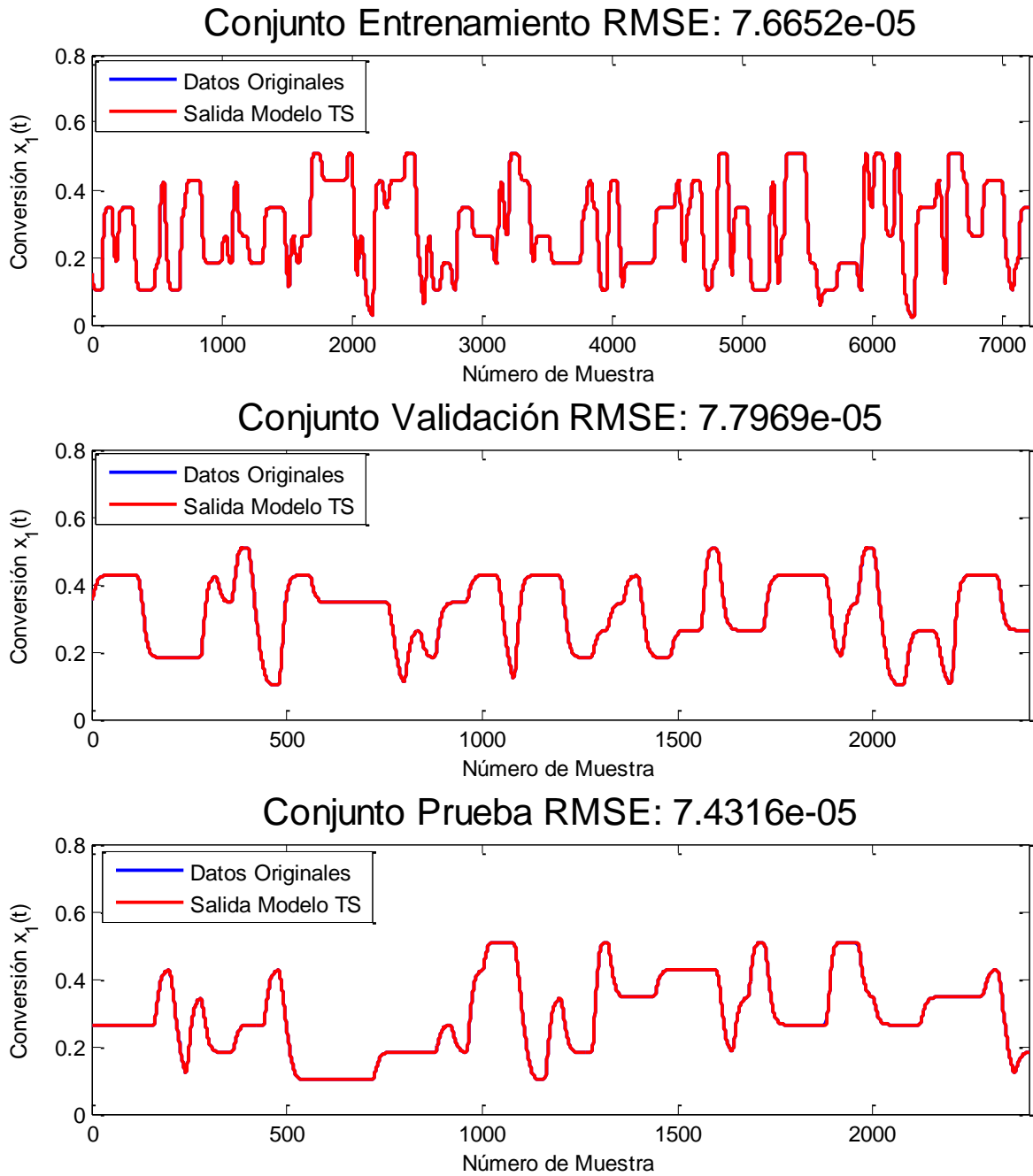
Cabe destacar que para identificación utilizando los datos en lazo cerrado tanto a uno como a  $N$  pasos se obtienen errores en los mismos órdenes de magnitud que los presentes en la identificación en lazo abierto, obteniendo buenos desempeños bajo el criterio del RMSE, demostrando que ambos experimentos son lo suficientemente informativos como para poder ajustar un modelo correctamente a las dinámicas presentes. Dado lo anterior, se desestima aumentar la complejidad de los modelos generados mediante la inclusión de un mayor número de reglas, debido a que en la práctica no se obtendrían mejoras considerables.

Para identificación a un paso, a diferencia de los resultados de identificación con datos en lazo abierto, los algoritmos de optimización no convergen a una misma estructura, donde nuevamente se aprecia que el algoritmo Nelder-Mead saca ventaja al algoritmo evolutivo PSO y al método convencional Quasi-Newton debido a su forma de abordar la función objetivo.

De los resultados se puede apreciar que el mejor modelo obtenido por identificación a  $N$  pasos posee un mejor índice de desempeño al predecir el comportamiento de la planta a un paso que los mismos modelos obtenidos por identificación a un paso, reafirmando los resultados obtenidos en la identificación en lazo abierto, donde utilizar sólo una identificación a  $N$  pasos obtiene buenos resultados en los dos horizontes de predicción utilizados. Por otro lado, y a diferencia de la identificación en lazo abierto, se tiene que el mejor modelo obtenido por identificación a un paso posee mejor índice de desempeño al predecir a  $N$  pasos que el mejor modelo obtenido por identificación a  $N$  pasos. Lo anterior da a entender que el tipo de datos utilizados genera que los *solvers* empleados no logren converger a una solución óptima, obteniendo mejores resultados en distintos escenarios. A su vez, se destaca el hecho que los modelos óptimos obtenidos tanto en identificación a un paso como a  $N$  pasos poseen la misma estructura.

Los modelos de predicción óptimos obtenidos, tanto a  $N$  pasos como a un paso, son de orden 4, por lo que la clase de modelos difusos utilizados tampoco se pueden adecuar óptimamente a los datos generados en lazo cerrado utilizando el mismo orden que la planta original, siendo necesaria la inclusión de estados ficticios para representar de mejor manera el comportamiento del sistema en los distintos puntos de operación.

En la Figura 4.13 se observa el resultado para cada conjunto de datos del mejor modelo de predicción a un paso obtenido utilizando los datos en lazo cerrado (Nelder-Mead con CI's conocidas), donde se aprecia que la diferencia entre los índices de desempeño de los mejores modelos de predicción a un paso obtenidos con datos en lazo abierto y cerrado es despreciable, obteniéndose en ambos casos una identificación (casi) perfecta al tener información del instante anterior.



**Figura 4.13: Resultado Mejor Modelo Identificación a un Paso del CSTR con Datos en Lazo Cerrado. Arriba: Conjunto de Entrenamiento. Medio: Conjunto de Validación. Abajo: Conjunto de Prueba.**

El modelo óptimo obtenido en el proceso de identificación a un paso posee la estructura resumida en la Tabla 4.8.

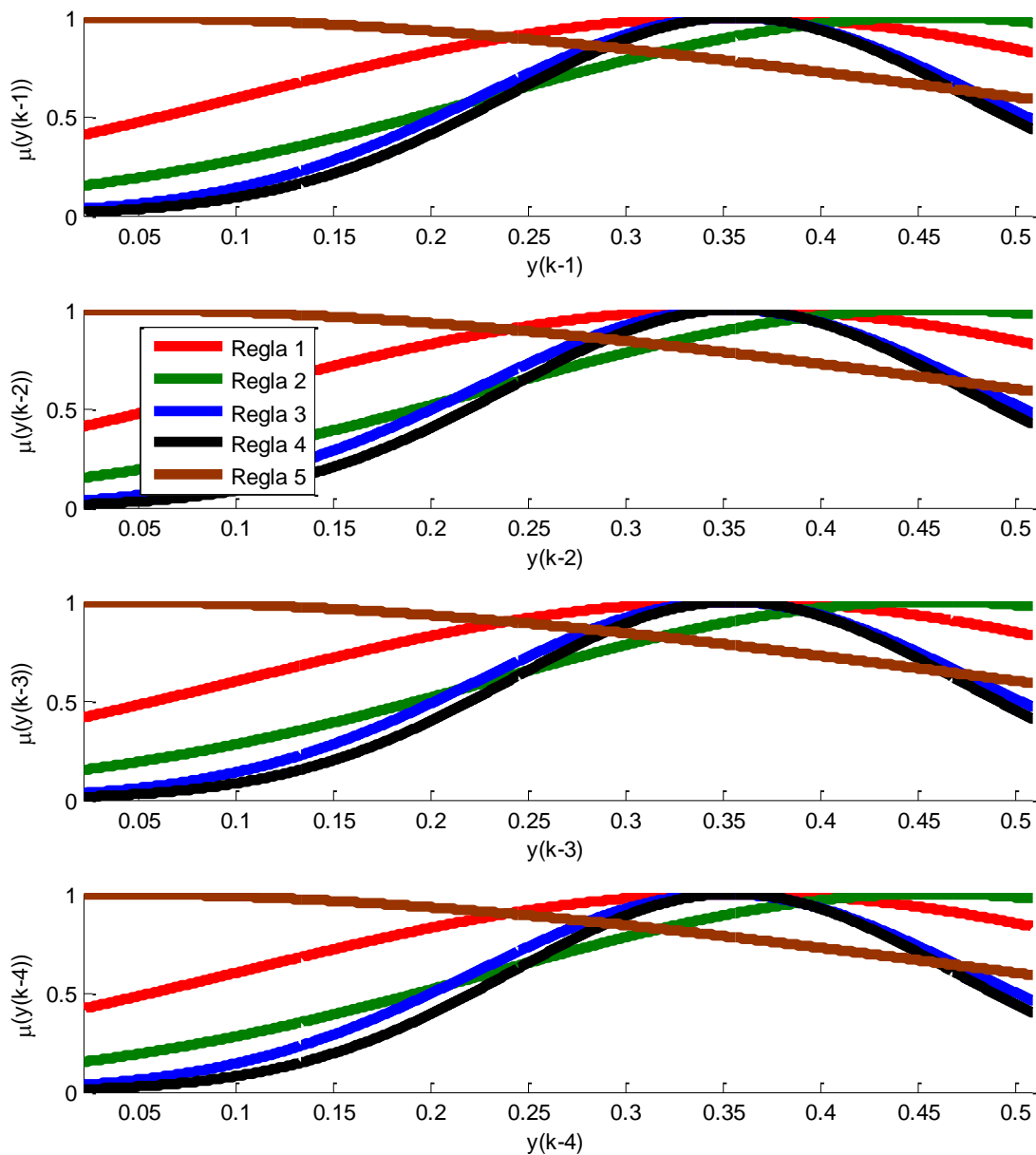
**Tabla 4.8: Estructura Mejor Modelo Identificación a un Paso del CSTR con Datos en Lazo Cerrado.**

<b>Premisas</b>	$y(k-1), \dots, y(k-4), u(k-1), \dots, u(k-3)$		
<b>Cantidad de Reglas</b>	5		
<b>Parámetros Premisas</b>	$R_1$	$a_{1,i}$	[4.0392, 4.0143, 3.9778, 3.9369, 0.0488, 0.0494, 0.0500]
		$b_{1,i}$	[0.3542, 0.3546, 0.3555, 0.3566, 300.0, 300.0, 300.0]
	$R_2$	$a_{2,i}$	[4.4754, 4.4695, 4.4591, 4.4519, 0.0205, 0.0181, 0.0430]
		$b_{2,i}$	[0.4570, 0.4574, 0.4581, 0.4585, 408.6303, 413.7021, 490.0]
	$R_3$	$a_{3,i}$	[7.8107, 7.7783, 7.8745, 7.8611, 0.0324, 0.0321, 0.0327]
		$b_{3,i}$	[0.3551, 0.3531, 0.3528, 0.3509, 377.1135, 376.8500, 378.1738]
	$R_4$	$a_{4,i}$	[8.5190, 8.6207, 8.7242, 8.8208, 0.0390, 0.0394, 0.0390]
		$b_{4,i}$	[0.3574, 0.3568, 0.3561, 0.3554, 387.8083, 387.8258, 388.2685]
	$R_5$	$a_{5,i}$	[2.1203, 2.1159, 2.1113, 2.1064, 0.0331, 0.0327, 0.0319]
		$b_{5,i}$	[0.0215, 0.0215, 0.0215, 0.0215, 349.4982, 349.0090, 348.5072]
<b>Parámetros Consecuencias</b>	$R_1$	$\theta_1^T = [2.7604, -2.6747, 1.0309, -0.1177, 9.2641 \cdot 10^{-6}, -6.5779 \cdot 10^{-6}, -2.6228 \cdot 10^{-6}]$	
	$R_2$	$\theta_2^T = [3.2321, -3.8753, 2.0336, -0.3910, 5.2741 \cdot 10^{-6}, -3.7887 \cdot 10^{-6}, -6.2584 \cdot 10^{-7}]$	
	$R_3$	$\theta_3^T = [2.9921, -3.1227, 1.2606, -0.1305, 1.6102 \cdot 10^{-5}, -1.2634 \cdot 10^{-5}, -3.1484 \cdot 10^{-6}]$	
	$R_4$	$\theta_4^T = [2.4297, -1.6281, -0.0565, 0.2548, 1.5254 \cdot 10^{-5}, -1.7942 \cdot 10^{-6}, -1.3335 \cdot 10^{-5}]$	
	$R_5$	$\theta_5^T = [2.9380, -3.0463, 1.2664, -0.1582, 5.3301 \cdot 10^{-6}, -4.1153 \cdot 10^{-6}, -1.1752 \cdot 10^{-6}]$	

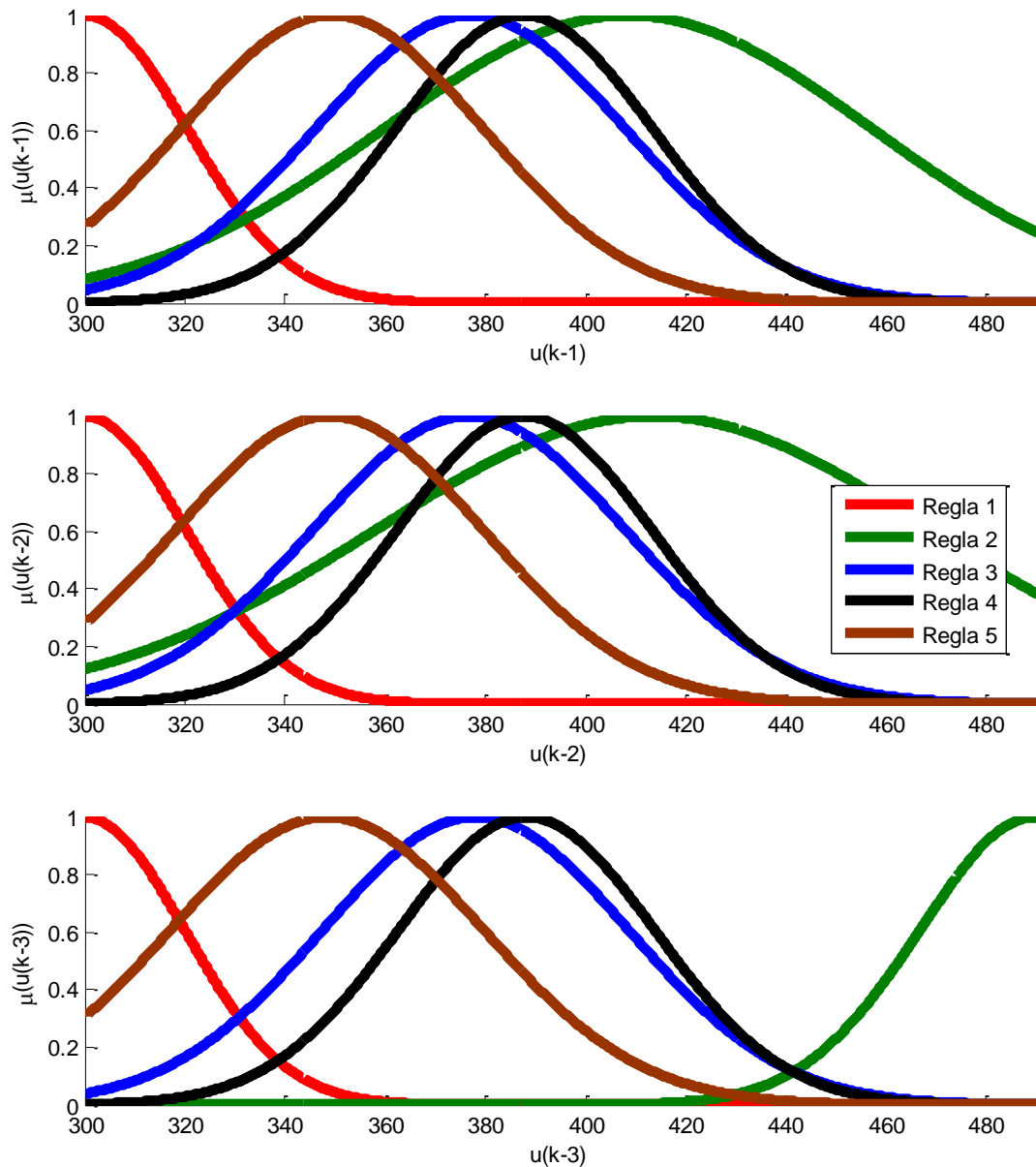
En la Figura 4.14 y en la Figura 4.15 se presentan los conjuntos difusos asociados a cada premisa del modelo descrito en la Tabla 4.8. En cuanto a los conjuntos difusos de los regresores de la salida, se observa que la Regla 3 y la Regla 4 son sobrepuestas por la función de pertenencia de la Regla 1, sin embargo, se tiene un comportamiento dispar en los conjuntos difusos asociados a los regresores de la entrada, donde las tres reglas en cuestión abarcan distintas regiones de operación. A su vez, se tiene que la Regla 3 y la Regla 4 presentan funciones de pertenencia relativamente cercanas, tanto en los conjuntos difusos asociados a las premisas  $y(k-j)$  como en los conjuntos asociados a

las premisas  $u(k - j)$ , por lo que la estructura podría ser optimizada aplicando el criterio CCM.

Por otro lado, algunos de los conjuntos difusos de los regresores de la salida aportan información que permite identificar el rango de operación en que se trabaja, a diferencia de los modelos obtenidos con datos en lazo abierto, y por lo tanto dichas variables se debiesen mantener como premisas del modelo. Adicionalmente, los conjuntos difusos de los regresores de la entrada presentan una forma más definida, logrando identificar de mejor manera las zonas de operación del sistema.



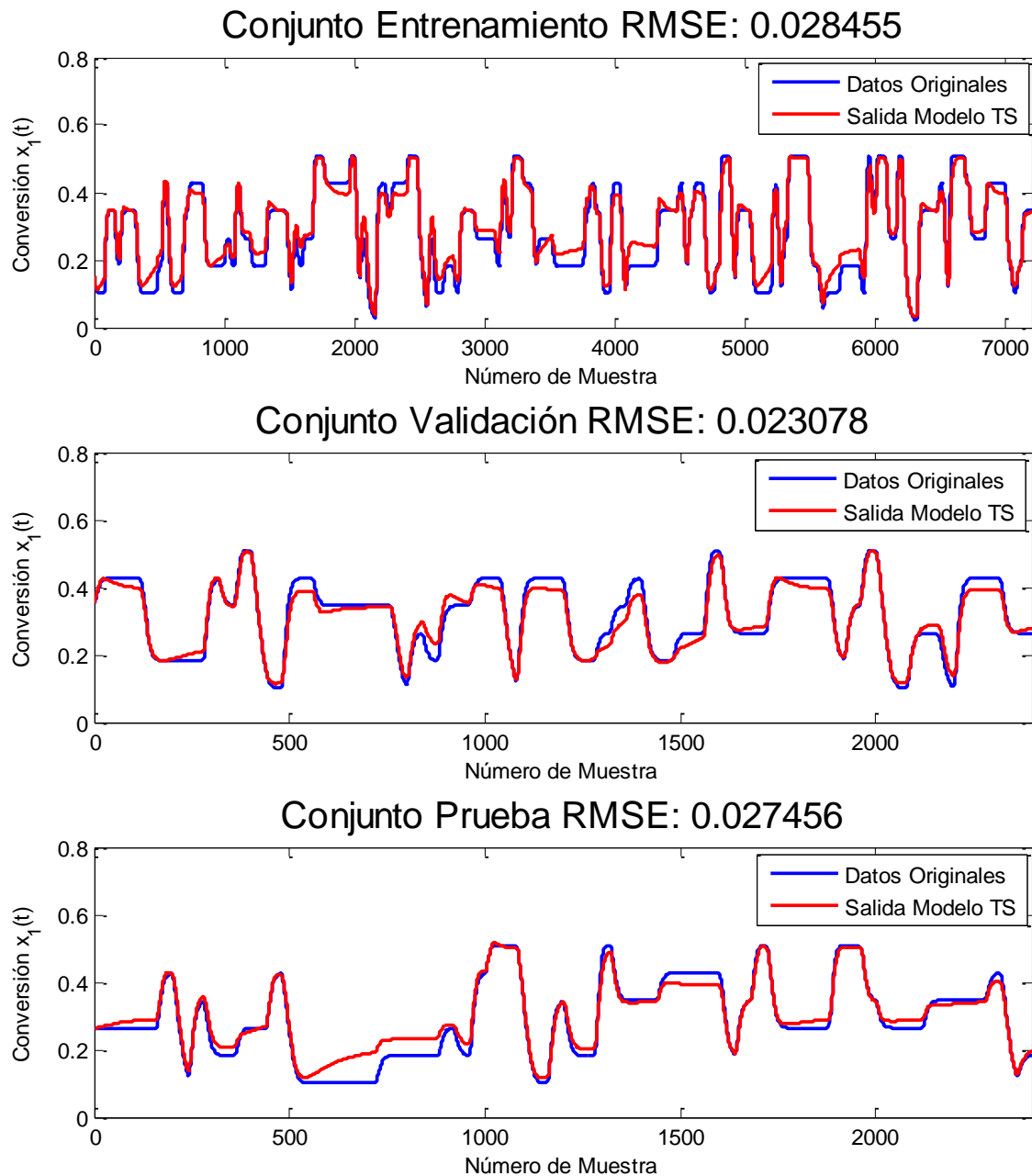
**Figura 4.14: Conjuntos Difusos de los Regresores  $y(k - j)$  para el Mejor Modelo Identificación a un Paso del CSTR con Datos en Lazo Cerrado.**



**Figura 4.15: Conjuntos Difusos de los Regresores  $u(k - j)$  para el Mejor Modelo Identificación a un Paso del CSTR con Datos en Lazo Cerrado.**

En cuanto a los modelos locales, nuevamente todos cumplen con ser controlables. A su vez, y a diferencia de los resultados obtenidos en la identificación en lazo abierto, todos los modelos locales son estables. Luego, estudiando la estabilidad del modelo difuso completo según lo expuesto en la Sección 8.5 de anexos, se concluye que el modelo difuso es estable. Lo anterior se justifica en base de que los datos son tomados en lazo cerrado, luego la planta actúa en todo momento de manera estable, debido al proceder del controlador.

En cuanto a la identificación a  $N$  pasos, en la Figura 4.16 se presenta el desempeño del modelo óptimo de predicción a  $N$  pasos (PSO-QN utilizando CI's conocidas), donde se observa que el modelo es capaz de representar las no linealidades del CSTR con un error de predicción bajo.



**Figura 4.16: Resultado Mejor Modelo Identificación a  $N$  Pasos del CSTR con Datos en Lazo Cerrado. Arriba: Conjunto de Entrenamiento. Medio: Conjunto de Validación. Abajo: Conjunto de Prueba.**

A su vez, el modelo óptimo obtenido posee la estructura resumida en la Tabla 4.9.

**Tabla 4.9: Estructura Mejor Modelo Identificación a  $N$  Pasos del CSTR con Datos en Lazo Cerrado.**

<b>Premisas</b>	$y(k-1), \dots, y(k-4), u(k-1), \dots, u(k-3)$		
<b>Cantidad de Reglas</b>	5		
<b>Parámetros Premisas</b>	$R_1$	$a_{1,i}$	[4.0392, 4.0143, 3.9778, 3.9369, 0.0488, 0.0494, 0.0500]
		$b_{1,i}$	[0.3542, 0.3546, 0.3555, 0.3566, 300.0, 300.0, 300.0]
	$R_2$	$a_{2,i}$	[4.4754, 4.4695, 4.4591, 4.4519, 0.0205, 0.0181, 0.0430]
		$b_{2,i}$	[0.4570, 0.4574, 0.4581, 0.4585, 408.6303, 413.7021, 490.0]
	$R_3$	$a_{3,i}$	[7.8107, 7.7783, 7.8745, 7.8611, 0.0324, 0.0321, 0.0327]
		$b_{3,i}$	[0.3551, 0.3531, 0.3528, 0.3509, 377.1135, 376.8500, 378.1738]
	$R_4$	$a_{4,i}$	[8.5190, 8.6207, 8.7242, 8.8208, 0.0390, 0.0394, 0.0390]
		$b_{4,i}$	[0.3574, 0.3568, 0.3561, 0.3554, 387.8083, 387.8258, 388.2685]
	$R_5$	$a_{5,i}$	[2.1203, 2.1159, 2.1113, 2.1064, 0.0331, 0.0327, 0.0319]
		$b_{5,i}$	[0.0215, 0.0215, 0.0215, 0.0215, 349.4982, 349.0090, 348.5072]
<b>Parámetros Consecuencias</b>	$R_1$	$\theta_1^T = [2.7601, -2.6743, 1.0309, -0.1177, 9.2643 \cdot 10^{-6}, -6.5772 \cdot 10^{-6}, -2.6225 \cdot 10^{-6}]$	
	$R_2$	$\theta_2^T = [3.2323, -3.8757, 2.0337, -0.3911, 5.2751 \cdot 10^{-6}, -3.7893 \cdot 10^{-6}, -6.2589 \cdot 10^{-7}]$	
	$R_3$	$\theta_3^T = [2.9920, -3.1224, 1.2605, -0.1305, 1.6098 \cdot 10^{-5}, -1.2634 \cdot 10^{-5}, -3.1488 \cdot 10^{-6}]$	
	$R_4$	$\theta_4^T = [2.4295, -1.6279, -0.0565, 0.2548, 1.5258 \cdot 10^{-5}, -1.7502 \cdot 10^{-6}, -1.3335 \cdot 10^{-5}]$	
	$R_5$	$\theta_5^T = [2.9379, -3.0460, 1.2662, -0.1581, 5.3293 \cdot 10^{-6}, -4.1138 \cdot 10^{-6}, -1.1483 \cdot 10^{-6}]$	

Cabe destacar que, debido a que los modelos óptimos obtenidos por identificación a un paso y a  $N$  pasos poseen la misma estructura, los parámetros de las premisas,  $a_{r,i}$  y  $b_{r,i}$ , son los mismos, dado a que no se altera la etapa de *clustering* en la metodología propuesta. Por otro lado, los parámetros de los consecuentes sólo varían mínimamente, sin embargo, dicha variación influye con respecto al desempeño del modelo.

Los modelos locales del modelo difuso óptimo obtenido por identificación a  $N$  pasos cumplen con ser controlables y estables, reafirmando los resultados obtenidos por identificación a un paso al utilizar datos en lazo cerrado.

Una vez generados los modelos de predicción difusos que posean la propiedad de ser controlables, se procede a validar la metodología propuesta mediante el diseño de un controlador predictivo difuso sobre el reactor de estanque continuamente agitado.

#### 4.2.1.2. Control Predictivo Difuso del CSTR

En esta oportunidad se compara el desempeño de distintos controladores predictivos, los cuales resuelven el siguiente problema de optimización:

$$\begin{aligned} \min_{u(k)} J(z(k), u(k)) &= Q \sum_{i=1}^{N_p} \left( r(k+i) - y_{fuzzy}(k+i|z(k)) \right)^2 + u^2(k), \\ \text{s. a. } y_{fuzzy}(k+i|z(k)) &= \sum_{r=1}^{N_r} h_r(z(k+i)) \cdot \theta_r^T z(k+i), \\ u(k) &\in [300, 490] [^\circ K], \end{aligned} \quad (4.16)$$

donde  $r(k+i)$  es la referencia en el instante  $k+i$ , en este caso  $r(k+i) = r = 0.5088, \forall i$ ;  $Q > 0$  es un parámetro de sintonización del controlador;  $N_p$  es el horizonte de predicción;  $y_{fuzzy}(k+i|z(k))$  es la predicción a  $i$  pasos realizada por el modelo difuso;  $z(k+i) = [y(k+i-1), \dots, y(k+i-n_y), u(k+i-1), \dots, u(k+i-n_u)]$ ; y  $u(k)$  es la acción de control en el instante  $k$ . Cabe destacar que se asegura que  $y_{fuzzy}(k+1|z(k))$  es una función de  $u(k)$  debido a que  $u(k) \in z(k+1)$ . En este caso se utiliza el algoritmo SQP para resolver el problema de optimización en (4.16) (para más información ver Sección 8.2.1 de anexos).

Dado los resultados obtenidos en la Sección 4.2.1.1, se descartan todos aquellos modelos que poseen condiciones iniciales aleatorias para las variables de optimización, debido a su pobre desempeño como predictor en comparación a los modelos obtenidos al utilizar condiciones iniciales conocidas. Por otro lado, para sintonizar los controladores (determinar el valor de  $Q$ ) se utiliza el procedimiento expuesto en la Sección 8.2.2 de anexos, utilizando la integral del tiempo por el error absoluto (ITAE) como función a optimizar al momento de la sintonización. A su vez, ya que el objetivo de control es maximizar la conversión dentro del reactor, se utiliza el error en estado estacionario,  $e_{ss}$ , como principal métrica de desempeño, donde  $e_{ss} = |r - y_{ss}|$ , con  $y_{ss}$  el valor en estado estacionario de la salida de la planta.



En este caso se considera  $N_p = 1$ , para resolver el mismo problema expuesto en el Capítulo 1 como motivación; y  $N_p = 10$  con el fin de evaluar la respuesta del controlador en un horizonte de predicción distinto.

Los resultados del sistema en lazo cerrado logrados al utilizar modelos de predicción en un esquema MPC obtenidos a partir de datos en lazo abierto se resumen en la Tabla 4.10, mientras que la Tabla 4.11 muestra los resultados al utilizar modelos de predicción obtenidos a partir de datos en lazo cerrado. En ambos casos se cuantifica el esfuerzo de control asociado a la prueba realizada,  $u^T(k)u(k)$ .

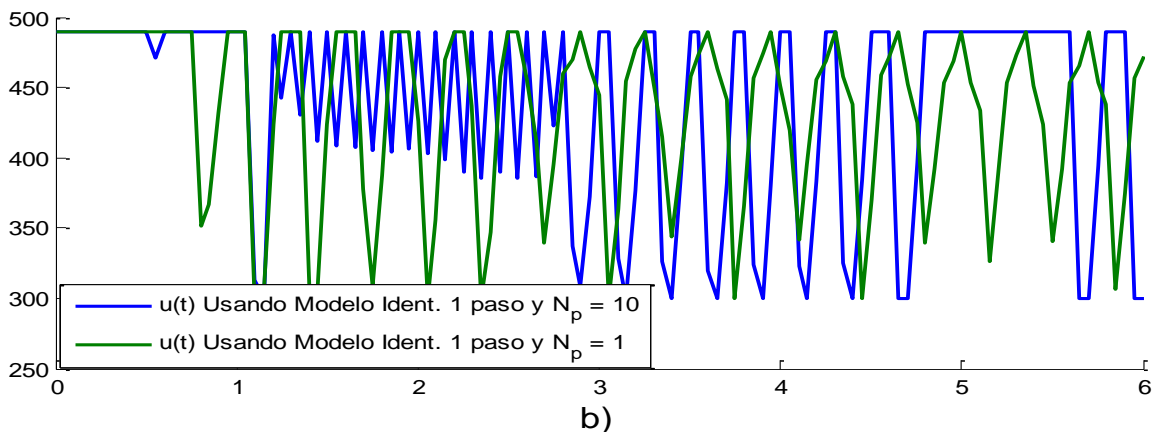
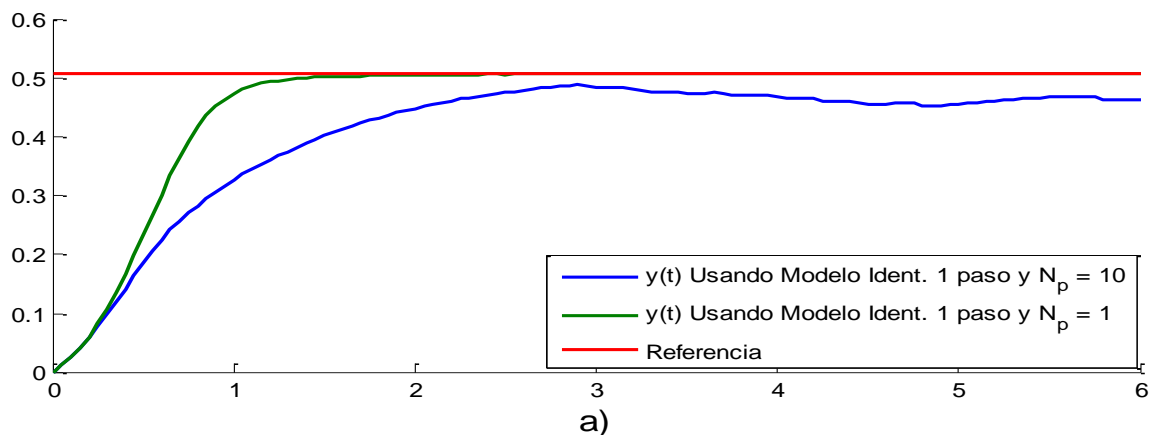
Como se observa en la Tabla 4.10, los modelos de predicción obtenidos en lazo abierto logran generar buenos resultados cuando se tiene un horizonte de predicción a un paso, alcanzando errores en estado estacionario menores al 1%, sin embargo, al aumentar el valor de  $N_p$  su desempeño decae drásticamente, alcanzando errores sobre el 5%. Lo anterior se debe a que los modelos obtenidos poseen un comportamiento inestable, afectando su predicción en horizontes de predicción mayores. A su vez, dichos modelos son entrenados utilizando datos que difieren del comportamiento en lazo cerrado del sistema, por lo que en horizontes mayores el modelo pierde representatividad, afectando directamente al desempeño del controlador. Note también que el error en estado estacionario al utilizar  $N_p = 10$  es mayor al emplear el modelo de predicción obtenido por identificación a un paso, debido a que dicho modelo tiene un peor desempeño en horizontes de predicción mayores, como se observa en la Tabla 4.2. A su vez, el esfuerzo de control aumenta en todos los escenarios estudiados al utilizar un horizonte de predicción  $N_p = 10$ , debido a que los controladores son sintonizados con el objetivo de minimizar el error en estado estacionario, dando menor peso a la componente asociada a  $u(k)$  en la función objetivo del control predictivo.

**Tabla 4.10: Resultados Control Predictivo Difuso del CSTR con Modelos de Predicción Obtenidos con Datos en Lazo Abierto.**

<i>Modelo de Predicción</i>			<b>Control Predictivo Difuso</b>					
<b>Ident.</b>	<b>Solver</b>	$\begin{bmatrix} n_y^* \\ n_u^* \\ \text{reglas}^* \end{bmatrix}$	$Q$		$e_{ss}$		$u^T(k)u(k)$	
			$N_p = 1$	$N_p = 10$	$N_p = 1$	$N_p = 10$	$N_p = 1$	$N_p = 10$
1 Paso	NM/ QN/ PSO	[5,4,4]	$10^{12}$	$10^{12}$	<b>6.9797 e-04 (0.1371%)</b>	0.0432 (8.5004%)	2.3367 e+07	2.3814 e+07
$N$ pasos	QN/ PSO	[5,4,3]	$10^{12}$	$10^{12}$	4.6325 e-03 (0.9105%)	<b>0.0279 (5.5028%)</b>	2.3173 e+07	2.3554 e+07
	NM	[5,4,3]	$10^{12}$	$10^{12}$	4.6379 e-03 (0.9115%)	0.028 (5.5148%)	2.3172 e+07	2.3553 e+07

Cabe destacar que los modelos de predicción obtenidos por identificación a  $N$  pasos poseen la misma estructura, y sólo difieren levemente en el valor de sus parámetros, por lo que se justifica que dichos modelos generen un desempeño en lazo cerrado similar, obteniendo errores en estado estacionario parecidos.

En la Figura 4.17 a) se observa la respuesta del control predictivo difuso al utilizar el modelo de predicción obtenido con identificación a un paso, donde se ve que al utilizar  $N_p = 1$  el reactor logra alcanzar la referencia deseada, mientras que al emplear  $N_p = 10$  su respuesta empeora, comenzando a oscilar en un margen inferior al esperado debido a las malas predicciones del modelo en horizontes mayores. La Figura 4.17 b) presenta la acción de control asociada a ambas pruebas. En particular, al emplear  $N_p = 1$  se tienen entradas más drásticas, alternando constantemente entre el valor máximo y mínimo del flujo de entrada  $u(t)$  con el fin de minimizar la función objetivo en cada iteración. A su vez, al utilizar  $N_p = 10$  se mantiene el efecto oscilatorio de la entrada, y producto de la baja ponderación del esfuerzo de control con respecto al error de predicción en la función objetivo se obtienen respuestas con mayor energía.



**Figura 4.17: Resultados Control Predictivo Difuso del CSTR Usando Modelos de Predicción Obtenidos con Datos en Lazo Abierto e Identificación a un Paso. a) Salida  $y(t)$  vs Tiempo [s]. b) Entrada  $u(t)$  [K] vs Tiempo [s].**

Cabe recordar que en este caso no se busca minimizar la variación de la acción de control  $\Delta u(t)$ , sino que se busca disminuir la acción de control misma  $u(t)$ . Luego, el controlador tiende al menor valor posible (en este caso  $u_{min} = 300 [K]$ ) cada vez que los errores inducidos por el modelo de predicción lo permitan, respetando la relación entre dichos errores y el valor de la acción de control descrita en la función objetivo del problema de optimización del controlador.

Similar al caso presentado anteriormente, los resultados del controlador predictivo difuso al utilizar modelos de predicción obtenidos con datos en lazo cerrado no difieren mucho uno de otro en los casos que se tiene la misma estructura, como se observa en la Tabla 4.11. Por otro lado, a diferencia de emplear modelos de predicción obtenidos con datos en lazo abierto, utilizar modelos obtenidos por identificación en lazo cerrado genera resultados sobresalientes en todos los escenarios estudiados, logrando errores en estado estacionario menores al 1%.

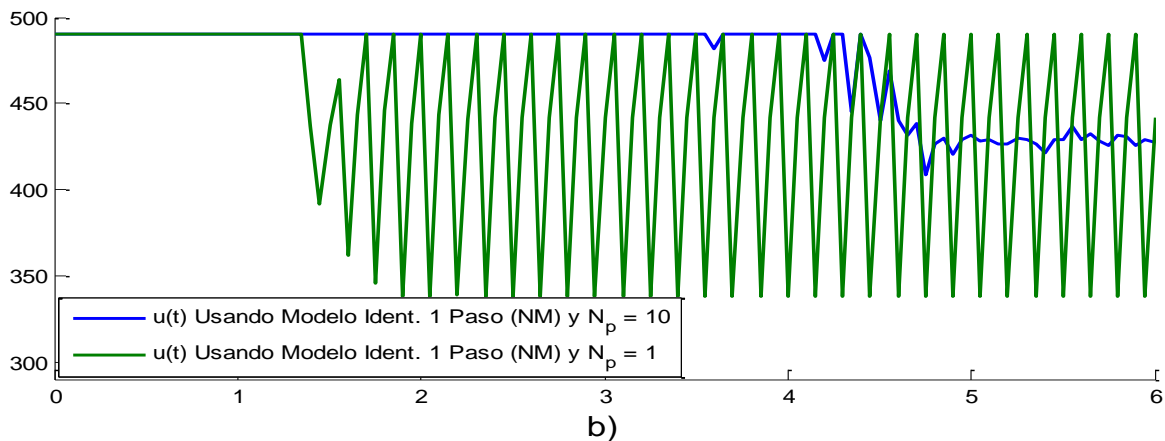
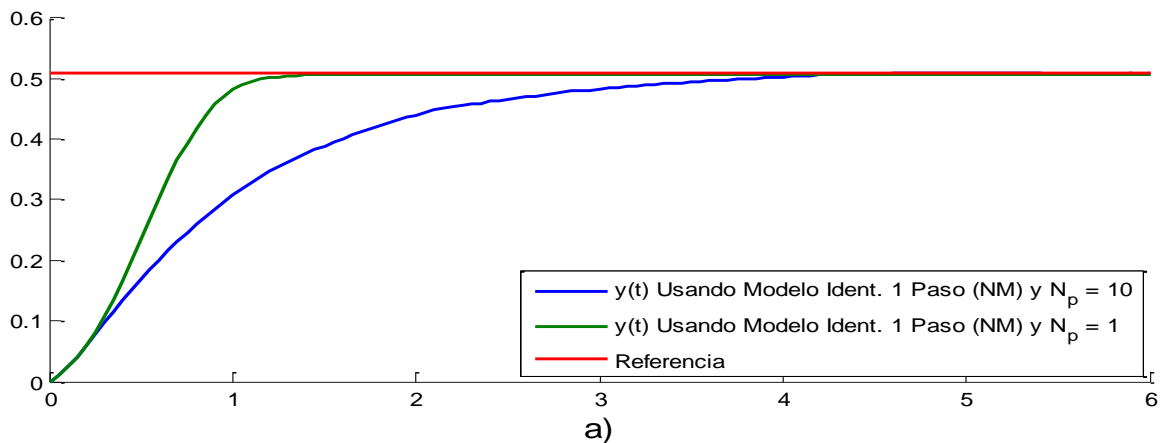
**Tabla 4.11: Resultados Control Predictivo Difuso del CSTR con Modelos de Predicción Obtenidos con Datos en Lazo Cerrado.**

<b>Modelo de Predicción</b>			<b>Control Predictivo Difuso</b>					
<b>Ident.</b>	<b>Solver</b>	$\begin{matrix} n_y^* \\ n_u^* \\ \text{reglas}^* \end{matrix}$	<b>Q</b>		<b><math>e_{ss}</math></b>		<b><math>u^T(k)u(k)</math></b>	
			$N_P = 1$	$N_P = 10$	$N_P = 1$	$N_P = 10$	$N_P = 1$	$N_P = 10$
1 Paso	QN/ PSO	[4,4,4]	2.6308 e+10	2.6698 e+10	2.2587 e-03 (0.4439%)	8.8881 e-04 (0.1746%)	2.3674 e+07	2.5965 e+07
	NM	[4,3,5]	4 e+10	4 e+10	<b>1.7584 e-03 (0.3456%)</b>	<b>7.1008 e-04 (0.1395%)</b>	2.3743 e+07	2.7271 e+07
N pasos	QN/ PSO	[4,3,5]	4 e+10	4 e+10	1.7606 e-03 (0.3460%)	7.1095 e-04 (0.1397%)	2.3757 e+07	2.7364 e+07
	NM	[4,3,5]	4 e+10	4 e+10	1.7586 e-03 (0.3456%)	7.1305 e-04 (0.1401%)	2.3760 e+07	2.7214 e+07

En este caso el modelo que mejor desempeño tiene en lazo cerrado es el obtenido por identificación a un paso utilizando NM como *solver*, ya sea para  $N_p = 1$  como para  $N_p = 10$ , sin embargo, las diferencias en rendimiento son mínimas entre todos los modelos de predicción estudiados. Cabe recordar que el modelo en cuestión es el modelo óptimo obtenido por identificación a un paso, y su desempeño como predictor en horizontes mayores supera mínimamente a los obtenidos por identificación a  $N$  pasos, como se presenta en la Tabla 4.6 y en la Tabla 4.7. Luego, dicha pequeña mejora en el rendimiento en lazo cerrado se debe a las mejores predicciones que otorga dicho modelo. Por otro lado, con respecto al esfuerzo de control se presenta el mismo comportamiento

obtenido al utilizar modelos identificados con datos en lazo abierto, generando valores mayores al aumentar el horizonte de predicción debido a la ponderación en la función objetivo del controlador.

En la Figura 4.18 a) se observa la respuesta de la planta al utilizar el controlador predictivo difuso con el modelo de predicción obtenido por identificación a un paso, utilizando NM como *solver*, mientras que la Figura 4.18 b) presenta la acción de control asociada. Se aprecia que la respuesta del sistema es más lenta al utilizar horizontes de predicción mayores, obteniendo una respuesta más suave, logrando converger a la referencia de mejor manera. Por otro lado, la acción de control al utilizar  $N_p = 1$  presenta nuevamente cambios drásticos una vez que el error de la planta es pequeño, debido a que en dicho escenario el término asociado a  $u(k)$  en la función objetivo toma mayor peso en comparación a cuando se tiene  $N_p = 10$ , y por lo tanto el algoritmo de optimización entrega soluciones con  $u(k)$  menores.



**Figura 4.18: Resultados Control Predictivo Difuso del CSTR Usando Modelos de Predicción Obtenidos con Datos en Lazo Cerrado e Identificación a un Paso (NM). a) Salida  $y(t)$  vs Tiempo [s]. b) Entrada  $u(t)$  [K] vs Tiempo [s].**

Luego, dados los resultados expuestos en la Tabla 4.10 y en la Tabla 4.11, generar modelos a partir de datos en lazo cerrado para el diseño de estrategias de control basada en modelos permite obtener un mejor desempeño en lazo cerrado en miras de alcanzar objetivos de control, en comparación a utilizar modelos obtenidos por identificación en lazo abierto, debido a que los primeros identifican de mejor manera las frecuencias de interés del comportamiento en lazo cerrado del sistema. A su vez, se destaca el hecho de que los modelos utilizados para la predicción del comportamiento de la planta son controlables, y gracias a dicha propiedad es posible inferir adecuadamente acciones de control que permiten alcanzar la referencia deseada, en contraste a los resultados obtenidos al utilizar modelos no controlables, como se presenta en el Capítulo 1 como motivación.

Cabe destacar que el controlador predictivo difuso diseñado no es una estrategia que asegure error en estado estacionario nulo, debido a que no incluye una acción integral que permita que  $e_{ss} \rightarrow 0$ , sino que busca equilibrar el error de seguimiento y el esfuerzo de control con el fin de obtener una estrategia óptima. Sin embargo, se destaca el hecho de que los errores en estado estacionario obtenidos por los controladores implementados son menores al 1%, siendo éste considerado como un buen desempeño para el caso de estudio abordado.

#### 4.2.2. Sistema no Lineal Teóricamente Controlable

Como segunda aplicación de la metodología propuesta se plantea una clase de sistemas no lineales dada por la ecuación de estado

$$\begin{aligned} \dot{x}_1(t) &= ax_2(t) + f_1(x_1(t)) \\ \dot{x}_2(t) &= f_2(x_1(t), x_2(t)) + u(t), \quad a \neq 0, \\ y(t) &= h(x(t)) = x_1(t) \end{aligned} \quad (4.17)$$

donde  $f_1(x_1(t))$  y  $f_2(x_1(t), x_2(t))$  son dos funciones no lineales diferenciables cualquiera,  $x(t) = [x_1(t), x_2(t)]^T$  es el estado,  $y(t)$  es la salida y  $a$  es una constante no nula. Notar que (4.17) corresponde a un sistema no lineal afín de la forma  $\dot{x}(t) = f(x(t)) + g(x(t)) \cdot u(t)$ , donde  $f(x(t)) = [ax_2(t) + f_1(x_1(t)), f_2(x_1(t), x_2(t))]^T$  y  $g(x(t)) = [0, 1]^T$ .

Aplicando las condiciones de accesibilidad de sistemas no lineales afines se obtiene que la matriz de accesibilidad,  $\mathbb{A}(x(t))$ , del sistema en (4.17) es

$$\mathbb{A}(x(t)) = \begin{bmatrix} 0 & -a \\ 1 & -\frac{df_2(x_1(t), x_2(t))}{dx_2} \end{bmatrix}. \quad (4.18)$$

Notar que  $\det(A(x(t))) = a \neq 0, \forall x(t)$ ; por lo tanto  $A(x(t))$  es de rango completo y en consecuencia el sistema es accesible para todo  $x(t)$ , lo cual hace al sistema completamente controlable.

Por otro lado, también se pueden derivar condiciones de estabilidad del sistema en (4.17). Para ello se emplea el enfoque clásico de Lyapunov, el cual determina la estabilidad de sistemas no lineales a partir del siguiente teorema.

Sea  $x_e = 0$  un punto de equilibrio de (4.17) en una región  $D \subset \mathbb{R}^n$  que contiene a  $x_e$ . Si existe una función continua diferenciable  $V: D \rightarrow \mathbb{R}$  tal que

$$V(0) = 0 ; V(x) > 0 \text{ en } D \setminus \{0\} ; \dot{V}(x) \leq 0 \text{ en } D, \quad (4.19)$$

entonces  $x_e = 0$  es estable [49]. Además:

- Si  $\dot{V}(x) < 0$  en  $D \setminus \{0\}$ , entonces  $x_e = 0$  es asintóticamente estable.
- Si  $\dot{V}(x) < -aV(x)$ , para algún  $a > 0$ , entonces el equilibrio es exponencialmente estable.
- Si las condiciones en (4.19) se cumplen para  $D = \mathbb{R}^n$  entonces  $x_e$  es (asintóticamente/exponencialmente) estable globalmente.

Luego, por simplicidad se impone que  $f_1(0) = f_2(0,0) = 0$  tal que  $x_e = 0$  sea punto de equilibrio. Sea  $V(x(t)) = \frac{1}{2}x(t)^T x(t)$ , entonces se tiene que cumplir que

$$\begin{aligned} \dot{V}(x(t)) &= x_1(t)\dot{x}_1(t) + x_2(t)\dot{x}_2(t) \\ &= ax_1(t)x_2(t) + x_1(t)f_1(x_1(t)) + x_2(t)f_2(x_1(t), x_2(t)) \leq 0, \end{aligned} \quad (4.20)$$

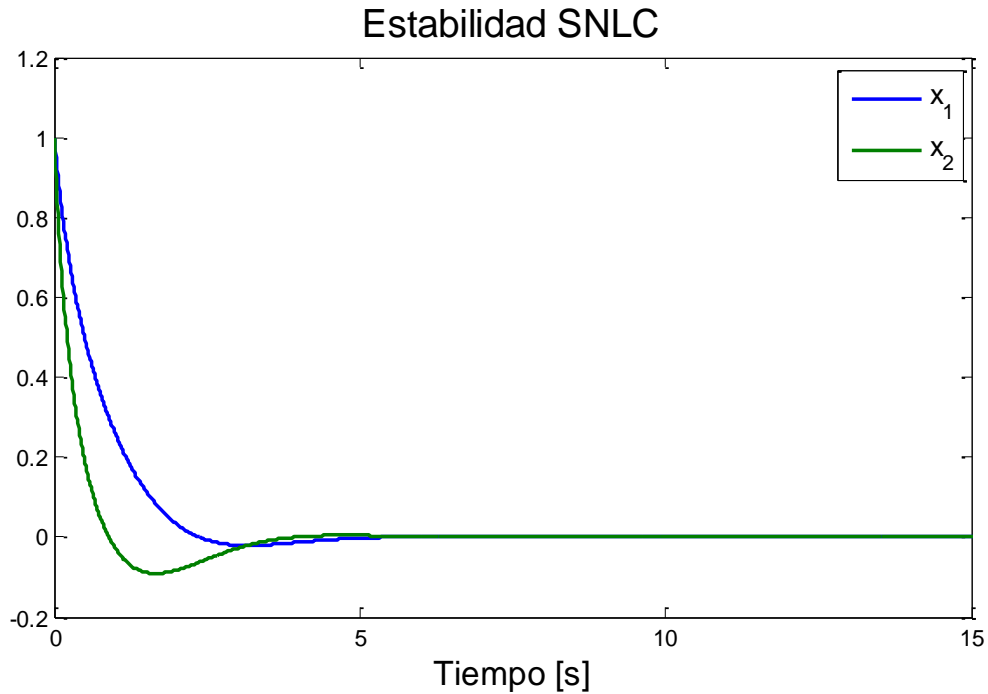
de donde se impone que

$$\begin{aligned} x_1(t)f_1(x_1(t)) &\leq 0 \\ ax_1(t)x_2(t) + x_2(t)f_2(x_1(t), x_2(t)) &\leq 0. \end{aligned} \quad (4.21)$$

Dado lo anterior, se considera el siguiente sistema no lineal controlable (SNLC):

$$\begin{aligned} \dot{x}_1(t) &= x_2(t) - x_1(t) \exp(x_1(t)) \\ \dot{x}_2(t) &= -x_1(t) - x_2(t) \exp(x_2(t)) + u(t) \\ y(t) &= h(x(t)) = x_1(t), \end{aligned} \quad (4.22)$$

el cual cumple con las condiciones de estabilidad presentes en (4.21). Se comprueba la estabilidad del modelo mediante simulación no forzada del sistema utilizando como condiciones iniciales  $x_1(0) = x_2(0) = 1$ . La respuesta se presenta en la Figura 4.19, donde se puede observar que ambos estados convergen al origen.



**Figura 4.19: Simulación no Forzada SNLC.**

Una vez analizada la planta a tratar se continúa con el procedimiento de identificación.

#### **4.2.2.1. Identificación del SNLC**

Similar a lo realizado en la Sección 4.2.1.1, se analiza la respuesta al escalón del sistema en estudio para determinar el tiempo de respuesta de la planta para el diseño de la señal de excitación. En la Figura 4.20 se observa dicho comportamiento frente a distintas condiciones iniciales. Del gráfico se observa que el tiempo de estabilización de la planta es cercano a los 2 [s], mientras que superado los 3 [s] la planta está completamente estable. Luego se considera  $f_{max} = 1$  [Hz] para capturar información en estado transitorio, y  $f_{min} = f_{max}/5$  [Hz] para capturar datos en estado permanente.

### Respuesta al Escalón SNLC

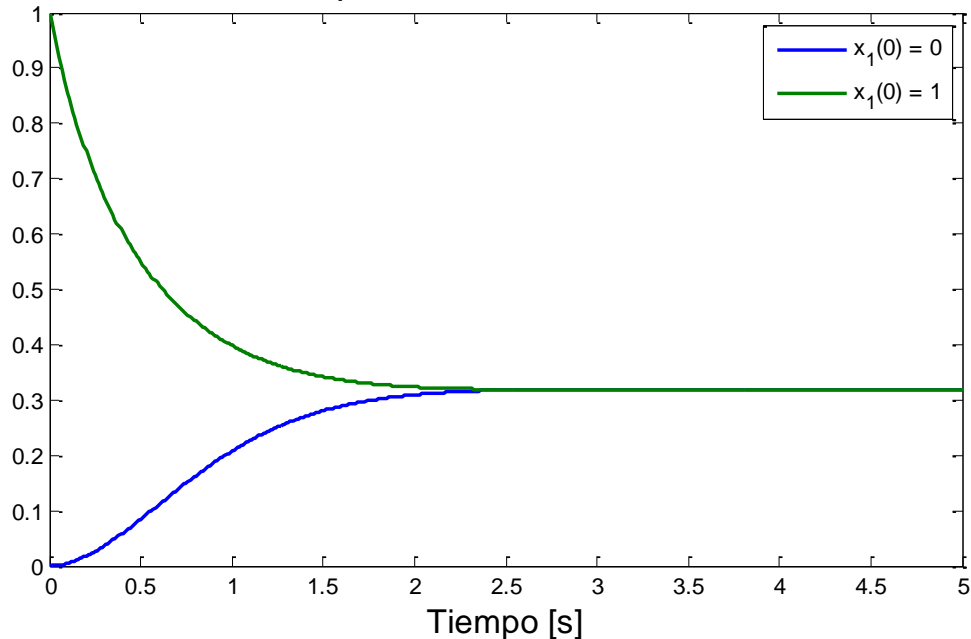


Figura 4.20: Respuesta al Escalón SNLC.

El tiempo de muestreo a considerar es de  $T_s = 0.01$  [s]. Con estos datos se genera una PRBS de orden  $n = 4$ , y en consecuencia un largo máximo  $N = 15$ . La cantidad de muestras por bit son  $N_s = 40$  y  $\Delta t = 0.4$  [s], lo que da una duración total de 6 [s]. Similar al caso del CSTR, se generan varias PRBS con los mismos parámetros para formar una prueba de identificación de mayor duración.

Dados los resultados obtenidos en la Sección 4.2.1, y como el objetivo de la identificación difusa es el diseño de una estrategia de control predictivo, se limita el estudio del proceso actual sólo a identificación en lazo cerrado. Además, como a priori no se tienen restricciones sobre la variable de salida  $y(t)$ , sin pérdida de generalidad se opta por trabajar en el rango  $[0,1]$ .

Para obtener datos en lazo cerrado se diseña un controlador por linealización exacta, por lo que es necesario encontrar la transformación de estado que permita linealizar el sistema [98]. En el caso general, se considera el modelo en (4.17) para describir el diseño del controlador a continuación.

Primero se estudia el grado relativo del sistema. Para ello se define la derivada de Lie como

$$L_f h(x(t)) = \left\langle \frac{dh(x(t))}{dx(t)}, f(x(t)) \right\rangle, \quad (4.23)$$



donde  $\langle \cdot, \cdot \rangle$  denota el producto punto. Además  $L_f L_g h(x(t)) = \left\langle \frac{dL_g h(x(t))}{dx(t)}, f(x(t)) \right\rangle$  y  $L_f^2 h(x(t)) = \left\langle \frac{dL_f h(x(t))}{dx(t)}, f(x(t)) \right\rangle$ .

Un sistema tiene grado relativo  $r$  en  $x_0$  si:

- $L_g L_f^k h(x(t)) = 0, \forall x(t)$  en una vecindad de  $x_0, \forall k < r - 1$
- $L_g L_f^{r-1} h(x_0) \neq 0$

Luego, para (4.17) se tiene que

$$\begin{aligned} L_g h(x(t)) &= 0, & L_f h(x(t)) &= ax_2(t) + f_1(x_1(t)) \\ L_g L_f h(x(t)) &= a & & \\ L_f^2 h(x(t)) &= (ax_2(t) + f_1(x_1(t))) \cdot \frac{df_1(x_1(t))}{dx_1(t)} + af_2(x_1(t), x_2(t)), \end{aligned} \quad (4.24)$$

donde se observa que (4.17) tiene grado relativo  $r = 2, \forall x(t)$ .

Luego, se plantea la transformación de estado

$$\left. \begin{aligned} z_1(t) &= x_1(t) \\ z_2(t) &= L_f h(x(t)) \end{aligned} \right\} \rightarrow \begin{aligned} \dot{z}_1(t) &= z_2(t) \\ \dot{z}_2(t) &= L_f^2 h(x(t)) + L_g L_f h(x(t))u(t). \end{aligned} \quad (4.25)$$

Si se define la acción de control

$$u(t) = \frac{1}{L_g L_f h(x(t))} \left( -L_f^2 h(x(t)) + v(t) \right), \quad (4.26)$$

es posible linealizar el sistema en las variables de estado  $z(t) = [z_1(t), z_2(t)]^T$ , resultando

$$\dot{z}(t) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} z(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} v(t). \quad (4.27)$$

Una vez linealizado el sistema se puede manipular  $v(t)$  tal que (4.27) se establezca. Para esto se reubican los polos del sistema definiendo  $v(t) = -c_0 z_1(t) - c_1 z_2(t)$ , con lo que el sistema queda

$$\dot{z}(t) = \begin{bmatrix} 0 & 1 \\ -c_0 & -c_1 \end{bmatrix} z(t). \quad (4.28)$$

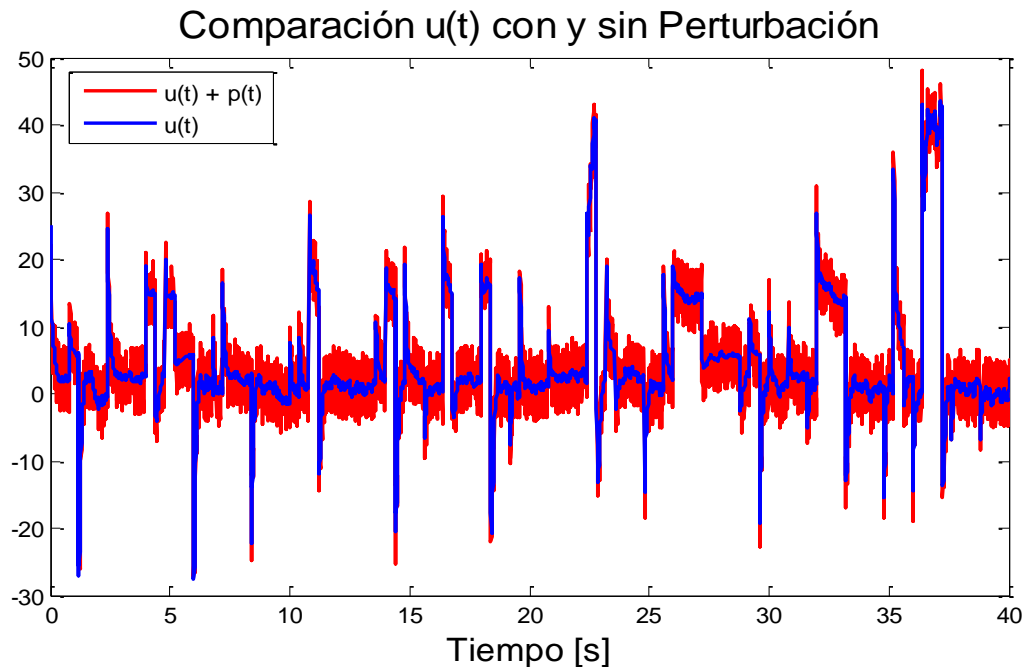
Igualando los polos de (4.28) a un par de polos deseados  $(q_1, q_2)$  se obtienen las siguientes relaciones:

$$\begin{aligned} c_1 &= -(q_1 + q_2) \\ c_0 &= q_1 q_2 \end{aligned} \quad (4.29)$$

Finalmente, para que el sistema siga una referencia deseada  $r(t)$  se debe emplear el controlador

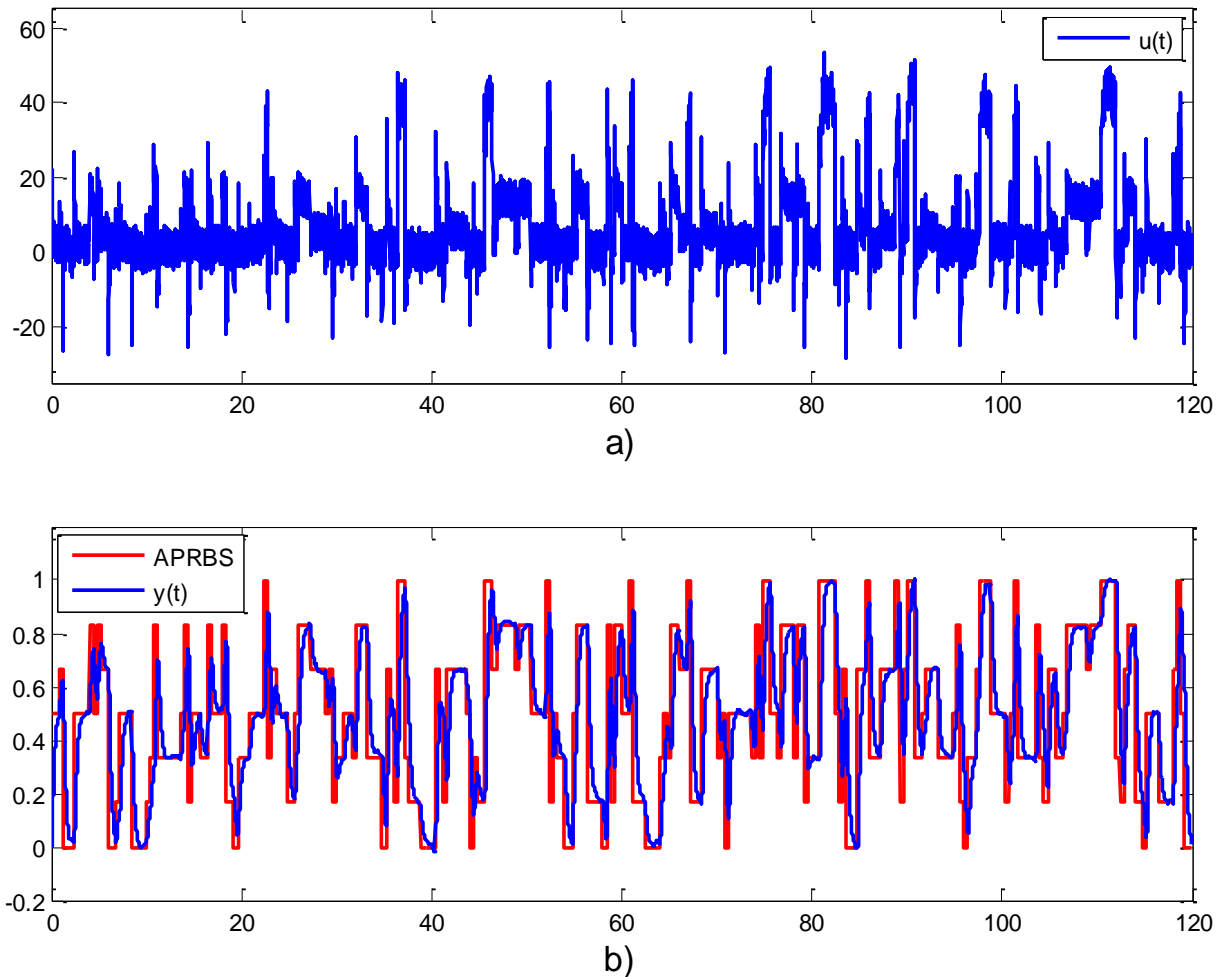
$$u(t) = \frac{1}{L_g L_f h(x(t))} \left( -L_f^2 h(x(t)) + \ddot{r}(t) - c_0 (h(x(t)) - r(t)) - c_1 (L_f h(x(t)) - \dot{r}(t)) \right) \quad (4.30)$$

Una vez diseñado el controlador se procede a realizar la prueba de identificación utilizando el esquema en lazo cerrado ilustrado en la Sección 2.2 (Figura 2.6). Se considera  $p(t)$  como una señal uniformemente distribuida en el rango  $[-5,5]$ . Dicho rango es determinado a partir de la respuesta de la planta, con el fin de que se observe cierto grado de variación en  $u(t)$ . En la Figura 4.21 se puede observar parte de los datos de la variable manipulada con y sin perturbación.



**Figura 4.21:** Comparación Señal de Control  $u(t)$  con y sin Perturbación  $p(t)$ .

Los polos deseados del sistema linealizado son ubicados, sin pérdida de generalidad, en  $-5$  y  $-10$ , por lo que  $c_0 = 50$  y  $c_1 = 15$ . Los datos generados para la identificación en lazo cerrado se pueden observar en la Figura 4.22.



**Figura 4.22: Datos de Identificación del SNLC en Lazo Cerrado. a) Entrada  $u(t)$  vs Tiempo [s]. b) APRBS y Datos de Salida  $y(t)$  vs Tiempo [s].**

En la Figura 4.23 se observa la función de autocorrelación parcial de los datos de salida del SNLC, donde se ve que se tiene información relevante hasta el quinto retardo, por lo que nuevamente se tiene que  $n_{y,max} = 5$ . A su vez, se vuelve a considerar  $\lambda = 10^{10}$  y  $reglas_{max} = 5$ . Debido a los resultados obtenidos en la Sección 4.2.1, se descarta desde el inicio el uso de condiciones iniciales aleatorias para los algoritmos de optimización a utilizar.

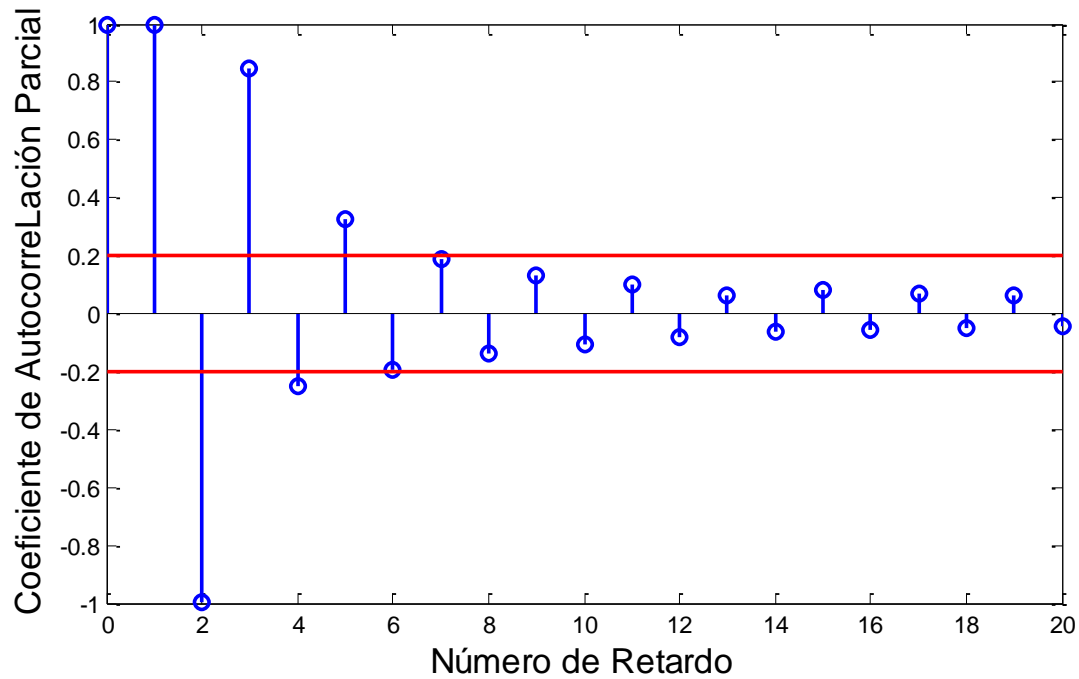


Figura 4.23: Función de Autocorrelación Parcial SNLC.

Los resultados obtenidos del proceso de identificación se resumen en la Tabla 4.12 y Tabla 4.13, para el caso de identificación a un paso y  $N$  pasos respectivamente. Para las pruebas utilizando el algoritmo PSO se emplean los mismos parámetros que en las pruebas de identificación del CSTR (ver Tabla 4.1), salvo que en este caso se limita el espacio de búsqueda a parámetros en el rango  $[-2.5, 2.5]$ , debido a que la solución óptima sin considerar restricciones de controlabilidad se obtiene dentro de dicho rango.

Tabla 4.12: Resultados Identificación del SNLC en Lazo Cerrado a un Paso.

<i>Solver</i>	RMSE Entrenamiento		RMSE Validación		RMSE Prueba		$[n_y^*, n_u^*, reglas^*]$
	1 Paso	$N$ Pasos	1 Paso	$N$ Pasos	1 Paso	$N$ Pasos	
	QN	1.4126 e-05	0.3156	<b>1.6775</b> <b>e-05</b>	0.4921	1.3559 e-05	
NM	1.4126 e-05	0.3156	<b>1.6775</b> <b>e-05</b>	0.4921	1.3559 e-05	1.4974	[5,5,5]
PSO	1.4126 e-05	0.3156	<b>1.6775</b> <b>e-05</b>	0.4921	1.3559 e-05	1.4974	[5,5,5]

**Tabla 4.13: Resultados Identificación del SNLC en Lazo Cerrado a  $N$  Pasos.**

<i>Solver</i>	RMSE Entrenamiento		RMSE Validación		RMSE Prueba		$[n_y^*, n_w^*, reglas^*]$
	1	$N$	1	$N$	1	$N$	
	Paso	Pasos	Paso	Pasos	Paso	Pasos	
QN	2.3333 e-04	0.0417	2.9443 e-04	<b>0.0503</b>	2.6353 e-04	0.0468	[2,2,4]
NM	2.3539 e-04	0.0419	3.0105 e-04	0.0515	2.7565 e-04	0.0492	[2,2,4]
PSO	9.3804 e-05	0.0507	1.1019 e-04	0.0630	9.1990 e-05	0.0492	[5,2,4]

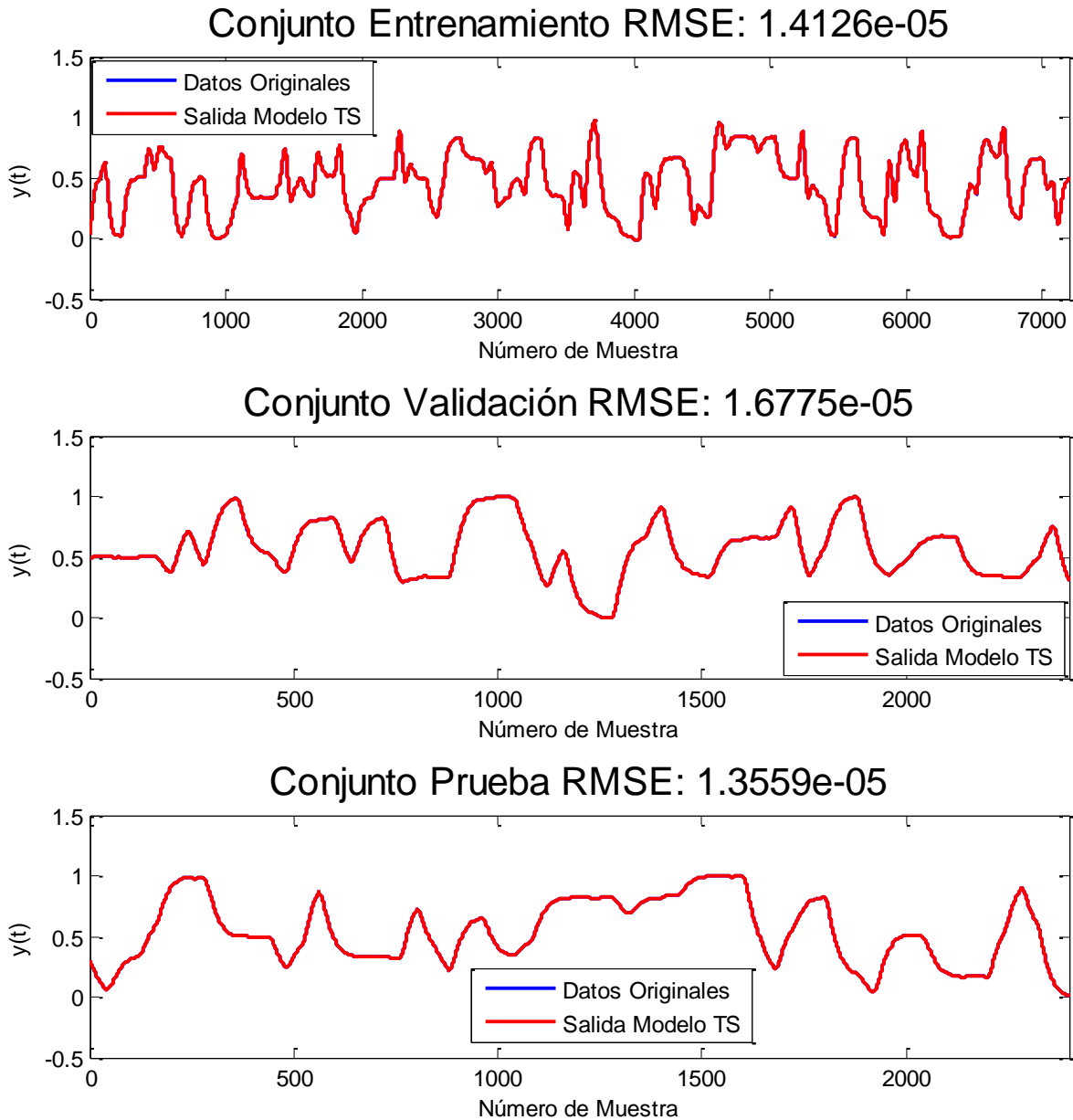
Similar a los resultados obtenidos en la Sección 4.2.1.1, los algoritmos de optimización utilizados logran converger a la región factible, encontrando parámetros de los consecuentes del modelo difuso que permiten que éste sea controlable en los diferentes puntos de operación, según las restricciones impuestas. A su vez, varias de las estructuras estudiadas no poseen parámetros tal que el modelo difuso sea controlable, por lo que son descartadas como posibles soluciones.

En cuanto a la identificación a un paso, todos los algoritmos de optimización utilizados convergen a la misma estructura óptima, la cual posee un bajo índice de error en cuanto a la predicción a un paso, mas no así para un horizonte de predicción mayor, donde los errores se acumulan constantemente, malogrando el rendimiento del modelo. Por otro lado, y similar a los resultados obtenidos en la Sección 4.2.1.1, las pruebas de identificación a  $N$  pasos retornan modelos con buenos desempeños para predicciones tanto a uno como a  $N$  pasos, demostrando nuevamente las ventajas de emplear horizontes de predicción mayores por sobre la identificación a un paso usualmente utilizada en la literatura.

En la Tabla 4.13 se observa que en la identificación a  $N$  pasos ninguno de los *solvers* empleados converge exactamente al mismo modelo de predicción, obteniéndose en cada caso índices de error diferentes. Se aprecia que en este caso el algoritmo con mejor desempeño es Quasi-Newton, mientras que PSO obtiene el peor resultado, convergiendo a una solución subóptima, contrario a lo obtenido en el CSTR, donde ambos algoritmos frecuentemente convergen a la misma solución, y donde también por lo general Nelder-Mead obtiene los mejores resultados, demostrando así que el mejor modelo puede ser obtenido por cualquiera de los algoritmos de optimización empleados, dependiendo esto de la aplicación que se aborde, debido a que el problema de optimización a resolver es no convexo.

Con respecto al orden de los modelos obtenidos, se tiene que el mejor modelo de predicción obtenido por identificación a  $N$  pasos es de segundo orden, igual que la planta original, lo que indica que no necesariamente un modelo de mayor orden es mejor, en cuyo caso es más probable que se generen sobreajustes del modelo, obteniendo peores desempeños al ser puestos a prueba utilizando distintos datos.

En la Figura 4.24 se presenta el desempeño del modelo difuso obtenido por identificación a un paso, donde se observa que éste es capaz de predecir certeramente el comportamiento de la planta original cuando posee información del instante anterior, lo cual refleja un buen diseño del experimento de identificación para la obtención de datos de entrada-salida que permitan identificar un modelo de calidad.



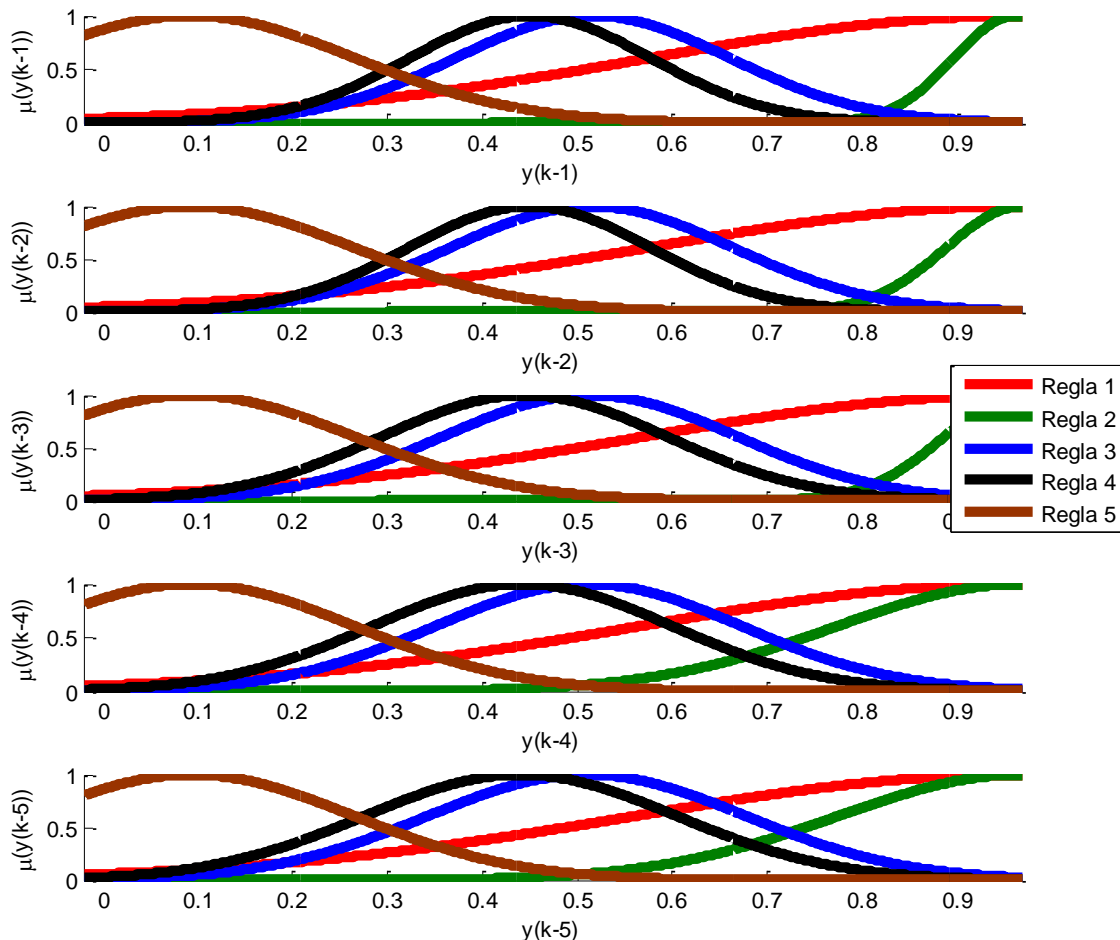
**Figura 4.24: Resultado Mejor Modelo Identificación a un Paso del SNLC. Arriba: Conjunto de Entrenamiento. Medio: Conjunto de Validación. Abajo: Conjunto de Prueba.**

El modelo difuso obtenido por identificación a un paso posee la estructura resumida en la Tabla 4.14.

**Tabla 4.14: Estructura Mejor Modelo Identificación a un Paso del SNLC con Datos en Lazo Cerrado.**

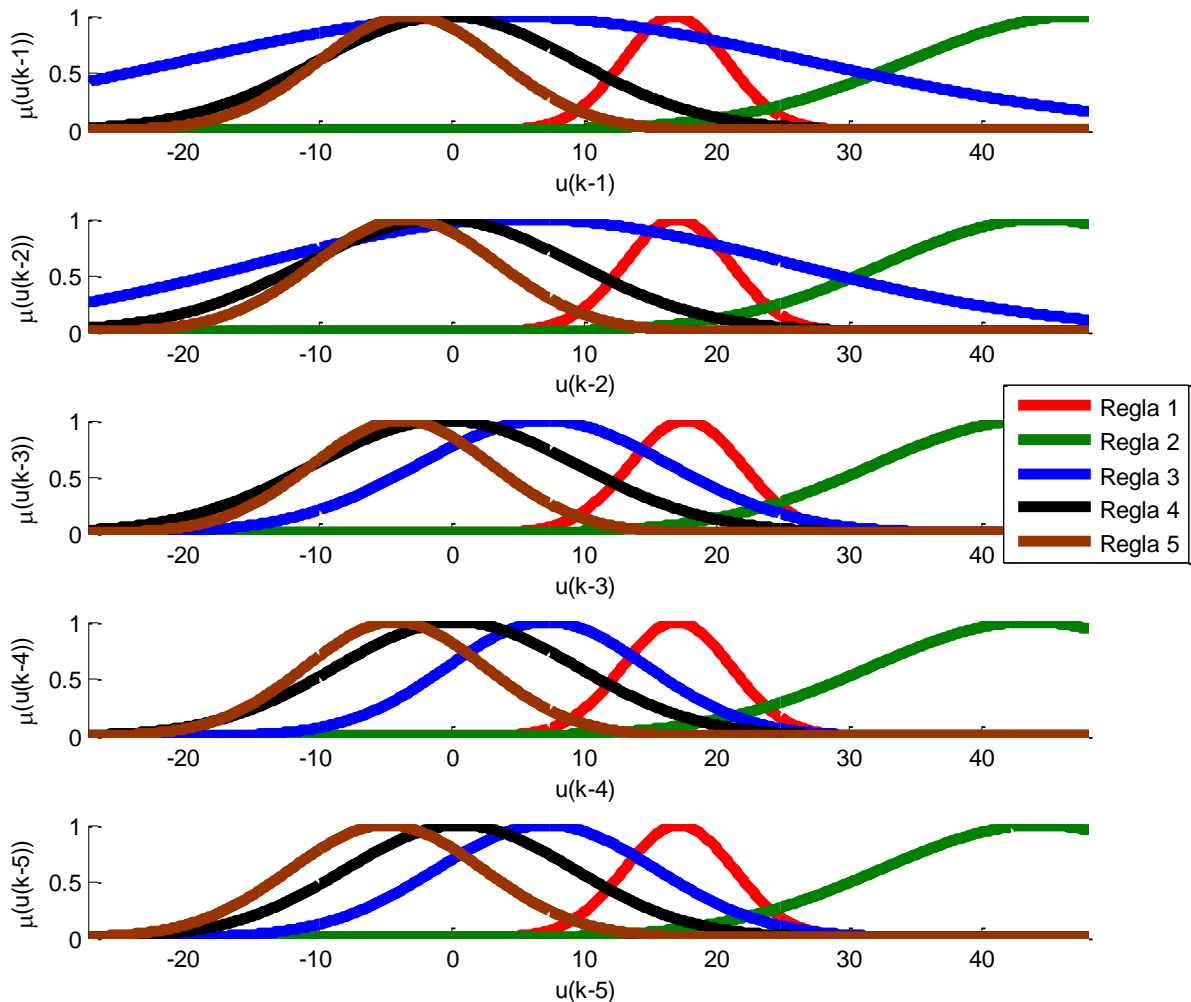
<b>Premisas</b>	$y(k - 1), \dots, y(k - 5), u(k - 1), \dots, u(k - 5)$		
<b>Cantidad de Reglas</b>	5		
<b>Parámetros Premisas</b>	$R_1$	$a_{1,i}$	[2.5633, 2.5417, 2.5170, 2.4894, 2.4595, 0.2549, 0.2507, 0.2353, 0.2431, 0.2410]
		$b_{1,i}$	[0.9681, 0.9681, 0.9681, 0.9681, 0.9681, 16.9115, 17.1085, 17.7673, 17.0596, 17.2465]
	$R_2$	$a_{2,i}$	[15.4982, 12.7222, 12.4118, 5.1659, 5.1952, 0.0777, 0.0848, 0.0853, 0.0827, 0.0832]
		$b_{2,i}$	[0.9622, 0.9681, 0.9681, 0.9681, 0.9681, 47.3879, 44.6109, 43.6904, 43.8306, 44.4822]
	$R_3$	$a_{3,i}$	[6.9273, 6.6918, 6.4282, 6.1885, 5.9580, 0.0424, 0.0499, 0.1049, 0.1323, 0.1203]
		$b_{3,i}$	[0.5171, 0.5158, 0.5143, 0.5126, 0.5108, 3.4725, 5.5721, 7.0601, 7.3344, 7.2510]
	$R_4$	$a_{4,i}$	[7.8914, 7.8197, 6.6452, 6.3252, 6.0701, 0.1026, 0.0983, 0.1003, 0.1100, 0.1159]
		$b_{4,i}$	[0.4513, 0.4505, 0.4451, 0.4430, 0.4412, -0.2614, -0.7529, -0.3682, 0.4899, 0.7471]
	$R_5$	$a_{5,i}$	[5.7411, 5.7637, 5.7876, 5.8128, 5.8358, 0.1471, 0.1446, 0.1473, 0.1476, 0.1377]
		$b_{5,i}$	[0.0922, 0.0929, 0.0935, 0.0941, 0.0947, -3.0988, -3.3190, -3.6522, -4.1595, -4.7840]
<b>Parámetros Consecuencias</b>	$R_1$	$\theta_1^T = [3.6813, -5.1478, 3.3097, -0.9024, 0.0593, 4.6108 \cdot 10^{-5}, -4.5453 \cdot 10^{-5}, -3.5537 \cdot 10^{-5}, 3.8412 \cdot 10^{-5}, -3.1665 \cdot 10^{-6}]$	
	$R_2$	$\theta_2^T = [4.2470, -7.1315, 5.9096, -2.4147, 0.3895, 4.2535 \cdot 10^{-5}, -7.5189 \cdot 10^{-5}, 5.7224 \cdot 10^{-6}, 5.2937 \cdot 10^{-5}, -2.4263 \cdot 10^{-5}]$	
	$R_3$	$\theta_3^T = [3.8327, -5.7307, 4.1592, -1.4585, 0.1972, 4.8177 \cdot 10^{-5}, -4.6897 \cdot 10^{-5}, -3.3277 \cdot 10^{-5}, 4.2732 \cdot 10^{-5}, -1.0736 \cdot 10^{-5}]$	
	$R_4$	$\theta_4^T = [3.5848, -4.8712, 3.0601, -0.8468, 0.0731, 4.9745 \cdot 10^{-5}, -3.1024 \cdot 10^{-5}, -4.3869 \cdot 10^{-5}, 3.1774 \cdot 10^{-5}, -3.5471 \cdot 10^{-6}]$	
	$R_5$	$\theta_5^T = [3.6440, -4.9843, 3.0768, -0.7769, 0.0404, 4.9610 \cdot 10^{-5}, -3.3783 \cdot 10^{-5}, -4.6716 \cdot 10^{-5}, 3.2902 \cdot 10^{-5}, -2.0593 \cdot 10^{-6}]$	

En la Figura 4.25 y en la Figura 4.26 se presentan los conjuntos difusos de las premisas del modelo difuso utilizando los parámetros presentes en la Tabla 4.14. En particular, hay dos pares de conjuntos difusos asociados a los regresores de la salida (Figura 4.25) que podrían ser mezclados como un solo *cluster*, a saber, las funciones de pertenencia de la Regla 3 y la Regla 4 poseen medias bastantes cercanas, mientras que el conjunto difuso de la Regla 1 superpone completamente a la Regla 2. A su vez, se observa el comportamiento de los conjuntos difusos asociados a los regresores de la entrada (Figura 4.26) de las reglas mencionadas anteriormente, y se aprecia que la Regla 1 ya no superpone a la Regla 2, mientras que la Regla 3 presenta un comportamiento dispar respecto a la Regla 4, dependiendo de la premisa. Dicho lo anterior, no es directo reducir la complejidad del modelo mediante la disminución del número de reglas, ya que en este caso todas ellas abarcan distintas condiciones de operación que permiten que el modelo obtenga un buen desempeño a la hora de predecir el comportamiento de la planta. Adicional a lo anterior, se tiene que los conjuntos difusos obtenidos poseen en general varianzas acotadas, permitiendo identificar de mejor manera los rangos de operación de cada regla.



**Figura 4.25: Conjuntos Difusos de los Regresores  $y(k - j)$  para el Mejor Modelo Identificación a un Paso del SNLC con Datos en Lazo Cerrado.**

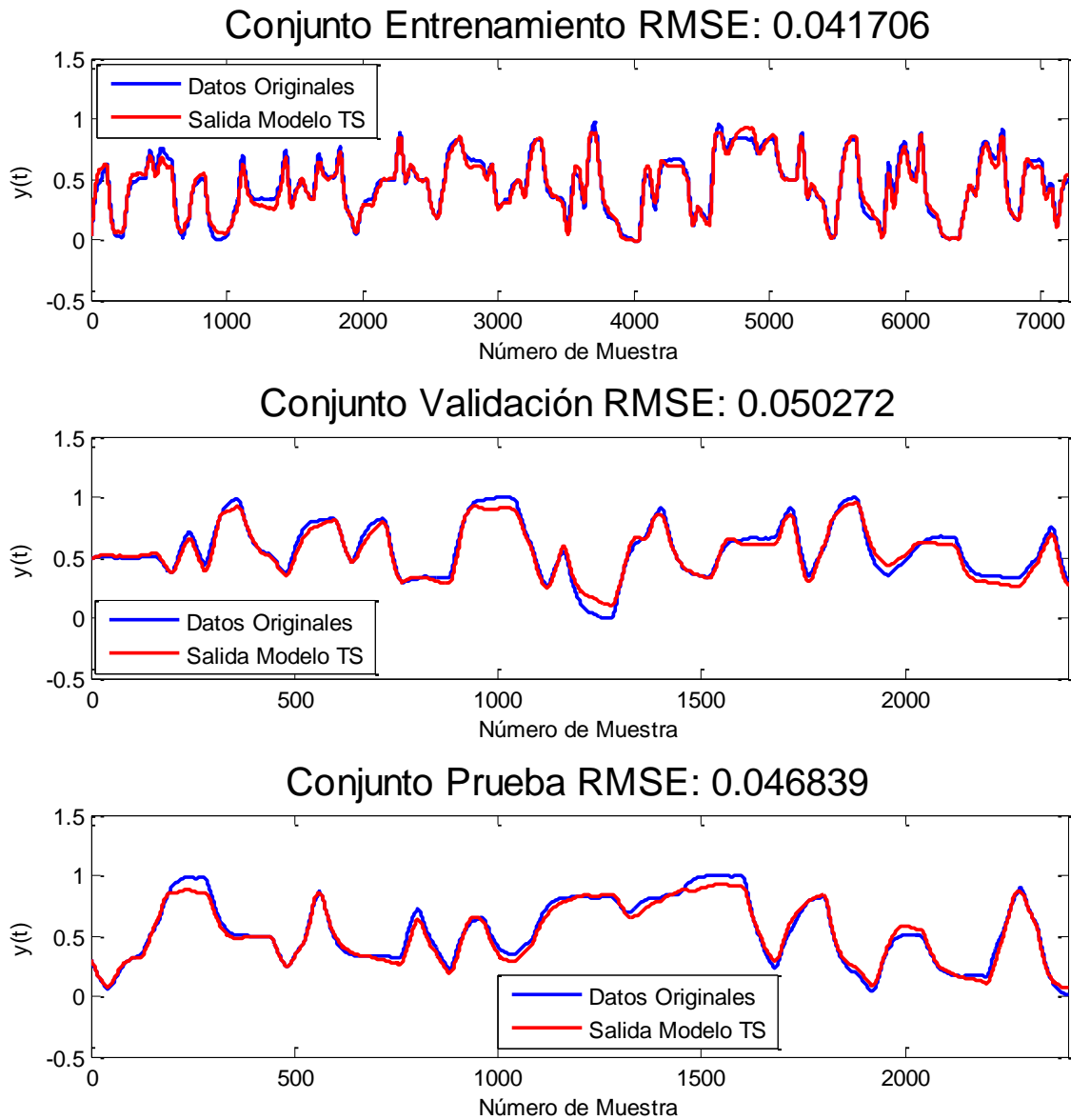




**Figura 4.26: Conjuntos Difusos de los Regresores  $u(k - j)$  para el Mejor Modelo Identificación a un Paso del SNLC con Datos en Lazo Cerrado.**

Se confirma que todos los modelos locales del modelo difuso obtenido por identificación a un paso son controlables, sin embargo, la segunda regla del modelo posee un consecuente inestable, afectando a la estabilidad del modelo completo, a pesar de que los datos de entrada-salida utilizados para la identificación son tomados en lazo cerrado, por lo que los datos sólo contienen comportamientos estables de la planta. Lo anterior reafirma el hecho de que el proceso de identificación de sistemas no siempre permite obtener un modelo que herede fielmente las características de la planta original, y más aún cuando se trabaja con un modelo en tiempo discreto de una planta en tiempo continuo, debido a que la discretización puede generar pérdida de información.

Por otro lado, en la Figura 4.27 se presenta el desempeño del mejor modelo obtenido por identificación a  $N$  pasos (usando QN como método de optimización), donde se observa la buena predicción del modelo óptimo utilizando horizontes de predicción mayores.



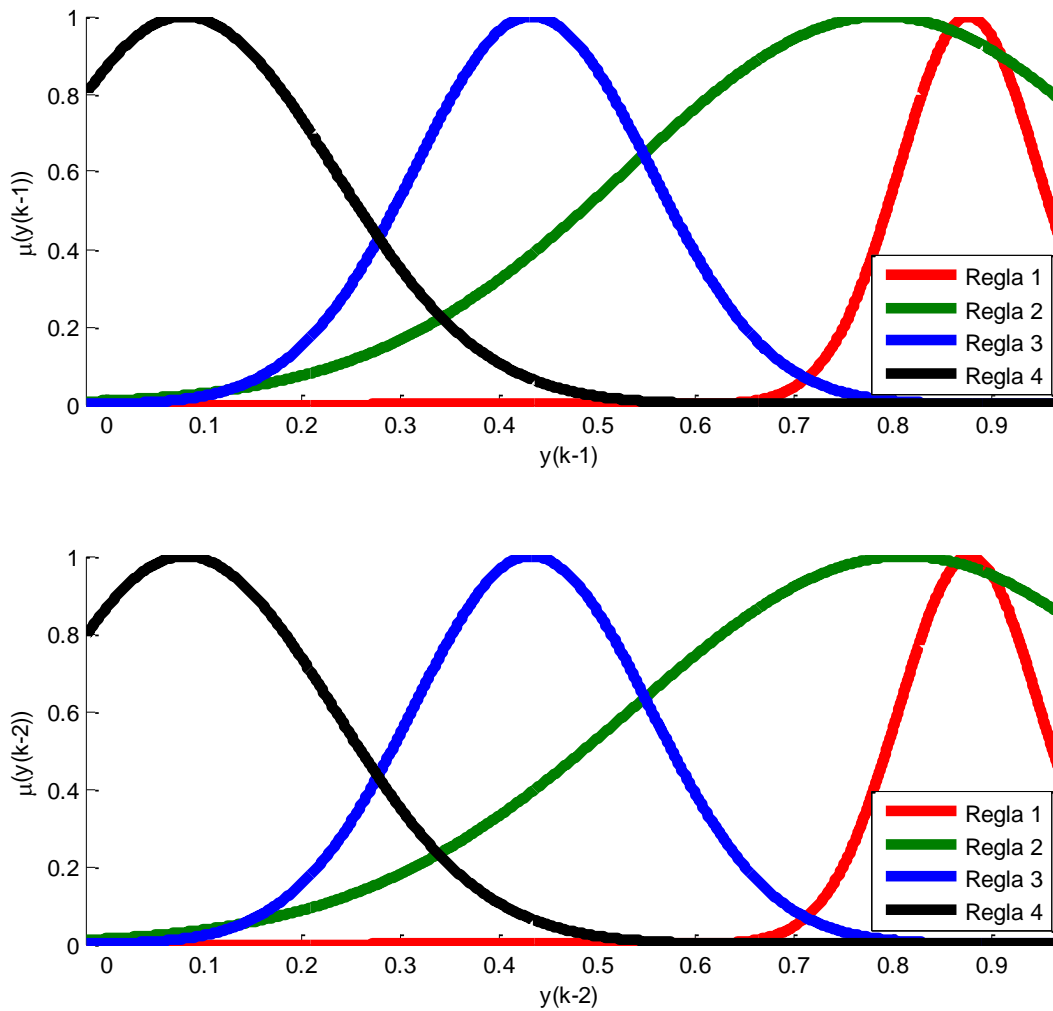
**Figura 4.27: Resultado Mejor Modelo Identificación a  $N$  Pasos del SNLC. Arriba: Conjunto de Entrenamiento. Medio: Conjunto de Validación. Abajo: Conjunto de Prueba.**

La estructura del modelo obtenido por identificación a  $N$  pasos se detalla en la Tabla 4.15.

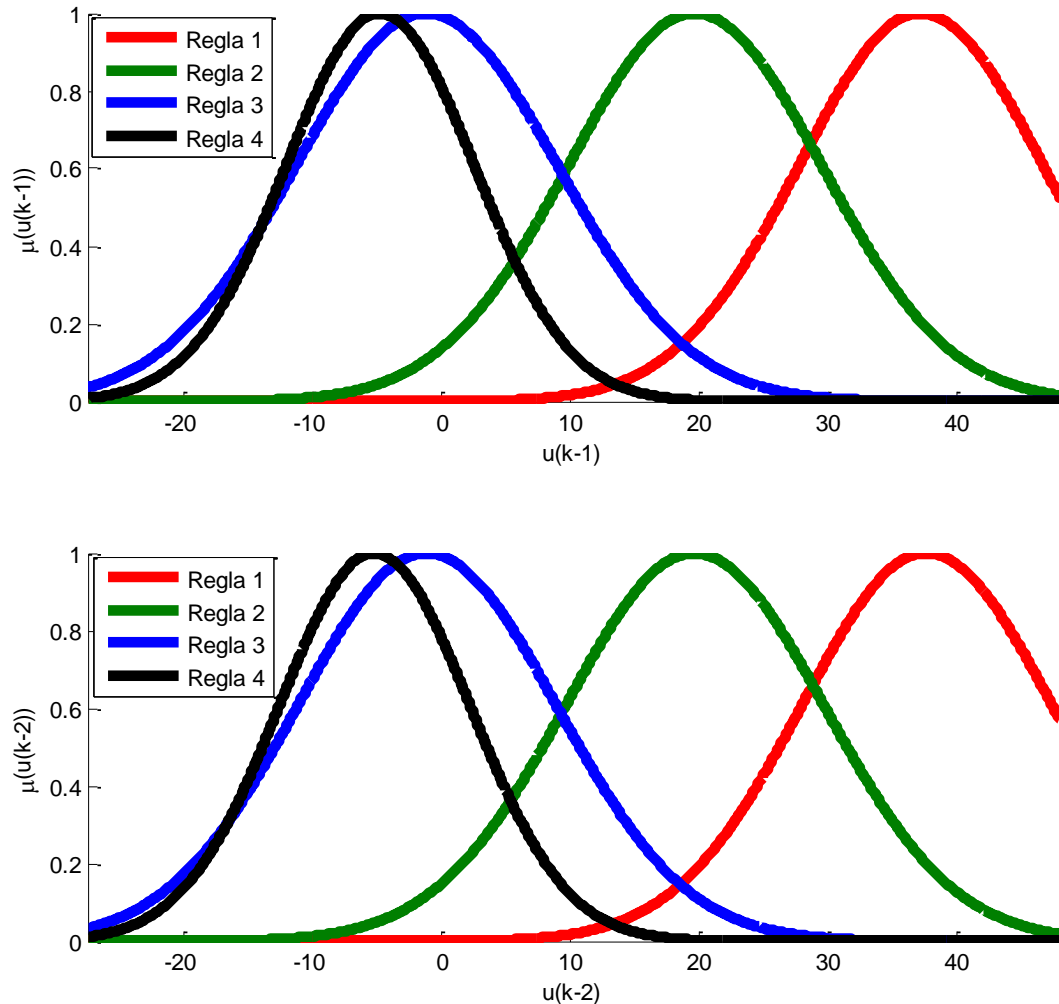
Tabla 4.15: Estructura Mejor Modelo Identificación a  $N$  Pasos del SNLC con Datos en Lazo Cerrado.

<b>Premisas</b>	$y(k-1), y(k-2), u(k-1), u(k-2)$		
<b>Cantidad de Reglas</b>	4		
<b>Parámetros Premisas</b>	$R_1$	$a_{1,i}$	[14.0963, 13.9429, 0.1053, 0.1028]
		$b_{1,i}$	[0.8776, 0.8785, 37.2617, 37.7136]
	$R_2$	$a_{2,i}$	[3.8631, 3.6051, 0.1015, 0.0997]
		$b_{2,i}$	[0.7907, 0.8136, 19.7364, 19.7073]
	$R_3$	$a_{3,i}$	[8.3161, 8.2789, 0.0983, 0.0995]
		$b_{3,i}$	[0.4342, 0.4333, -1.1001, -1.0660]
	$R_4$	$a_{4,i}$	[6.6278, 6.6493, 0.1365, 0.1350]
		$b_{4,i}$	[0.0814, 0.0821, -4.6981, -5.1344]
<b>Parámetros Consecuencias</b>	$R_1$	$\theta_1^T = [1.8206, -0.8246, 6.3564 \cdot 10^{-5}, 2.3320 \cdot 10^{-5}]$	
	$R_2$	$\theta_2^T = [1.8891, -0.8903, 5.1082 \cdot 10^{-5}, 3.0449 \cdot 10^{-5}]$	
	$R_3$	$\theta_3^T = [1.9527, -0.9531, 5.6203 \cdot 10^{-5}, 5.4921 \cdot 10^{-5}]$	
	$R_4$	$\theta_4^T = [1.9674, -0.9678, 4.8062 \cdot 10^{-5}, 4.8366 \cdot 10^{-5}]$	

De las gráficas de los conjuntos difusos presente en la Figura 4.28 y la Figura 4.29 se observa que ninguna de las funciones de pertenencia es prescindible, debido a que todas las reglas del modelo difuso abarcan distintas condiciones de operación del SNLC, por lo que reducir la complejidad del modelo mediante la disminución del número de reglas puede traer consigo una pérdida de desempeño del modelo. A su vez, se observa que el algoritmo de *clustering* retorna un ajuste bien definido de cada una de las zonas en que operan las variables.



**Figura 4.28: Conjuntos Difusos de los Regresores  $y(k - j)$  para el Mejor Modelo Identificación a  $N$  Pasos del SNLC con Datos en Lazo Cerrado.**



**Figura 4.29: Conjuntos Difusos de los Regresores  $u(k - j)$  para el Mejor Modelo Identificación a  $N$  Pasos del SNLC con Datos en Lazo Cerrado.**

Analizando los modelos locales descritos por los parámetros presentes en la Tabla 4.15 se tiene nuevamente que todos los consecuentes son controlables, y esta vez, a diferencia del modelo obtenido por identificación a un paso, todas las reglas poseen modelos locales estables. Se corrobora también la estabilidad del modelo difuso completo mediante el procedimiento descrito en la Sección 8.5 de anexos.

Una vez obtenidos los modelos de predicción difusos del SNLC, se procede al diseño de controladores predictivos para validar la metodología empleada.

#### 4.2.2.2. Control Predictivo Difuso del SNLC

En este caso se diseñan controladores predictivos que resuelvan el siguiente problema de optimización:

$$\begin{aligned}
 \min_{u(k), u(k+1), \dots, u(k+N_p-1)} J(z(k), u(k)) &= Q \sum_{i=1}^{N_p} \left( r(k+i) - y_{fuzzy}(k+i|z(k)) \right)^2 \\
 &+ \sum_{i=1}^{N_p} u^2(k+i-1), \quad (4.31) \\
 \text{s. a. } y_{fuzzy}(k+i|z(k)) &= \sum_{r=1}^{N_r} h_r(z(k+i)) \cdot \theta_r^T z(k+i), \quad i = 1, \dots, N_p \\
 |u(k+i)| &\leq 50, \quad i = 1, \dots, N_p,
 \end{aligned}$$

donde  $r(k+i)$  es la referencia en el instante  $k+i$ , en este caso se considera  $r(k+i) = r = 0.5, \forall i$ , para evitar que la planta tome valores distintos a los utilizados en el entrenamiento y provoque una mala predicción;  $Q > 0$  es un parámetro de sintonización del controlador;  $N_p$  es el horizonte de predicción (en este caso se considera  $N_p = 1$  y  $N_p = 10$ );  $y_{fuzzy}(k+i|z(k))$  es la predicción a  $i$  pasos realizada por el modelo difuso;  $z(k+i) = [y(k+i-1), \dots, y(k+i-n_y), u(k+i-1), \dots, u(k+i-n_u)]$ ; y  $u(k)$  es la acción de control en el instante  $k$ . Notar que se incluye la restricción  $|u(k+i)| \leq 50$  para evitar posibles problemas numéricos si es que la acción de control  $u(k)$  toma valores muy distantes a los conjuntos difusos entrenados, debido a que si numéricamente los grados de pertenencia  $w_r(z(k))$  de las premisas del modelo a los conjuntos es cero, entonces el grado de activación normalizado  $h_r(z(k))$  queda indeterminado.

Al igual que en el CSTR, en este caso se aplica SQP como algoritmo de optimización para resolver (4.31). A su vez, se sintonizan los controladores utilizando el ITAE como función objetivo, y como principal métrica de desempeño el error en estado estacionario  $e_{ss} = |r - y_{ss}|$ , donde  $y_{ss}$  es el valor en estado estacionario de la salida de la planta.

Los resultados obtenidos utilizando los distintos modelos de predicción se resumen en la Tabla 4.16. Se observa que todos los casos abordados presentan resultados sobresalientes, donde el controlador con peor desempeño en lazo cerrado alcanza un error en estado estacionario de apenas el 0.3807%.

Cabe destacar que los modelos que tienen la misma estructura (identificación a  $N$  pasos utilizando QN y NM) poseen un desempeño en lazo cerrado similar, donde la diferencia se asocia a las variaciones en el valor de los parámetros del modelo difuso, lo cual repercute en su desempeño como predictor, afectando el rendimiento del controlador.

El modelo que mejor desempeño en lazo cerrado genera es el obtenido por identificación a  $N$  pasos usando Quasi-Newton como algoritmo de optimización, lo cual se debe a que dicho modelo es el que mejor RMSE presenta al predecir el comportamiento de la planta en horizontes de predicción mayores, como se observa en la Tabla 4.13. Por otro lado, de los modelos obtenidos por identificación a  $N$  pasos, el que peor desempeño muestra cuando se utiliza  $N_p = 10$  es el obtenido al usar PSO como *solver*, debido a que dicho modelo es el que peor índice de rendimiento tiene al momento de predecir el comportamiento de la planta a  $N$  pasos.

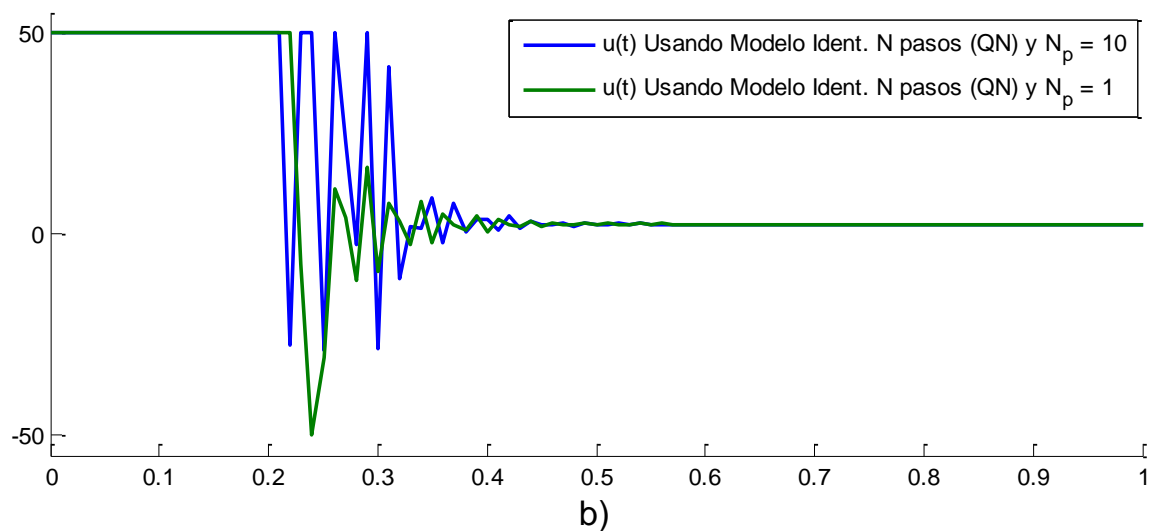
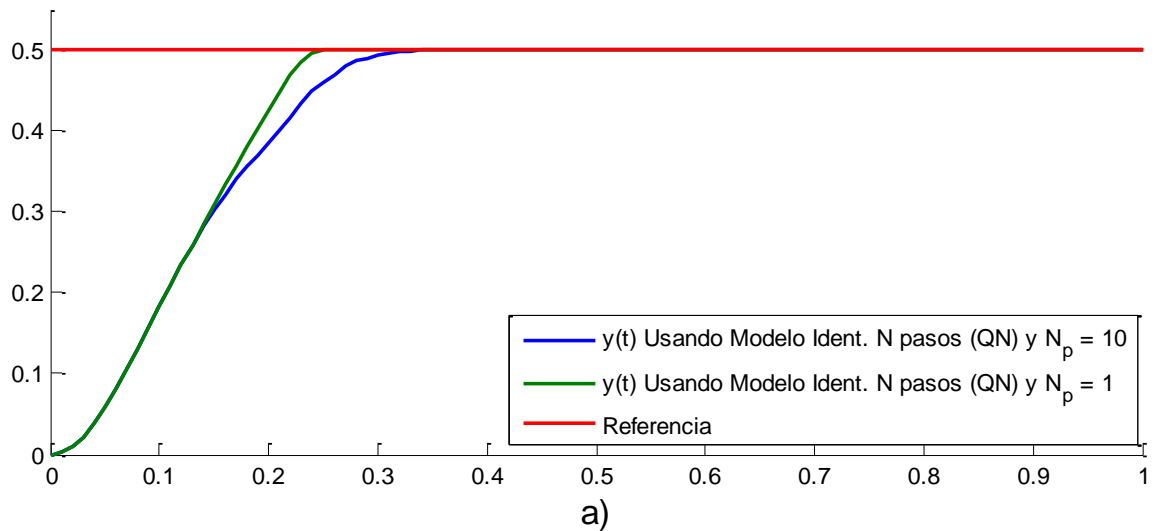
Con respecto al esfuerzo de control, se mantiene la tendencia de los resultados obtenidos en la Sección 4.2.1.2, donde el valor de  $u^T(k)u(k)$  aumenta cuando se considera un horizonte de predicción mayor debido a que los controladores son sintonizados con el fin de minimizar  $e_{ss}$ , sin tener en mayor consideración la variable manipulada, y en consecuencia el factor asociado a la entrada en la función objetivo del controlador predictivo posee una menor ponderación que los errores de predicción.

**Tabla 4.16: Resultados Control Predictivo Difuso del SNLC con Modelos de Predicción Obtenidos con Datos en Lazo Cerrado.**

<i>Modelo de Predicción</i>			<b>Control Predictivo Difuso</b>					
<b>Ident.</b>	<b>Solver</b>	$\begin{bmatrix} n_y^* \\ n_u^* \\ \text{reglas}^* \end{bmatrix}$	$Q$		$e_{ss}$		$u^T(k)u(k)$	
			$N_p = 1$	$N_p = 10$	$N_p = 1$	$N_p = 10$	$N_p = 1$	$N_p = 10$
1 Paso	NM/ QN/ PSO	[5,5,5]	3.1219 e+11	1.9329 e+08	5.2287 e-04 (0.10457%)	3.426 e-04 (0.0685%)	1.8912 e+05	1.9925 e+05
$N$ pasos	QN	[2,2,4]	0.0049 e+11	6.0093 e+08	1.0697 e-04 (0.0213%)	<b>3.2479</b> <b>e-05</b> <b>(0.0064%)</b>	6.2176 e+04	7.0308 e+04
	NM	[2,2,4]	0.0063 e+11	6.2524 e+08	<b>9.826</b> <b>e-05</b> <b>(0.0196%)</b>	3.788 e-05 (0.0075%)	6.5896 e+04	7.3570 e+04
	PSO	[5,2,4]	0.0032 e+11	3.2709 e+08	1.0927 e-04 (0.0218%)	1.9038 e-03 (0.3807%)	6.1545 e+04	1.3897 e+05

En la Figura 4.30 se presenta la respuesta del SNLC al emplear el control predictivo con el modelo obtenido por identificación a  $N$  pasos y Quasi-Newton como *solver*. En particular, la salida de la planta (Figura 4.30 a)) alcanza la referencia deseada al utilizar tanto  $N_p = 10$  como  $N_p = 1$ , donde a simple vista no se aprecian grandes diferencias en el error en estado estacionario, y la principal disimilitud se presenta en el tiempo de respuesta de la planta, donde el sistema es más lento al emplear  $N_p = 10$ , resultado similar al obtenido en los casos expuestos en la Sección 4.2.1.2. Por otro lado, la acción de control (Figura 4.30 b)) alcanza el valor límite en los primeros instantes de modo de

reducir el error rápidamente, para luego comenzar a oscilar y disminuir su valor absoluto, de tal manera que converja a un valor cercano a cero, otorgando una solución óptima en los instantes finales donde se minimiza tanto el error como la acción de control.



**Figura 4.30: Resultados Control Predictivo Difuso del SNLC Usando Modelos de Predicción Obtenidos con Datos en Lazo Cerrado e Identificación a  $N$  Pasos (QN). a) Salida  $y(t)$  vs Tiempo [s]. b) Entrada  $u(t)$  vs Tiempo [s].**

De lo anterior se tiene que los modelos difusos localmente controlables en los puntos de operación de interés obtenidos por la metodología propuesta permiten el diseño adecuado de controladores, generando buenos desempeños en lazo cerrado del sistema en estudio, respaldando la hipótesis planteada inicialmente.



Cabe destacar que en este caso, y a diferencia del CSTR, se logra implementar un controlador predictivo con un buen desempeño a partir de datos de entrada-salida en lazo cerrado obtenidos a partir de un controlador de otra clase (en este caso, un controlador por linealización exacta). Por lo tanto, si es posible obtener buenos datos de identificación, independiente de las estrategias de control utilizadas para ello, los modelos empíricos obtenidos a partir de los métodos propuestos permiten implementar estrategias de control de buena manera debido a su propiedad de controlabilidad.

Si bien en los casos de estudio analizados es necesario conocer el sistema original para poder diseñar una estrategia de control que permita obtener datos en lazo cerrado, dicho supuesto se podría obviar implementando otras estrategias de control que no requieran un conocimiento acabado del proceso en estudio. Un ejemplo de este tipo de estrategias puede ser un controlador difuso PID de Mamdani [63][110], donde el uso de la lógica difusa y la derivación de una base de reglas como la propuesta por MacVicar-Whelan [59] permite controlar procesos sin conocer explícitamente su estructura, siendo necesario solamente ajustar las ganancias del controlador de manera adecuada para lograr dicho cometido.

A modo de síntesis, de los resultados obtenidos en ambos casos de estudio analizados se puede apreciar que se obtiene un mejor desempeño al utilizar como condición inicial de las variables de optimización la solución del problema de identificación de parámetros de las consecuencias sin considerar las restricciones de controlabilidad, siendo una mejor aproximación que utilizar condiciones iniciales aleatorias, mejorando bastante el rendimiento de los algoritmos de identificación utilizados. Por otro lado, los mejores resultados de la identificación no siempre se obtienen con un mismo algoritmo de optimización, como se observa en la Tabla 4.6 y la Tabla 4.7 por ejemplo, sin embargo, todos los *solvers* empleados demuestran obtener resultados aceptables. En general, Nelder-Mead logra obtener mejores resultados debido al tipo de función objetivo con restricciones utilizada, ya que dicho método posee un mejor manejo de funciones discontinuas en comparación a Quasi-Newton y PSO.

A su vez, los resultados de identificación a  $N$  pasos obtenidos en ambas pruebas demuestran buenos resultados tanto en predicciones a varios pasos como a un paso, estableciendo cierta ventaja de este tipo de metodología por sobre la identificación a un paso utilizada comúnmente en la literatura.

Por último, la mayoría de los modelos óptimos obtenidos poseen un orden mayor a la planta original, de tal forma de ajustar mejor las no linealidades de la planta original. No obstante, por ejemplo para el CSTR, se tiene que los resultados utilizando el orden de la planta son aceptables, en especial para identificación a un paso, como se ve en las tablas presentes en la Sección 8.3.1 de anexos, obteniendo bajos valores de RMSE utilizando estructuras menos complejas.

Si bien la metodología descrita en este capítulo busca encontrar modelos que sean localmente controlables (asegurando al menos controlabilidad en los puntos de

operación presentes en los datos), descartando como soluciones del procedimiento de identificación aquellos que no lo sean, puede darse el caso en que no sea posible generar un modelo que cumpla estrictamente con las restricciones impuestas. No obstante, relajar las restricciones anteriores, con el fin de obtener modelos que sean controlables en un espacio menor, eventualmente podría ser suficiente para obtener un buen desempeño en lazo cerrado.

Luego, una vez planteada y validada la metodología de identificación difusa con restricciones de controlabilidad, se procede en el siguiente capítulo a abordar una nueva propuesta de identificación difusa: incluir términos de orden superior en los consecuentes del modelo difuso de tal manera de ampliar el espacio de controlabilidad de éste en comparación a una aproximación de primer orden.

## 5. Identificación Difusa con Términos de Orden Superior

Otra manera de mejorar el rendimiento en lazo cerrado de controladores basados en modelos difusos es incluir términos de orden superior en sus consecuencias. En particular, este trabajo se centra en el uso de aproximaciones de segundo orden, motivado por lo expuesto en [79], donde se plantea el uso de este tipo de aproximaciones como modelos de predicción de un controlador predictivo.

### 5.1. Aproximaciones de Segundo Orden

La motivación para utilizar aproximaciones de segundo orden se basa en la topología de funciones. Sea  $f: R^n \rightarrow R$  una función no lineal representada como  $y = f(z)$ . Si  $f(z)$  posee gradiente no nulo en algún punto  $z = z_0$ , entonces es posible encontrar una aproximación lineal en torno a dicho punto. Sin embargo, si  $z_0$  es un punto crítico (es decir, el gradiente de  $f$  en  $z_0$  es cero), entonces dicha aproximación lineal no es posible. Luego, si  $z = z_0$  es un punto crítico no degenerado (es decir, el determinante del Hessiano de la función es no nulo), entonces se puede emplear el Lema de Morse para describir el comportamiento local de  $f(z)$  entorno a  $z_0$ . En general, dicha aproximación casi nunca será numéricamente exacta, pero puede ser geoméricamente exacta [79].

El Lema de Morse indica que, dada una función  $f: R^n \rightarrow R$  con un punto crítico no degenerado en  $z = z_0$ , en la vecindad del punto crítico existe un cambio de coordenadas a  $y_1(z), \dots, y_n(z)$  tal que  $y_1(z_0) = \dots = y_n(z_0) = 0$  y  $f(z)$  toma la forma

$$f(z) = f(z_0) + \frac{1}{2} \sum_{i,j} H_{ij}(z_0) y_i y_j, \quad (5.1)$$

donde  $i, j = 1, \dots, n$  y  $H_{ij}(z_0)$  es el Hessiano de  $f(\cdot)$  con respecto a las coordenadas  $i, j$  evaluado en  $z_0$ . En otras palabras, el Lema de Morse indica que una función en un punto crítico no degenerado puede ser descrita de mejor manera utilizando términos de segundo orden, en comparación a utilizar sólo una aproximación lineal.

En [79] se plantea el uso de técnicas de perturbación regular para la obtención de modelos con términos de segundo orden. Para ello, considérese un sistema no lineal dado por

$$\begin{aligned} \dot{x}(t) &= F(x(t), u(t)) \\ y(t) &= G(x(t), u(t)), \end{aligned} \quad (5.2)$$

donde  $x(t) \in R^n$  es el estado,  $u(t) \in R^m$  la entrada e  $y(t) \in R^z$  la salida del sistema en el tiempo  $t$ . A su vez,  $F(\cdot)$  y  $G(\cdot)$  son vectores de funciones, e.g.,  $F(\cdot) = [f_1(\cdot), \dots, f_n(\cdot)]^T$ . El sistema en (5.2) puede ser aproximado alrededor de un punto de operación  $(x_0(t), u_0(t))$  utilizando series de Taylor de la forma

$$\begin{aligned} \dot{x}(t) = & F(x_0(t), u_0(t)) + \nabla_x F(t) \Delta x(t) + \nabla_u F(t) \Delta u(t) + \frac{1}{2} \Delta x^T(t) (\nabla_{xx} F(t)) \Delta x^T(t) \\ & + \Delta x^T(t) (\nabla_{xu} F(t)) \Delta u^T(t) + \frac{1}{2} \Delta u^T(t) (\nabla_{uu} F(t)) \Delta u^T(t) + \dots, \end{aligned} \quad (5.3)$$

con  $\Delta x(t) = x(t) - x_0(t)$ ;  $\Delta u(t) = u(t) - u_0(t)$ ;  $\nabla_x F(t)$  y  $\nabla_u F(t)$  son las derivadas de primer orden de  $F(x(t), u(t))$  con respecto a  $x$  y  $u$  respectivamente, evaluadas en el punto de operación; y  $\nabla_{xx} F(t)$ ,  $\nabla_{xu} F(t)$  y  $\nabla_{uu} F(t)$  son las derivada de segundo orden de  $F(x(t), u(t))$  evaluadas en  $(x_0(t), u_0(t))$ .

Sea  $x_{p1}(t)$  y  $x_{p2}(t)$  las trayectorias asociadas a las perturbaciones de primer y segundo orden respectivamente, luego se puede aproximar (5.3) considerando

$$\begin{aligned} x(t) &= x_0(t) + x_{p1}(t) + x_{p2}(t) \\ \dot{x}_{p1}(t) &= F(x_0(t), u_0(t)) + J_x(t) x_{p1}(t) + J_u(t) \Delta u(t) \\ \dot{x}_{p2}(t) &= J_x(t) x_{p2}(t) + \Xi(x_{p1}(t), \Delta u(t)), \end{aligned} \quad (5.4)$$

con  $J_x(t) = \left. \frac{dF(x(t), u(t))}{dx(t)} \right|_{(x(t), u(t))=(x_0(t), u_0(t))}$ ,  $J_u(t) = \left. \frac{dF(x(t), u(t))}{du(t)} \right|_{(x(t), u(t))=(x_0(t), u_0(t))}$  y el  $j$ -ésimo elemento de  $\Xi(x_{p1}(t), \Delta u(t))$  es determinado por la expresión

$$\begin{aligned} \Xi_j(x_{p1}(t), \Delta u(t)) = \\ \frac{1}{2} [x_{p1}^T(t) H_{xx,j}(t) x_{p1}(t) + 2x_{p1}^T(t) H_{xu,j}(t) \Delta u(t) + \Delta u^T(t) H_{uu,j}(t) \Delta u(t)], \end{aligned} \quad (5.5)$$

donde el  $j$ -ésimo elemento de los Hessianos  $H(t)$  está dado por

$$\begin{aligned} H_{xx,j}(t) &= \left. \frac{d^2 f_j(x(t), u(t))}{dx^2(t)} \right|_{(x(t), u(t))=(x_0(t), u_0(t))} \\ H_{xu,j}(t) &= \left. \frac{d^2 f_j(x(t), u(t))}{dx(t) du(t)} \right|_{(x(t), u(t))=(x_0(t), u_0(t))} \\ H_{uu,j}(t) &= \left. \frac{d^2 f_j(x(t), u(t))}{du^2(t)} \right|_{(x(t), u(t))=(x_0(t), u_0(t))} \end{aligned} \quad (5.6)$$

Dado lo anterior, se discretiza el modelo cuadrático continuo mediante la resolución de las ecuaciones diferenciales en (5.4) y la posterior integración en un intervalo de muestreo  $[t, t + T_s]$ , con  $T_s$  el tiempo de muestreo, donde se asume que  $\Delta u(t)$  se mantiene constante en dicho intervalo, y que las ecuaciones en (5.4) se resuelven secuencialmente. Luego, si se considera  $x_{p1}(0) = x_{p2}(0) = 0$  y  $(x_0(k), u_0(k)) = (x(k), u(k - 1))$ , la predicción a un paso está dada por

$$\begin{aligned} x(k+1) &= x(k) + \psi(k)F(x_0(t), u_0(t)) + \Gamma(k)\Delta u(k) + \psi(k)\eta(k) \\ \psi(k) &= \int_0^{T_s} \exp(J_x(k)\tau) d\tau, \quad \Gamma(k) = \int_0^{T_s} \exp(J_x(k)\tau) J_u(k) d\tau, \end{aligned} \quad (5.7)$$

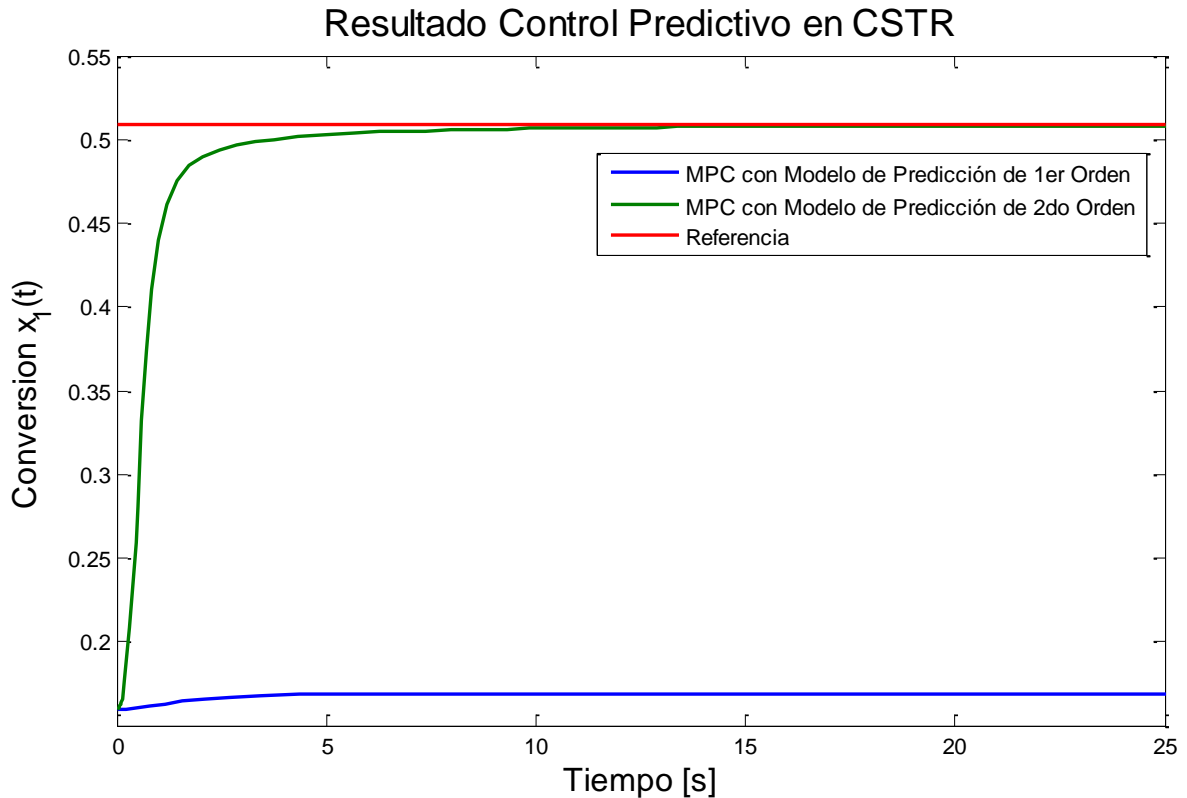
y donde el  $j$ -ésimo elemento de  $\eta(k)$  se computa como

$$\begin{aligned} \eta_j(k) &= \eta_{j,0}(k) + \eta_{j,1}(k)\Delta u(k) + \Delta u^T(k)\eta_{j,2}(k)\Delta u(k), \\ \eta_{j,0}(k) &= \frac{1}{2} [\psi(k)F(x_0(t), u_0(t))]^T H_{xx,j}(k) [\psi(k)F(x_0(t), u_0(t))], \\ \eta_{j,1}(k) &= [\psi(k)F(x_0(t), u_0(t))]^T [H_{xx,j}(k)\Gamma(k) + H_{xu,j}(k)], \\ \eta_{j,2}(k) &= \frac{1}{2} \Gamma^T(k) H_{xx,j}(k) \Gamma(k) + \Gamma^T(k) H_{xu,j}(k) + \frac{1}{2} H_{uu,j}(k). \end{aligned} \quad (5.8)$$

Con la aproximación en (5.7)-(5.8) es posible implementar un controlador predictivo que utilice el modelo de segundo orden como predictor, teniendo la ventaja de que en los puntos de operación donde se anulen los términos de primer orden (puntos críticos no degenerados), los términos de segundo orden actúan como respaldo, permitiendo inferir de manera correcta acciones de control, aumentando el espacio controlable del sistema en estudio en comparación a la aproximación de primer orden.

Para probar la efectividad de la estrategia de control predictivo utilizando aproximaciones de segundo orden se emplea como planta de prueba el CSTR presentado en el Capítulo 1. Cabe recordar que el objetivo de control del reactor es alcanzar el máximo valor de conversión, luego, la referencia es 0.5088.

En la Figura 5.1 se presenta el resultado al emplear el control predictivo utilizando aproximaciones de segundo orden como modelo de predicción, en comparación al resultado obtenido al usar una aproximación de primer orden como predictor. Se logra observar que las aproximaciones de segundo orden permiten mejorar el desempeño en lazo cerrado de la planta en estudio, debido a que los términos cuadráticos permiten aumentar el espacio de controlabilidad, logrando inferir adecuadamente acciones de control, con las cuales es posible alcanzar la referencia deseada.



**Figura 5.1: Resultado Control Predictivo Utilizando Modelos de Predicción de 1er Orden y de 2do Orden.**

Cabe destacar que la aproximación de segundo orden del sistema analizado debe ser computada en línea en cada instante  $k$ . Luego, si bien el controlador predictivo utilizando un modelo de predicción cuadrático tiene un buen desempeño en lazo cerrado, alcanzando la referencia deseada, éste puede verse afectado significativamente cuando se introducen perturbaciones en el proceso. Esto se debe a que la principal falencia del uso de aproximaciones en torno a un punto de operación es que éstas son válidas sólo en un rango acotado, por lo que si se produce un cambio en el punto de operación de la planta debido a perturbaciones el modelo pierde validez, y en consecuencia disminuye su calidad como predictor, afectando el desempeño del controlador predictivo [33].

Dado lo anterior, se propone considerar la metodología dada por el uso de aproximaciones de segundo orden y extenderla a los modelos de Takagi-Sugeno, de tal manera de aprovechar la robustez del modelo difuso al emplear modelos locales para distintos puntos de operación, viéndose de esta manera menos afectado por perturbaciones externas, y conservando la propiedad de ampliar el espacio de controlabilidad dada por las aproximaciones de segundo orden.

## 5.2. Modelos Difusos de Takagi-Sugeno con Términos de Segundo Orden

Si bien los modelos difusos de Takagi-Sugeno se han catalogado como aproximadores universales, éstos pueden tener los mismos problemas que puede presentar un modelo lineal, es decir, puede que las consecuencias del modelo difuso no logren reflejar el efecto de la entrada en la trayectoria de estados. Por lo tanto, de ocurrir lo anterior el controlador puede que no conduzca al sistema a las condiciones de operación deseadas.

Dado lo anterior, una mejora con respecto a los modelos de Takagi-Sugeno comúnmente utilizados (los cuales consideran modelos locales lineales/afines), es plantear el uso de aproximaciones de segundo orden en los consecuentes del modelo, de tal forma de aprovechar las ventajas que este tipo de aproximaciones otorga con respecto al rango de controlabilidad.

Luego, basado en lo expuesto en la Sección 2.1 y 5.1, las reglas del modelo difuso quedan planteadas de la forma

$$R_r: \text{ Si } z_1(k) \text{ es } MF_1^r \text{ y } \dots \text{ y } z_p(k) \text{ es } MF_p^r \text{ entonces} \\ y_r(z(k)) = C \left( x(k) + \psi(k)F(x_0(t), u_0(t)) + \Gamma(k)\Delta u(k) + \psi(k)\eta(k) \right), \quad (5.9)$$

con  $\psi(k)$ ,  $\Gamma(k)$  y  $\eta(k)$  definidos como en (5.7)-(5.8), donde cabe recordar que  $\eta(k)$  es el término que contiene los términos cuadráticos.  $C$  corresponde a la matriz de observación del sistema.

A diferencia de la metodología expuesta en la Sección 5.1, el modelo difuso con términos de segundo orden no requiere calcular los Jacobianos y Hessianos del sistema en cada instante de tiempo  $k$ , sino que dichos parámetros son determinados en el procedimiento de identificación difusa a partir de datos de entrada-salida del sistema, lo que permite reducir el tiempo de cómputo de las predicciones, haciendo más atractivo el uso de esta estrategia en aplicaciones de tiempo real en comparación al controlador predictivo presentado en la Sección 5.1. Para determinar la estructura y parámetros de un modelo difuso con términos de segundo orden se emplea la misma metodología de identificación presentada en el Capítulo 2.

Cabe destacar que, dado que los modelos locales se basan en aproximaciones de segundo orden en torno a un punto de operación (el cual corresponde al punto anterior), en esta oportunidad no se utiliza  $u(k-j)$  como regresores, sino que se utiliza su variación,  $\Delta u(k-j)$ . A su vez, los modelos presentes en esta sección son descritos en variables de estado, a diferencia de la metodología presentada en el Capítulo 4, donde se utilizan modelos entrada-salida para describir la dinámica del sistema. En consecuencia, se asume la observabilidad de los sistemas en estudio para poder realizar mediciones de los estados presentes y así formular el vector de premisas del modelo difuso TS.

A continuación se procede a aplicar la estrategia difusa propuesta en un caso de estudio: el reactor de estanque continuamente agitado; de tal forma de comparar el desempeño del controlador predictivo difuso propuesto en la siguiente sección con el controlador predictivo presentado en la Sección 5.1.

## 5.3. Caso de Estudio: Reactor de Estanque Continuosamente Agitado

Para validar la metodología propuesta en la Sección 5.2 se emplea nuevamente el CSTR descrito por las ecuaciones (1.1)-(1.3) en el Capítulo 1. Para ello, primero se plantea la etapa de identificación del modelo difuso con términos de segundo orden en sus consecuencias, para luego utilizarlo en el diseño de una estrategia de control predictivo difuso.

### 5.3.1. Identificación del CSTR

Para la identificación del modelo difuso con términos de segundo orden se utiliza el mismo conjunto de datos de identificación obtenidos en lazo cerrado empleado en las pruebas de la Sección 4.2.1.1. A su vez, se considera inicialmente como premisas del modelo

$$z(k) = [x_1(k-1), \dots, x_1(k-5), x_2(k-1), \dots, x_2(k-5), \Delta u(k-1), \dots, \Delta u(k-5), x_1^2(k-1), \dots, x_1^2(k-5), x_2^2(k-1), \dots, x_2^2(k-5), \Delta u^2(k-1), \dots, \Delta u^2(k-5)]. \quad (5.10)$$

Note que las premisas se escogen a partir de (5.9), donde se observa la dependencia de la salida con los estados, la variación de la entrada y sus respectivos términos cuadráticos, obtenidos a partir de las aproximaciones de segundo orden. La cantidad máxima de retardos es elegida a partir de la función de autocorrelación parcial de la conversión del reactor, presentada anteriormente en la Sección 4.2.1.1 (Figura 4.6).

En esta ocasión, para optimizar el número de variables se emplea el método de sensibilidades presentado en la Sección 2.3.1, esto debido a que ahora no se requiere conocer el orden del modelo para corroborar la controlabilidad, como ocurre en la propuesta expuesta en el Capítulo 4. A su vez, cabe destacar que los modelos locales planteados en (5.9) son lineales en los parámetros, por lo que para identificar los parámetros de las consecuencias es posible emplear la solución directa expuesta en la Sección 2.5.2, y en consecuencia no hay necesidad de utilizar un algoritmo de optimización. Debido a esto, la solución del problema de identificación de los parámetros de las consecuencias se obtiene de manera más rápida, por lo que es posible aumentar la complejidad de los modelos, es decir, aumentar la cantidad de reglas, sin incurrir en un costo computacional demasiado alto.



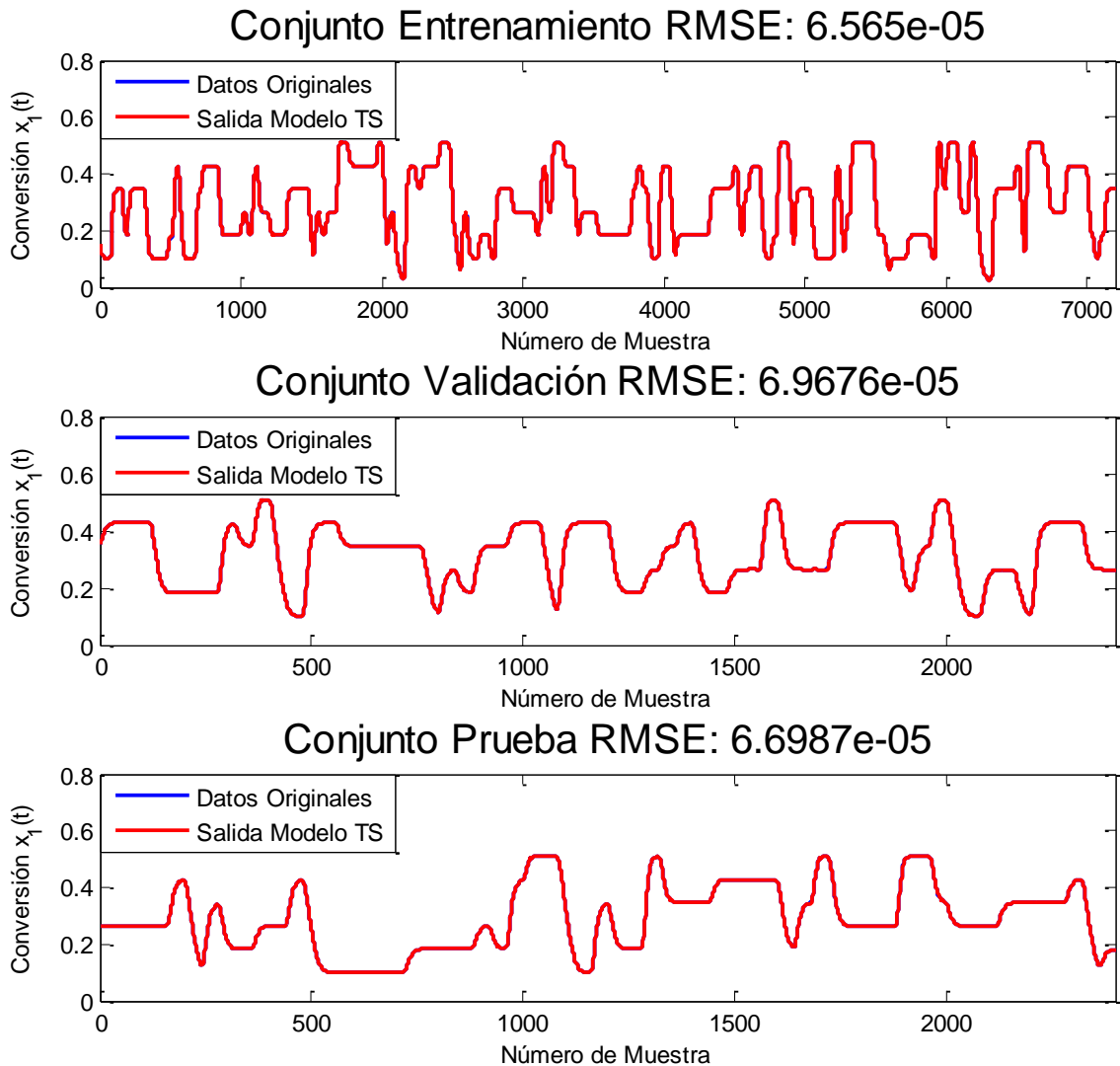
A modo de comparar los procesos de identificación, se genera un modelo con términos de primer orden solamente, cuyas premisas son las mismas que las expuestas en (5.10), descartando los términos de segundo orden. Con el mismo objetivo, esta vez se utiliza un número máximo de reglas igual a 10.

Los resultados de la prueba de identificación se presentan en la Tabla 5.1, donde se observa que la identificación utilizando términos de segundo orden permite un mejor ajuste ya que obtiene índices de desempeño mejores. A su vez, se destaca el hecho de que la cantidad de reglas óptimas al incluir términos cuadráticos se reduce, debido a que los modelos locales son de orden superior y poseen una mayor cantidad de parámetros, los cuales permiten representar el comportamiento de la planta en un rango de validez mayor, abarcando una mayor cantidad de puntos de interés.

**Tabla 5.1: Resultados Identificación Difusa Con Términos de 1er y 2do Orden.**

	<b>Identificación con Términos de 1er Orden</b>	<b>Identificación con Términos de 2do Orden</b>
<b>Número de Reglas</b>	8	4
<b>Variables óptimas</b>	$x_1(k-1), \dots, x_1(k-5)$ $x_2(k-1), \dots, x_2(k-5)$ $\Delta u(k-1), \dots, \Delta u(k-5)$	$x_1(k-1), \dots, x_1(k-5)$ $x_2(k-1), \dots, x_2(k-5)$ $\Delta u(k-1), \dots, \Delta u(k-5)$ $x_1^2(k-1), \dots, x_1^2(k-5)$ $x_2^2(k-1), \dots, x_2^2(k-5)$ $\Delta u^2(k-1), \dots, \Delta u^2(k-5)$
<b>RMSE Entrenamiento</b>	6.6908e-05	6.5650e-05
<b>RMSE Validación</b>	<b>7.0663e-05</b>	<b>6.9676e-05</b>
<b>RMSE Prueba</b>	6.7780e-05	6.6987e-05

En la Figura 5.2 se presenta el resultado del modelo difuso con términos de segundo orden en sus consecuencias, donde se observan resultados de la misma calidad que los obtenidos por el método propuesto en el Capítulo 4, por lo que es posible conseguir modelos con bajos errores de predicción utilizando tanto información de la entrada y los estados del sistema, como información de la entrada y la salida solamente.



**Figura 5.2: Resultado Mejor Modelo Identificación Difusa con Términos de Segundo Orden del CSTR. Arriba: Conjunto de Entrenamiento. Medio: Conjunto de Validación. Abajo: Conjunto de Prueba.**

La Tabla 5.2 y la Tabla 5.3 presentan los parámetros del mejor modelo difuso obtenido utilizando términos de segundo orden en los consecuentes, donde se aprecian grandes diferencias en el orden de los parámetros debido a los rangos en que trabaja cada una de las variables, donde en particular las premisas  $x_2^2(k-j)$  y  $\Delta u^2(k-j)$  tienen asociados valores bastante elevados debido a que  $x_2(k-j)$  y  $\Delta u(k-j)$  pueden tomar valores muy superiores a uno.

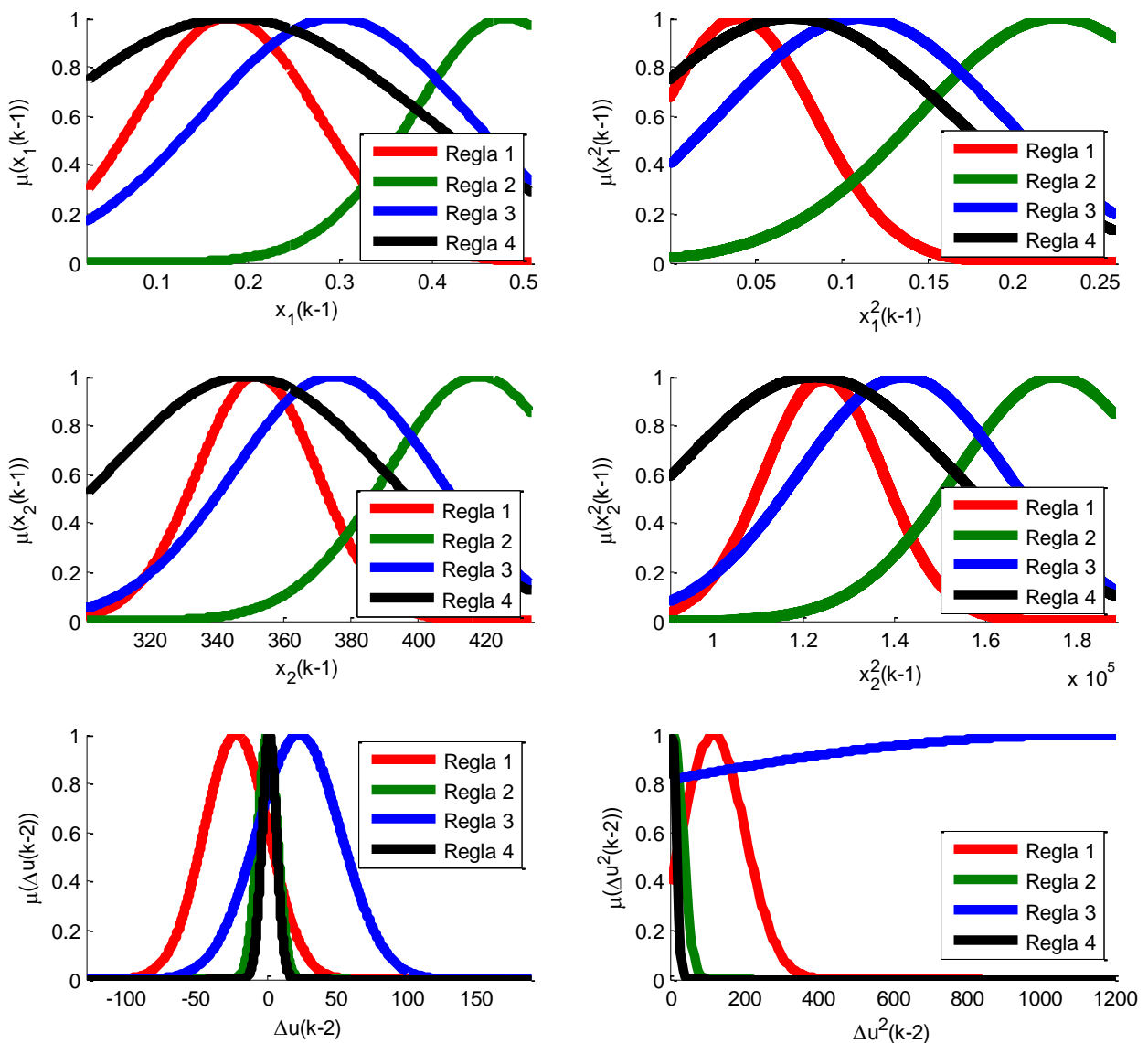
**Tabla 5.2: Parámetros Mejor Modelo Identificación Difusa con Términos de Segundo Orden del CSTR: Regla 1 y Regla 2.**

<b>Parámetros Premisas</b>	$R_1$	$a_{1,i}$	[10.1090, 10.1630, 10.2251, 10.2837, 10.3322, 0.0553, 0.0551, 0.0549, 0.0547, 0.0547, 0.0563, 0.0427, 0.0432, 0.0465, 4.4125e-05, 22.3715, 22.4339, 22.5444, 22.6369, 22.7064, 7.661e-05, 7.639e-05, 7.615e-05, 7.585e-05, 7.568e-05, 0.0175, 0.0117, 0.0086, 0.0075, 7.7140e-05]
		$b_{1,i}$	[0.1772, 0.1767, 0.1757, 0.1743, 0.1730, 352.0819, 352.0108, 351.9582, 351.8583, 351.7919, -11.2401, -22.5868, -27.6586, -28.8755, 189.6639, 0.0402, 0.0398, 0.0393, 0.0387, 0.0384, 1.243e+05, 1.242e+05, 1.242e+05, 1.241e+05, 1.241e+05, 71,9725, 117.0155, 170.2503, 199.1727, 1.872e+04]
	$R_2$	$a_{2,i}$	[9.7670, 9.7508, 9.6307, 9.4601, 9.3509, 0.0365, 0.0367, 0.0368, 0.0369, 0.0369, 0.1651, 0.1429, 0.1167, 0.1018, 0.0188, 12.5095, 12.4701, 12.4447, 12.4336, 12.4349, 4.577e-05, 4.593e-05, 4.608e-05, 4.619e-05, 4.626e-05, 0.0638, 0.0392, 0.0350, 0.0324, 0.0207]
		$b_{2,i}$	[0.4824, 0.4820, 0.4812, 0.4800, 0.4788, 418.3089, 418.3337, 418.3547, 418.3688, 418.2790, 0.0467, -0.1290, -0.3129, -1.2820, 13.9716, 0.2254, 0.2250, 0.2247, 0.2245, 0.2243, 1.753e+05, 1.753e+05, 1.753e+05, 1.753e+05, 1.752e+05, 7.4406, 9.4919, 15.8505, 21.8273, 30.4151]
<b>Parámetros Consecuencias</b>	$R_1$	$\theta_1^T = [3.6907, -5.2373, 3.5099, -1.0709, 0.1078, -6.334e-04, 0.0013, -0.0011, 5.530e-04, -1.7810e-04, 8.01e-06, -7.26e-07, 1.855e-07, -7.162e-07, -2.193e-07, -0.3397, 0.9056, -0.8097, 0.2476, -0.0046, 1.003e-06, -2.206e-06, 1.926e-06, -1.0299e-06, 3.073e-07, -3.037e-07, 2.094e-08, 1.224e-08, -6.666e-09, 1.969e-09]$	
	$R_2$	$\theta_2^T = [4.4578, -7.1105, 4.8557, -1.2283, 0.0167, 0.0019, -0.0051, 0.0032, 6.759e-04, -6.386e-04, 1.083e-05, -2.908e-05, -1.043e-05, -1.562e-06, 1.956e-06, -2.0180, 5.8217, -5.8196, 2.2671, -0.2411, -1.211e-06, 4.46e-06, -3.430e-06, -6.783e-07, 8.458e-07, -1.009e-07, -4.168e-07, 3.3922e-07, 3.905e-08, -3.481e-07]$	

**Tabla 5.3: Parámetros Mejor Modelo Identificación Difusa con Términos de Segundo Orden del CSTR: Regla 3 y Regla 4.**

<b>Parámetros Premisas</b>	$R_3$	$a_{3,i}$	[6.9015, 6.9424, 6.9830, 7.0219, 7.0561, 0.0329, 0.0329, 0.0328, 0.0326, 0.0326, 0.0622, 0.0332, 0.0234, 0.0232, 0.0578, 12.1947, 12.2611, 12.3258, 12.3812, 12.4296, 4.390e-05, 4.381e-05, 4.363e-05, 4.340e-05, 4.330, 8.185e-04, 5.509e-04, 1.971e-04, 1.293e-04, 0.0045]
		$b_{3,i}$	[0.2955, 0.2951, 0.2946, 0.2939, 0.2930, 375.1511, 375.1447, 375.1601, 375.1923, 375.2095, 9.9029, 22.5294, 129.6343, 130.9438, 7.3925, 0.1113, 0.1108, 0.1103, 0.1097, 0.1090, 1.417e+05, 1.417e+05, 1.417e+05, 1.418e+05, 1.418e+05, 3.788e-11, 1.162e+03, 1.536e+04, 2.339e+04, 3.788e-11]
	$R_4$	$a_{4,i}$	[4.8137, 4.9696, 5.1169, 5.2522, 5.3734, 0.0241, 0.0247, 0.0252, 0.0257, 0.0259, 0.0150, 0.1931, 0.2555, 0.2615, 0.2585, 10.7410, 10.9587, 11.1511, 11.3182, 11.4947, 3.238e-05, 3.322e-05, 3.408e-05, 3.472e-05, 3.521e-05, 1.405e-04, 0.0910, 0.0756, 0.0680, 0.0685]
		$b_{4,i}$	[0.1833, 0.1921, 0.1994, 0.2055, 0.2104, 349.0271, 350.6792, 352.2882, 353.5346, 354.5170, 132.9384, 1.0847, 0.0512, 0.0130, 0.1953, 0.0710, 0.0728, 0.0742, 0.0752, 0.0761, 1.222e+05, 1.236e+05, 1.249e+05, 1.258e+05, 1.266e+05, 1.423e+04, 6.9409, 1.8941, 0.0257, 3.788e-11]
<b>Parámetros Consecuencias</b>	$R_3$	$\theta_3^T = [ 3.1738, -3.6759, 1.9339, -0.5280, 0.0971, -2.254e-04, -4.563e-05, 4.732e-04, -3.455e-05, -1.711e-04, 1.42e-05, 2.462e-07, 4.57e-07, -7.72e-08, -8.805e-07, -0.4249, 1.0188, -0.8845, 0.3546, -0.0671, 4.196e-07, -1.224e-07, -5.401e-07, 3.007e-08, 2.225e-07, -6.898e-08, 1.7532e-08, 1.2194e-08, 3.531e-09, 6.943e-08]$	
	$R_4$	$\theta_4^T = [ 5.1015, -9.6185, 8.3938, -3.3425, 0.4654, -8.610e-04, 0.0031, -0.0022, -4.250e-04, 3.757e-04, 1.2093e-05, 9.596e-06, 1.078e-05, -1.270e-05, 9.640e-07, -0.8870, 3.3460, -4.2557, 2.0749, -0.2780, 7.416e-07, -4.139e-06, 4.398e-06, -9.948e-07, -5.006e-09, 1.330e-08, 3.767e-07, -2.422e-07, -3.796e-07, -2.164e-07]$	

En la Figura 5.3 se ilustran algunos conjuntos difusos asociados a las premisas del modelo difuso con términos de segundo orden óptimo, utilizando los parámetros presentes en la Tabla 5.2 y la Tabla 5.3. Se destaca en particular que mientras mayor es la complejidad en la estructura, ya sea mediante una mayor cantidad de reglas o una mayor cantidad de variables de entrada, es más difícil realizar un análisis respecto a las reglas que conforman el modelo, ya que mientras en una variable puede que una regla sea redundante, en otra variable dichas reglas se pueden complementar, abarcando distintas condiciones de operación de la planta. A modo de ejemplo se observan que las funciones de pertenencia de la Regla 1 y Regla 4 en  $x_1(k-1)$  se sobrepone, mientras que en  $\Delta u(k-2)$  los conjuntos difusos asociados a las mismas reglas mencionadas se complementan.



**Figura 5.3: Conjuntos Difusos de Algunas Premisas del el Mejor Modelo Identificación Difusa con Términos de Segundo Orden del CSTR.**

Luego de obtenido el modelo de predicción difuso con consecuencias con términos de segundo orden, se procede al diseño del controlador predictivo difuso para validar la estrategia propuesta.

### 5.3.2. Control Predictivo Difuso del CSTR

Debido a que se desea comparar el desempeño del modelo de predicción difuso con términos de segundo orden con respecto a aproximaciones de segundo orden realizadas en cada iteración, el problema de optimización del controlador predictivo esta dado por

$$\begin{aligned} \min_{\Delta u(k)} J(z(k), u(k)) &= Q \left( r(k+1) - y_{fuzzy}(k+1|z(k)) \right)^2 + \Delta u^2(k), \\ \text{s. a. } y_{fuzzy}(k+1|z(k)) &= \sum_{r=1}^{N_r} h_r(z(k+1)) \cdot \theta_r^T z(k+1), \\ u(k) &\in [300, 490] [K], \end{aligned} \quad (5.11)$$

donde  $r(k+1)$  es la referencia en el instante  $k+1$  (recordar que para el CSTR  $r(k+1) = r = 0.5088$ );  $Q > 0$  es un parámetro de sintonización del controlador;  $y_{fuzzy}(k+1|z(k))$  es la predicción a un paso realizada por el modelo difuso;  $z(k+1)$  son las premisas del modelo difuso presentes en la Tabla 5.1; y  $\Delta u(k)$  es la variación de la acción de control en el instante  $k$ . Se emplea el *solver* SQP para resolver el problema de optimización planteado en (5.11).

El controlador se sintoniza a partir del procedimiento presentado en la Sección 8.2.2 de anexos, utilizando el ITAE, buscando reducir el error en estado estacionario  $e_{ss}$ . Los resultados obtenidos utilizando el control predictivo con modelos de predicción difuso, empleando tanto términos de primer orden como de segundo orden, se presentan en la Tabla 5.4.

**Tabla 5.4: Resultados Control Predictivo Difuso del CSTR con Modelos de Predicción Difusos con Términos de 1er y 2do Orden.**

<i>Modelo de Predicción</i>	<b>Control Predictivo Difuso</b>		
	$Q$	$e_{ss}$	$u^T(k)u(k)$
<b>TS con Términos de 1er Orden</b>	$6.6613 \cdot 10^5$	5.6012e-04 (0.1101%)	2.2175e+07
<b>TS con Términos de 2do Orden</b>	$5 \cdot 10^5$	<b>1.9189e-04</b> <b>(0.0377%)</b>	2.1787e+07

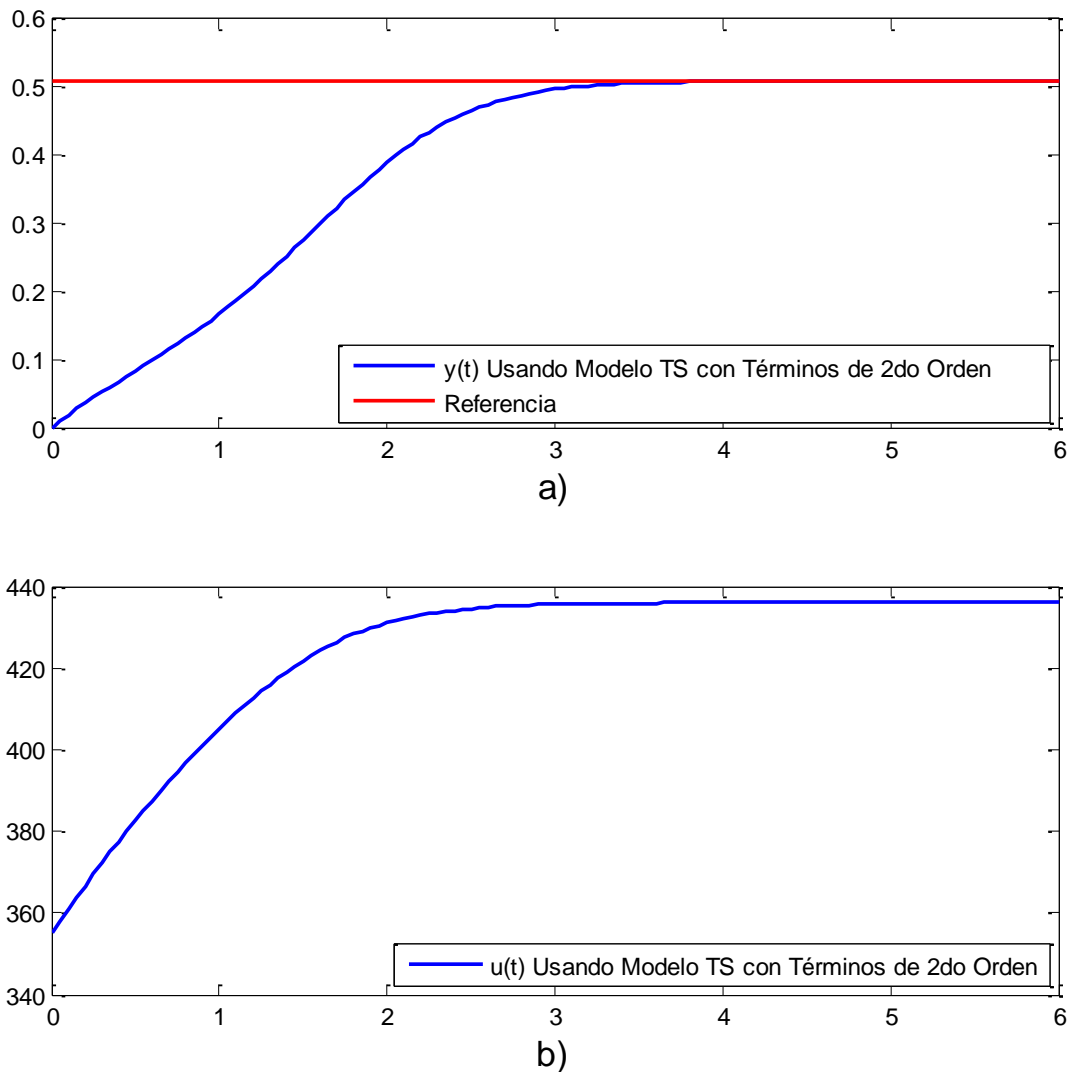
De los resultados obtenidos se observa que el modelo difuso con términos de segundo orden, aparte de reducir el número de reglas en el proceso de identificación, también

permite obtener un mejor desempeño en lazo cerrado, alcanzando un error en estado estacionario menor a la mitad del error obtenido utilizando el modelo difuso con términos de primer orden como predictor. A su vez, se destaca que el error en estado estacionario obtenido por el control predictivo utilizando el modelo difuso con términos de segundo orden es menor comparado con los resultados presentados en la Sección 4.2.1.2 (ver Tabla 4.10/Tabla 4.11), debido a que los términos de mayor orden presentes en el modelo de predicción permiten realizar una buena aproximación del comportamiento de la planta, y en consecuencia se generan acciones de control precisas, las cuales permiten alcanzar la referencia deseada con una buena exactitud. Adicionalmente, y a diferencia de las estrategias propuestas en la Sección 4.2.1.2, el controlador predictivo diseñado considera la variación de la acción de control  $\Delta u(k)$ , por lo que se tiene una acción integral a la salida del controlador, lo que ayuda a reducir el error en estado estacionario cuando se tienen referencias tipo escalón.

Con respecto al esfuerzo de control, utilizar términos de orden superior permite minimizar la energía de control en comparación a utilizar términos de primer orden solamente, debido a que los términos asociados principalmente a  $\Delta u^2(k-j)$  otorgan mayor peso a la acción de control en la función objetivo del control predictivo, obteniendo una respuesta mejor.

En la Figura 5.4 a) se ilustra la respuesta del reactor al utilizar el controlador predictivo con el modelo de predicción difuso con términos de segundo orden, donde se observa el buen comportamiento de la planta, alcanzando el nivel de conversión deseado. Por otro lado, la acción de control se presenta en la Figura 5.4 b), donde se aprecia una respuesta suave en comparación a los resultados obtenidos en la Sección 4.2.1.2, debido a que el controlador diseñado busca minimizar la variación de la acción de control,  $\Delta u(k)$ , y no directamente la acción de control  $u(k)$  como en la propuesta planteada en el Capítulo 4.

Una vez diseñado el controlador predictivo difuso y verificado su correcto funcionamiento, se procede a realizar pruebas de robustez frente a perturbaciones, con el fin de comparar la respuesta del control predictivo al utilizar dos modelos de predicción distintos: uno utilizando aproximaciones de segundo orden, y otro utilizando el modelo difuso con términos de segundo orden. Para estas pruebas se consideran perturbaciones en la entrada y en la salida que emulen el ruido asociado a los sensores que se pueden encontrar en la realidad. En este caso se añaden componentes aleatorias uniformemente distribuidas a cada variable, considerando una perturbación de amplitud de 5 para la variable manipulada, mientras que se emplea una perturbación de amplitud 0.025 para la variable controlada.

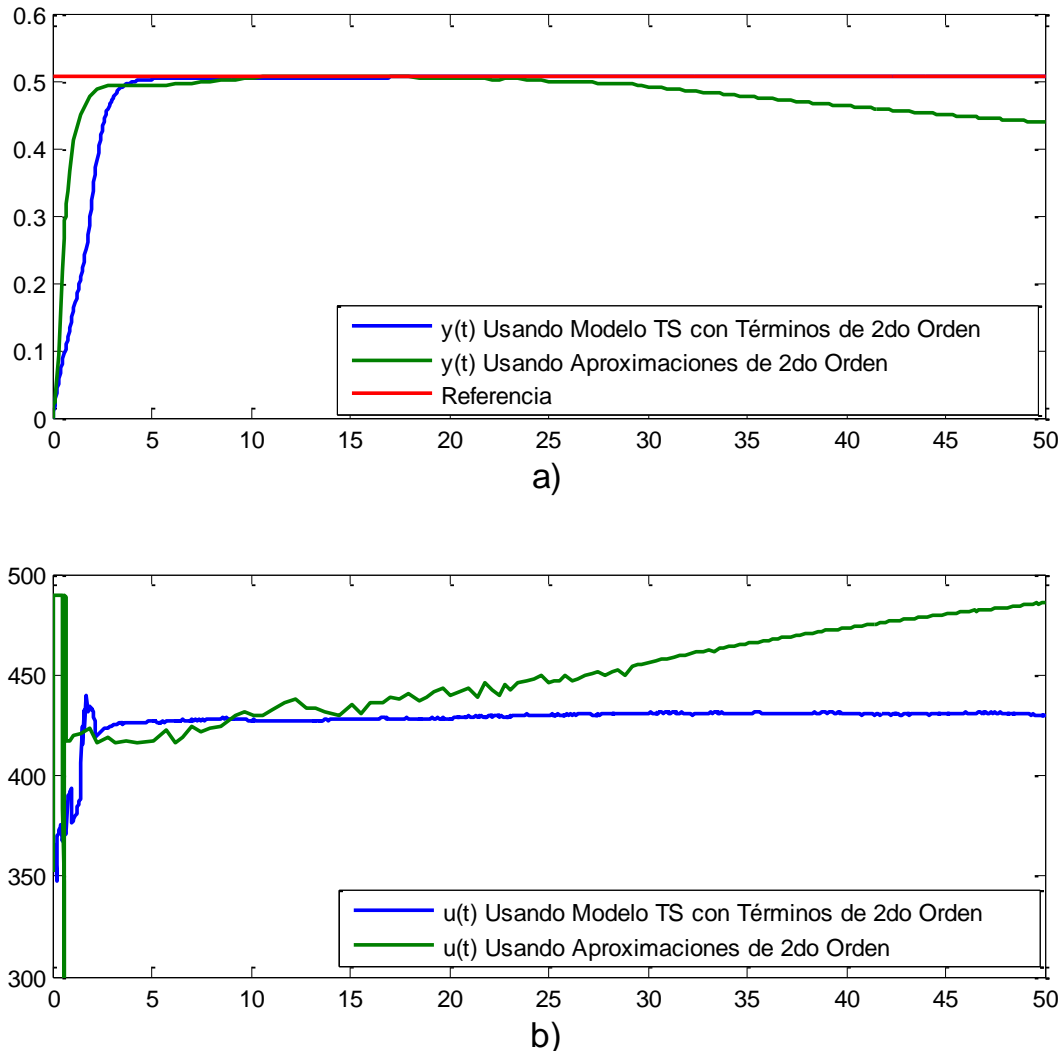


**Figura 5.4: Resultados Control Predictivo Difuso del CSTR Usando Modelo de Predicción Difuso con Términos de 2do Orden. a) Salida  $y(t)$  vs Tiempo [s]. b) Entrada  $u(t)$  [K] vs Tiempo [s].**

En la Figura 5.5 se observan los resultados obtenidos de la prueba de robustez, donde se destaca que a pesar de las perturbaciones incluidas, el modelo difuso con términos de segundo orden permite un control adecuado de la planta, manteniendo la salida en la referencia deseada debido a que considera información de distintos puntos de operación. Por otro lado, el controlador que utiliza aproximaciones de segundo orden para predecir el comportamiento de la planta no logra mantener el sistema en el punto deseado, debido a que las perturbaciones sacan al sistema del punto de operación, impidiendo predecir adecuadamente el comportamiento del proceso, afectando el desempeño en lazo cerrado. A su vez, la Figura 5.5 b) ilustra las acciones de control, donde se observa una respuesta menos suave que la ilustrada en la Figura 5.4, efecto producido por las perturbaciones presentes en el sistema. Sin embargo, el controlador predictivo difuso logra contrarrestar dicho efecto, convergiendo en una zona óptima, mientras que el



controlador predictivo utilizando aproximaciones de segundo orden no permite el mismo desempeño, comenzando a aumentar el valor de  $u(k)$ , afectando la respuesta de la planta.



**Figura 5.5: Resultados Control Predictivo Difuso del CSTR Frente a Perturbaciones Usando Modelo de Predicción con Aproximaciones de 2do Orden y Modelo de Predicción Difuso con Términos de 2do Orden. a) Salida  $y(t)$  vs Tiempo [s]. b) Entrada  $u(t)$  [K] vs Tiempo [s].**

Las propuestas de identificación difusa presentes en el Capítulo 4 y 5 logran buenos resultados en lazo cerrado, siendo ambas metodologías una buena opción para abordar el problema del espacio de controlabilidad que se puede encontrar en algunos modelos dinámicos. Adicionalmente, debido a que el controlador diseñado utilizando un modelo de predicción difuso con términos de segundo orden en sus consecuencias posee acción integral, es posible obtener un error en estado estacionario menor en comparación a la estrategia de control planteada en 4.2.1.2, donde se emplean modelos difusos con

consecuencias lineales y un control sin acción integral. A su vez, el controlador con el modelo difuso con términos de segundo orden conlleva un menor esfuerzo de control y, dado a que en dicho caso se busca minimizar  $\Delta u(k)$ , se obtiene una acción de control mucho más suave en comparación a los resultados reportados en la Sección 4.2, por lo que al menos en los casos de estudio analizados se tiene que incorporar términos de orden superior en las consecuencias del modelo difuso conduce a una mejora en el rendimiento del lazo de control completo, alcanzando errores menores con un menor esfuerzo de control.

Si bien en este trabajo se diseñan estrategias de control predictivo que permiten obtener un buen desempeño en lazo cerrado, para poder implementar este tipo de metodología en la realidad es necesario tener en consideración el tiempo necesario para resolver el problema de optimización del controlador predictivo, de tal modo de obtener una acción de control antes del tiempo de muestreo utilizado para realizar mediciones en el proceso. Dicho esto, en la Sección 8.2.3 de anexos se presenta algunas consideraciones para implementar un controlador predictivo en la práctica.

## 6. Conclusiones

En este trabajo se proponen dos nuevas metodologías de identificación difusa: una donde se incluyen restricciones al momento de identificar los parámetros de las consecuencias, con el fin de obtener modelos difusos controlables en los puntos de operación de interés; y otra donde se incorporan aproximaciones de segundo orden en los consecuentes del modelo, con el propósito de aumentar el espacio de controlabilidad. En particular, se muestra mediante casos de estudio que el uso de estos tipos de modelos permite diseñar estrategias de control predictivo con las cuales se obtiene un buen desempeño en lazo cerrado, alcanzando bajos errores en estado estacionario.

Este desarrollo (y en general cualquier proceso de identificación) está fuertemente ligado a los datos de identificación utilizados, ya que las conclusiones acerca de los modelos obtenidos a partir del procedimiento se basan en ellos, por lo que se debe dar especial énfasis a un diseño apropiado de un experimento de identificación para la obtención de datos de entrada-salida que representen adecuadamente los comportamientos de la planta que se quieran modelar. En particular, las estrategias de identificación aquí planteadas requieren que la planta original sea controlable, o al menos accesible en las condiciones de operación que se deseen identificar, ya que en caso contrario no se podría manipular la entrada del sistema tal de tomar datos de los estados deseados, debido a la no controlabilidad del sistema, imposibilitando la obtención de un modelo de predicción que represente correctamente la planta en las condiciones de operación requeridas. Luego, es importante definir bien las dinámicas de interés de la planta, analizar si es posible obtener una acción de control que permita alcanzar dichos comportamientos deseados, y diseñar una estrategia que permita calcular dichas acciones de control.

Adicionalmente, se enfatiza la ventaja de utilizar datos en lazo cerrado para la identificación de sistemas con el fin de diseñar estrategias de control basadas en modelos, esto porque el modelo entrenado asimila mejor la relación entre la entrada y la salida de la planta al operar en lazo cerrado en comparación a modelos entrenados con datos en lazo abierto, dándole mayor énfasis a las frecuencias que importan en la operación en lazo cerrado, y por lo tanto se logra una mejor predicción y control del proceso. Cabe destacar que las frecuencias bajas toman mayor importancia que las frecuencias altas al momento de identificar un modelo de predicción a  $N$  pasos, y es por esto que algunos autores sugieren filtrar los datos antes de iniciar el procedimiento de identificación [94], lo cual podría mejorar los resultados obtenidos en este trabajo.

La propuesta descrita en el Capítulo 4, donde se incluyen restricciones de controlabilidad en el procedimiento de identificación, utiliza una clase particular de modelos difusos, definida a partir de funciones de pertenencia gaussianas y modelos locales lineales. No obstante, dicha metodología puede ser extendida a otra clase de modelos difusos, cambiando la estructura antes mencionada, y respetando que las funciones a utilizar sean diferenciables para poder aproximar la salida del modelo difuso a través de un modelo de primer orden. Dado lo anterior, en dicho caso se deben derivar

nuevamente las expresiones necesarias para formular la aproximación de primer orden, de manera similar a lo expuesto en la Sección 4.1.

A su vez, el uso de funciones objetivo penalizadas como la presentada en (4.12)-(4.13) puede ser extendido para incluir restricciones distintas a las de controlabilidad, tales como restricciones de estabilidad de sistemas difusos (utilizando lo expuesto en la Sección 8.5 de anexos) o algún tipo de restricciones físicas que poseen los procesos reales. De esta manera, utilizar este tipo de estrategia en la identificación de sistemas puede mejorar la representatividad de los modelos obtenidos, ya que permite que éstos hereden las principales propiedades y características del proceso original.

Otro aspecto a considerar es que realizar una identificación a uno o  $N$  pasos no necesariamente retorna los mismos modelos. Si bien un modelo de predicción óptimo a un paso minimiza el error en el primer instante de tiempo, luego dicho error se propaga, generando un sobreajuste en el modelo y empeorando su desempeño para horizontes de predicción mayores; por otro lado, un modelo de predicción óptimo a  $N$  pasos busca minimizar el error acumulado durante todo el horizonte de predicción, por lo que puede obtener un desempeño peor al predecir a un paso en comparación al desempeño de un modelo entrenado para minimizar el error en el primer instante, sin embargo, dicho error no aumenta considerablemente durante un horizonte de predicción mayor, retornando mejores resultados en dicho escenario. Por lo anterior, es necesario tener en cuenta el propósito del modelo a obtener, y utilizar el tipo de identificación que mejor se adapte a tales objetivos. En particular, en este trabajo se utilizan los modelos difusos para ser aplicados a estrategias de control predictivo, por lo que realizar una identificación a más pasos es más acertado para obtener modelos que mejor predigan el comportamiento de la planta y, en consecuencia, inferir acciones de control adecuadas.

Adicionalmente, la mayoría de los modelos óptimos encontrados en los casos de estudio utilizan un mayor número de regresores que la planta original, aumentando el orden de los modelos difusos en comparación al proceso. La inclusión de estados extras permite una mejor identificación del sistema y en consecuencia se obtiene una mejor representación de las no linealidades presentes en los datos de identificación, sin embargo, se debe tener el cuidado de evitar un sobreajuste a los datos de entrenamiento, por lo que se sugiere emplear validación cruzada u otra metodología para descartar tales soluciones.

Cabe destacar que los modelos difusos empleados son de tiempo discreto, mientras que los procesos originales son de tiempo continuo; luego, dicha discretización debe ser considerada en el estudio de las propiedades del modelo, ya que podría ser la causante de la pérdida de estabilidad y/o controlabilidad, u otra característica del sistema original. A su vez, se puede generar pérdida de información en caso de que no se respete el tiempo de muestreo de Shannon, ya que en dicho caso no se podría reconstruir fielmente las señales de la planta, generando modelos erróneos.

Como se menciona en [78], la mayoría de los procesos físicos son controlables, mas su representación matemática no necesariamente posee dicha propiedad. A modo de

ejemplo, es posible obtener modelos de predicción a un paso del CSTR que sólo sean autorregresivos (y en consecuencia los modelos obtenidos son completamente no controlables) y que tengan buenos desempeños como predictores, como se observa en la Figura 8.3 en la Sección 8.3.2 de anexos; y como el modelo no depende explícitamente de la variable manipulada no se podrían diseñar estrategias de control. Caso análogo se tiene con la estabilidad de los sistemas, donde algunos de los modelos obtenidos en la Sección 4.2 no poseen dicha propiedad a pesar de que la planta original es estable, lo cual refuerza el hecho de que un modelo no necesariamente hereda las características del sistema original.

Ocasionalmente los modelos de primer orden pueden resultar restrictivos desde el punto de vista de control, incluso en la vecindad del punto de operación en el que son calculados, lo cual genera que el espacio de controlabilidad del sistema se vea reducido. Dado lo anterior, la propuesta de utilizar aproximaciones de orden superior permite suplir esta desventaja, aumentando el espacio controlable del sistema, y generando mejores resultados en lazo cerrado. Si bien utilizar aproximaciones de mayor orden permite predecir de mejor manera el comportamiento de la planta, emplear este tipo de modelos requiere del uso de una mayor cantidad de parámetros, lo cual puede afectar en el esfuerzo computacional necesario para resolver el problema de optimización planteado por el control predictivo. A su vez, se destaca que el uso de aproximaciones punto a punto como modelos de predicción en un controlador predictivo puede generar una pérdida de desempeño al momento de afrontar perturbaciones externas, debido a que el ruido puede sacar al sistema del punto de operación donde la aproximación es computada, generando una mala predicción y en consecuencia afectando el rendimiento del controlador. Por otro lado, los modelos difusos presentan aproximaciones en distintos puntos de operación, por lo que si alguna perturbación logra mover al sistema de un punto de operación a otro no se genera una pérdida significativa en el desempeño del modelo como predictor, aumentando la robustez del controlador en estos casos.

En cuanto al diseño de controladores predictivos, se destaca la capacidad para alcanzar los objetivos de control de las estrategias implementadas utilizando los modelos difusos propuestos. A su vez, es importante destacar la condición de controlabilidad al momento de diseñar estrategias de control predictivo. En caso de que el modelo de predicción sea no controlable, pueden existir referencias deseadas que no puedan ser alcanzadas debido a que no existe una entrada  $u(k)$  que permita llevar al sistema al punto de operación deseado, lo cual conduce a que el problema de optimización del control predictivo esté mal planteado, debido a que no existe una solución óptima que permite minimizar el error de seguimiento. Luego, la controlabilidad de los modelos empleados en este tipo de estrategias es un requisito fundamental para obtener un buen desempeño en lazo cerrado.

La principal desventaja de los controladores predictivos recae en el tiempo empleado para determinar la acción de control óptima, por lo que en muchos casos su uso en aplicaciones reales se ve restringido debido a que el costo computacional de resolver el problema de optimización es mayor que el tiempo de muestreo utilizado en la planta. Luego, para evitar esto último es necesario minimizar los cálculos a realizar. Una forma

de lograr esto es aplicando el principio de parsimonia expuesto en la Sección 2.4, con el cual se pueden obtener modelos no tan buenos como el modelo óptimo, pero que sean más simples, requieran menor esfuerzo computacional para calcular predicciones en un horizonte dado, y sigan cumpliendo con su propósito. A su vez, las estrategias planteadas en la Sección 8.2.3 demuestran minimizar el tiempo de cómputo, obteniendo cálculos más eficientes. Por otro lado, en la literatura también se han reportado otro tipo de estrategias para lidiar con el esfuerzo computacional del controlador predictivo. En particular, en [76] se propone la implementación de un control predictivo neuronal por emulación, el cual consiste en modelar el comportamiento del controlador mediante una red neuronal perceptrón multicapa, y reemplazar el controlador por dicha red, de tal forma de que ya no sea necesario calcular una acción de control óptima mediante la minimización de un funcional de costos en cada iteración, disminuyendo el esfuerzo computacional.

## 6.1. Aportes de la Tesis

A partir de lo realizado, los principales aportes de esta tesis se pueden resumir de la siguiente manera:

- **Propuestas de nuevos métodos de identificación:** En el presente trabajo se plantean dos nuevas propuestas para la identificación de modelos difusos de Takagi-Sugeno: inclusión de restricciones de controlabilidad e inclusión de términos de mayor orden en los consecuentes. Dichas propuestas destacan por no ser abordadas en la literatura anteriormente y permitir mejorar el desempeño de los modelos obtenidos desde el punto de vista de la controlabilidad de los sistemas, en comparación a modelos más simples como una aproximación de primer orden.
- **Mejoras en desempeño en lazo cerrado:** Las propuestas planteadas permiten obtener modelos que tengan la propiedad de ser controlables en los rangos de operación de interés, o en su defecto poseen un espacio de controlabilidad mayor al de modelos obtenidos mediante aproximaciones de primer orden, y gracias a ello es posible mejorar el desempeño en lazo cerrado de los procesos estudiados mediante una mejor deducción de acciones de control a la hora de diseñar una estrategia para alcanzar un punto de operación deseado.
- **Toolbox de identificación:** Se mejora el toolbox de identificación de modelos difusos Takagi-Sugeno desarrollado por Núñez et al. [77], donde se incluyen nuevas funciones para abordar las propuestas descritas en este trabajo y se solucionan errores anteriores presentes en la programación. A su vez, se implementan funciones para el diseño de señales de excitación PRBS y APRBS para experimentos de identificación en lazo abierto y lazo cerrado.

## 6.2. Trabajo Futuro

Diversas líneas de investigación pueden ser derivadas a partir del trabajo realizado. A su vez, algunas mejoras pueden ser implementadas de tal manera de optimizar el procedimiento de identificación utilizado.

Dada a la formulación de la función objetivo penalizada para incluir restricciones de controlabilidad, algoritmos de optimización basados en gradiente como Quasi-Newton pueden obtener peor desempeño que otros como Nelder-Mead, esto debido a que la función de penalización,  $\varphi(\cdot)$  en (4.12)-(4.13), sólo considera un valor constante  $\lambda$  para penalizar el caso en que no se cumplen las restricciones, por lo que dicha información, al no depender de las variables de optimización, no es tan relevante para determinar la dirección de decrecimiento en algoritmos como Quasi-Newton. Por lo tanto, para aplicaciones donde algoritmos basados en gradiente no logren converger a una solución aceptable, se podría modificar la función  $\varphi(\cdot)$  tal que ésta dependa de los parámetros a determinar. Dado a que la solución del problema de identificación de los parámetros de las consecuencias sin considerar restricciones de controlabilidad es un buen punto de partida para encontrar una solución óptima, una opción es tomar una penalización del tipo  $\varphi(S_{fuzzy}(z(k), \theta), \theta) = \lambda \exp((\theta - \theta^*)^2)$ , en caso de que  $S_{fuzzy}(z(k), \theta)$  sea singular, donde  $\theta^*$  representa la solución del problema de identificación de los parámetros de las consecuencias sin considerar restricciones de controlabilidad; luego el algoritmo tendería a la región cercana a  $\theta^*$  mientras no se cumplen las restricciones.

A su vez, dada la forma de imponer restricciones sobre el modelo a identificar presente en el Capítulo 4, es posible extender este tipo de identificación difusa a otros casos. En particular, en la literatura se han reportado estrategias de intervalos difusos fundamentadas en los parámetros de las consecuencias del modelo, obteniendo rangos superiores e inferiores de dichos parámetros para representar los límites superior e inferior del intervalo [96]. Por lo tanto, garantizar propiedades como la estabilidad y controlabilidad de los modelos que representan dichos límites puede permitir mejorar el diseño de controladores robustos que empleen intervalos difusos, de tal manera de mejorar el desempeño en lazo cerrado del sistema.

Por otra parte, en la literatura se indica que la identificación de sistemas debe estar enfocada a resolver un propósito inicial, ya sea obtener un modelo para predecir o para diseñar una estrategia de control, y basado en ello se puede obtener una familia de modelos que representen distintas dinámicas del sistema a modelar, yendo desde los más simples a los más complejos [8]. Inspirado en lo anterior, se propone abordar una nueva idea de identificación difusa: incluir modelos locales de distinta complejidad. La idea se fundamenta en que algunos procesos dinámicos pueden poseer rangos de operación donde se comporten como un modelo de un orden determinado, mientras que en otras zonas se puede requerir de un modelo de orden mayor para poder representar de buena manera las dinámicas no lineales del sistema. Luego, planteando modelos locales de distinto orden, y aprovechando la lógica difusa otorgada por los modelos de Takagi-Sugeno, sería posible modelar de manera más eficiente los procesos de la

industria, es decir, obtener un modelo difuso con una menor cantidad de parámetros, pero con igual desempeño que otro que posea una mayor cantidad de parámetros debido a que todos los modelos locales poseen el mismo orden.

Una alternativa que eventualmente podría mejorar los resultados expuestos en este trabajo es la implementación efectiva del método CCM detallado en la Sección 2.4, de tal manera que dentro del procedimiento de identificación se aplique dicha técnica para obtener un número óptimo de *clusters*. Cabe recordar que la idea del algoritmo es evitar tener dos reglas con condiciones de operación similares, por lo tanto, el método evita que la proyección de dos *clusters* se comporte de manera similar sobre todas las variables de entrada, por lo que eventualmente no existiría información redundante en alguna de las reglas del modelo difuso si se utiliza este método.

Debido a que los métodos propuestos en este trabajo abordan principalmente modificar la parte de las consecuencias del modelo difuso, también es de interés estudiar el efecto de modificar la parte de las premisas, a saber, la etapa de *clustering* y la posterior proyección en el espacio de entrada de los conjuntos difusos, de tal manera de optimizar el procedimiento de identificación. En la literatura se han propuesto distintos tipos de *clustering* para obtener las agrupaciones difusas. En particular, se ha planteado el uso del algoritmo Gath-Geva (GG) [2], el cual tiene la ventaja de poder encontrar *clusters* de distinta forma, tamaño y densidad, debido a que modifica la definición de distancia por una forma exponencial, lo cual evita tener restricciones sobre el volumen de las agrupaciones como en GK [10]. La desventaja de este enfoque es que, al poseer una definición de distancia del tipo exponencial, es más sensible a las condiciones iniciales. Luego, se podría aplicar primero GK y utilizar la partición obtenida como condición inicial del algoritmo GG, con el fin de obtener una mejor identificación de los conjuntos difusos y, en consecuencia, una mejor representación de las condiciones de operación de la planta a modelar.

Como se menciona en los alcances de este trabajo, las estrategias estudiadas son validadas en sistemas de una entrada y una salida, por lo que aplicar directamente las propuestas diseñadas a sistemas MIMO no necesariamente derivaría en las mismas conclusiones obtenidas. Por lo tanto, se propone analizar el caso de los modelos de múltiples entradas y múltiples salidas y extender tanto las propuestas como las implementaciones realizadas en el contexto de este trabajo a dichos sistemas. Lo anterior conlleva analizar cómo realizar el procedimiento de identificación, ya que algunos autores presentan métodos donde se requiere que las salidas sean separables, identificando varios subsistemas MISO (*Multiple Input-Single Output*) [31]; mientras que otros proponen realizar el procedimiento de identificación sin la necesidad de separar la planta original en subsistemas [82].

Finalmente, en la literatura se han reportado nuevos métodos para analizar la controlabilidad de los modelos dinámicos no lineales, los cuales se basan en utilizar medidas de matrices [62]. A partir de este concepto matemático se plantean nuevos teoremas para corroborar la controlabilidad, por lo que se propone estudiar en detalle este tipo de estrategias y analizar si es factible ser utilizada en algún algoritmo de



identificación difusa como el presentado en el Capítulo 4. A su vez, otros estudios hacen uso de medidas de matrices para estudiar la controlabilidad robusta de sistemas difusos, otorgando una herramienta más potente para el diseño de estrategias de control en escenarios con incertidumbre o ruido [20][21].

## 6.3. Publicaciones

En el contexto de este trabajo se generaron las siguientes publicaciones:

- Gutiérrez, L.; Valencia, F.; Sáez, D.; Márquez, A., “New Fuzzy Model with Second Order Terms for the Design of a Predictive Control Strategy”, 2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Beijing, China, pp. 2025 – 2031, 2014.
- Sierra, G.; Gutiérrez, L.; Bayas, A., “Application of Genetic Algorithm and Particle Swarm Optimization to a Predictive Control Strategy for a Continuous Stirred Tank Reactor”, X IEEE Escuela de Verano Latino-Americana en Inteligencia Computacional, EVIC, Santiago, 2014. Obtención del 3er Lugar en Concurso de Posters.
- Gutiérrez, L.; Valencia, F.; Sáez, D., “A New Method for Identification of Fuzzy Models with Controllability Constraints”, Enviada para revisión a IEEE Transactions on Fuzzy Systems, 2016.

## 7. Bibliografia

- [1] Abonyi, J.; Babuska, R.; Szeifert, F., “Fuzzy Modeling with Multivariate Membership Functions: Gray-Box Identification and Control Design”, *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics*, vol. 31, no. 5, pp. 755-767, 2001.
- [2] Abonyi, J.; Babuska, R.; Szeifert, F., “Modified Gath-Geva Fuzzy Clustering for Identification of Takagi-Sugeno Fuzzy Models”, *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics*, vol. 32, no. 5, 2002.
- [3] Ahmadi, M., Jafari, A. H. “Is Nonlinear Model Predictive Control with Fuzzy Predictive Model Proper for Managing the Blood Glucose Level in Type I Diabetes?”, *J. Biomedical Science and Engineering*, vol. 5, pp. 63-74, 2012.
- [4] Alimi, S.; Chtourou, M., “Stability Analysis of Fuzzy Identification for Nonlinear Discrete Systems - Part I: Theoretical Study”, *Second International Conference on Developments in eSystems Engineering (DESE)*, pp. 349 – 353, 2009.
- [5] AmirAskari, M.; Menhaj, M.B., “Fuzzy Model Predictive Control Based on Modified Fuzzy Relational Model”, *2013 13th Iranian Conference on Fuzzy Systems (IFSC)*, pp. 1–5, 2013.
- [6] Angelov, P.P.; Filev, D.P., “An Approach to Online Identification of TS Fuzzy Models”, *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 34, no. 1, pp. 484 – 498, 2004.
- [7] Angelov, P.; Xydeas, C.; Filev, D., “On-Line Identification of MIMO Evolving Takagi-Sugeno Fuzzy Models”, *Proceedings. 2004 IEEE International Conference on Fuzzy Systems*, vol. 1, pp. 55-60, 2004.
- [8] Aström, K. J.; Wittenmark, B., “Computer-Controlled Systems: Theory and Design”, Third Edition, Prentice Hall, 1997.
- [9] Ayday, C.T.; Eksin, I., “ Fuzzy Identification of Nonlinear Systems”, *Proceedings of the IECON '93 International Conference on Industrial Electronics, Control, and Instrumentation*, vol.1, pp. 289-292, 1993.
- [10] Babuska, R., “Fuzzy Modelling for Control”, Springer Science, 1998.
- [11] Babuska, R.; van der Veen, P. J.; Kaymak, U., “Improved Covariance Estimation for Gustafson-Kessel Clustering”, *IEEE International Conference on Fuzzy Systems*, pp. 1081-1085, 2002.
- [12] Balasko, B.; Abonyi, J.; Feil, B., “Fuzzy Clustering and Data Analysis Toolbox: For Use with MATLAB”, Department of Process Engineering, University of Veszprem, Hungary, 2005.
- [13] Biglarbegian, M.; Sadeghian, A.; Melek, W. “On the Accessibility/Controllability of Fuzzy Control Systems”, *Information Sciences*, vol. 202, pp. 58–72, 2012.
- [14] Bliselius, P. A., “Application of Pseudo-Random Signals to the Agesta Nuclear Power Station”, *Aktiebolaget Atomenergi*, 1970.
- [15] Bowerman, B. L.; O’Connell, R. T.; Koehler, A. B., “Forecasting, Time Series, and Regression”, 4th ed. Belmont, CA: Thomson Brooks/Cole, 2005.
- [16] Bullock, E., “Sylvester Matrix and Resultants”, *Math 499: Introduction to P-Adic Numbers*, Department of Mathematics, Rice University, 2007.
- [17] Camacho, E.F.; Bordons, C., “Model Predictive Control”, Springer, 2004.

- [18] Chen, G.; Pham, T.T.; Weiss, J. J., “Fuzzy Modeling of Control Systems”, IEEE Transactions on Aerospace and Electronic Systems, vol. 31, no. 1, pp. 414–429, 1995.
- [19] Chen, W.H, “Predictive Control of General Nonlinear Systems Using Approximation”, IEE Proceedings - Control Theory and Applications, vol. 151, no. 2, pp. 137–144, 2004.
- [20] Chen, S.-H.; Ho, W.-H.; Chou, J.-H., “Robust Controllability of T–S Fuzzy-Model-Based Control Systems With Parametric Uncertainties”, IEEE Transactions on Fuzzy Systems, vol. 17, no. 6, pp. 1324–1335, 2009.
- [21] Chen, S.-H.; Ho, W.-H.; Tsai, J.-T.; Chou, J.-H., “Regularity and Controllability Robustness of TS Fuzzy Descriptor Systems with Structured Parametric Uncertainties”, Information Sciences, vol. 277, pp. 36–55, 2014.
- [22] de Bruin, H.A.E.; Roffel, B., “A New Identification Method for Fuzzy Linear Models of Nonlinear Dynamic Systems”, Journal of Process Control, vol. 6, no. 5, pp. 277–293, 1996.
- [23] Djebrani, S.; Abdessemed, F., “Fuzzy Model Based Predictive Control of Nonlinear Systems”, 12th IEEE International Conference on Electronics, Circuits and Systems, ICECS, pp. 1-4, 2005.
- [24] Economou, C. G., “An Operator Theory Approach to Nonlinear Controller Design”, Ph.D. Dissertation, California Institute of Technology, Pasadena, CA, 1985.
- [25] Eksin, I.; Erol, O. K., “A Fuzzy Identification Method for Nonlinear Systems”, Turk J Elec Engin, vol.8, no.2, 2000.
- [26] Fairweather, A.J.; Foster, M.P.; Stone, D.A., “Battery Parameter Identification with Pseudo Random Binary Sequence Excitation (PRBS)”, Journal of Power Sources, vol. 196, no. 22, pp. 9398–9406, 2011.
- [27] Friedland, B., “Control System Design: An Introduction to State-Space Methods”, Dover Publications, 1986.
- [28] Gemmer, J., “Geometric Nonlinear Dynamical Control”, Department of Mathematics, University of Arizona, 2007.
- [29] Gershwin, Stanley B.; Jacobson, D. “A Controllability Theory for Nonlinear Systems”, IEEE Transactions on Automatic Control, vol. 16, no. 1, pp. 37–46, 1971.
- [30] Gomme, P.; Klein, P., “Second-Order Approximation of Dynamic Models Without The Use of Tensors”, Journal of Economic Dynamics and Control, vol. 35, no. 4, pp. 604–615, 2011.
- [31] Grisales, V.H.; Gauthier, A.; Roux, G., “Fuzzy Model Identification of a Biological Process Based on Input-Output Data Clustering”, The 14th IEEE International Conference on Fuzzy Systems, pp. 927 – 932, 2005.
- [32] Gustafson, D.E.; Kessel, W.C., “Fuzzy Clustering with a Fuzzy Covariance Matrix”, IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes, pp. 761 – 766, 1978.
- [33] Haddad, W.M.; Chellaboina, V.S.; Fausz, J.L.; Leonessa, A. “Optimal Nonlinear Robust Control for Nonlinear Uncertain Cascade Systems”, Proceedings of the 1997 American Control Conference, vol. 1, pp. 403-407, 1997.

- [34] Hao Ying, “General Takagi-Sugeno Fuzzy Systems are Universal Approximators”, *Fuzzy Systems Proceedings of the 1998 IEEE World Congress on Computational Intelligence*, vol. 1, pp. 819-823, 1998.
- [35] Hastle, T.; Tibshirani, R.; Friedman, J., “The Elements of Statistical Learning: Data Mining, Inference, and Prediction”, Second Edition, Springer, 2009.
- [36] Hedrick, J. K.; Girard, A., “Control of Nonlinear Dynamic Systems: Theory and Applications”, 2010.
- [37] Hegyi, A., “Model Predictive Control for Integrating Traffic Control Measures”, Ph.D. Dissertation, Delft University of Technology, 2004.
- [38] Hermann, R.; Krener, A., “Nonlinear Controllability and Observability”, *IEEE Transactions on Automatic Control*, vol. AC-22, no. 5, 1977.
- [39] Huang, Y.L.; Lou, H.H.; Gong, J.P.; Edgar, T.F., “Fuzzy Model Predictive Control”, *IEEE Transactions on Fuzzy Systems*, vol. 8, no. 6, pp. 665–678, 2000.
- [40] JAG (Jagatpreet Singh), “PSO Toolbox”, 2003.
- [41] Jakubczyk, B.; Sontag, E. D., “Controllability of Nonlinear Discrete-Time Systems: A Lie-Algebraic Approach”, *SIAM Journal on Control and Optimization*, vol. 28, no. 1, pp. 1–33, 1990.
- [42] Jakubczyk, B., “Introduction to Geometric Nonlinear Control; Controllability and Lie Bracket”, Institute of Mathematics, Polish Academy of Science, Warsaw, Poland, 2001.
- [43] James, M. R., “Controllability and Observability of Nonlinear Systems”, Mathematics Department and Systems Research Center, University of Maryland, College Park, 1987.
- [44] Jin-Hwan Kim; Uk-Youl Huh, “Fuzzy Model Based Predictive Control”, *The 1998 IEEE International Conference on Fuzzy Systems Proceedings, IEEE World Congress on Computational Intelligence*, vol. 1, pp. 405 – 409, 1998.
- [45] Kachroo, P., “Modeling, Analysis and Control of Fuzzy Systems”, *IEEE International Conference on Systems, Man and Cybernetics, 1995. Intelligent Systems for the 21st Century*, vol. 3, pp. 2046 – 2051, 1995.
- [46] Kalman, R. E.; Ho, Y. C.; Narendra, K. S., “Controllability of Linear Dynamical Systems”, *Contributions to Differential Equations*, vol. 1, pp. 189-213, 1963.
- [47] Kar, I.; Majhi, S., “Discrete State Space Models”, *Digital Control Systems, National Programme on Thecnology Enhanced Learning, Indian Institute of Technology Guwahati*.
- [48] Kennedy, J.; Eberhart, R., “Particle Swarm Optimization”, *Proceedings of IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948.
- [49] Khalil, H. K., “Nonlinear Systems”, Third Edition, Prentice Hall, 2002.
- [50] Klamka, J., “Controllability of Nonlinear Discrete Systems”, *International Journal of Applied Mathematics and Computer Science*, vol.12, no.2, pp. 173–180, 2002.
- [51] Kukulj, D.; Levi, E., “Identification of Complex Systems Based on Neural and Takagi-Sugeno Fuzzy Model”, *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, vol. 34, no. 1, 2004.
- [52] Kumar, K. K.; Patwardhan, S. C., “Nonlinear Predictive Control of Systems Exhibiting Input Multiplicities Using the Multimodel Approach”, *Industrial & Engineering Chemistry Research*, vol. 41, pp. 3186-3198, 2002.

- [53] Lee, C. C., “Fuzzy Logic in Control Systems: Fuzzy Logic Controller – Part I”, IEEE Transactions on Systems, Man, and Cybernetics, vol. 20, no. 20, pp. 404 – 418, 1990.
- [54] Lewis, A., “A Brief on Controllability of Nonlinear Systems”, Department of Mathematics and Statistics, Queen’s University, Kingston, 2002.
- [55] Li, C.; Zhou, J.; Fu, B.; Kou, P.; Xiao, J., “T–S Fuzzy Model Identification With a Gravitational Search-Based Hyperplane Clustering Algorithm”, IEEE Transactions on Fuzzy Systems, vol. 20, no. 2, pp. 305-317, 2012.
- [56] Liang Wang; Langari, R., “A Decomposition Approach for Fuzzy Systems Identification”, Proceedings of the 34th IEEE Conference on Decision and Control, vol. 1, pp. 261-266, 1995.
- [57] Lin, P. T.; Wang, C. H.; Lee, T. T., “Time-Optimal Control of T–S Fuzzy Models via Lie Algebra”, IEEE Transactions on Fuzzy Systems, vol. 17, no. 4, pp. 737-749, 2009.
- [58] Ljung, L., “System Identification: Theory for the User”, Second Edition, Prentice Hall, 1999.
- [59] MacVicar-Whelan, P.J., “Fuzzy Sets for Man-Machine Interaction”, International Journal of Man-Machine Studies, vol. 8, no. 6, pp. 687-697, 1976.
- [60] Magnus, J.; Neudecker, H., “Matrix Differential Calculus with Applications in Statistics and Econometrics”, John Wiley and Sons, 1999.
- [61] Mai Van Sy; Phan Xuan Minh, “Fuzzy Model Predictive Control Using Takagi-Sugeno Model”, International Conference on Control, Automation and Systems (ICCAS), pp. 632–637, 2008.
- [62] Maidens, J.; Arcak, M., “Reachability Analysis of Nonlinear Systems Using Matrix Measures”, IEEE Transactions on Automatic Control, vol. 60, no. 1, pp. 265-270, 2015.
- [63] Mamdani, E. H., “Application of Fuzzy Algorithms for Control of Simple Dynamic Plant”, Proceedings of the Institution of Electrical Engineers, vol. 121, no. 12, pp. 1585-1588, 1974.
- [64] Maroto, R., “Ecuaciones para la Sintonización de Controladores PID con Acción Derivativa Aplicada a la Señal Realimentada”, Escuela de Ingeniería Eléctrica, Universidad de Costa Rica, 2007.
- [65] Mathworks, “Autocorrelation and Partial Autocorrelation”, Disponible Online en: <http://www.mathworks.com/help/econ/autocorrelation-and-partial-autocorrelation.html>.
- [66] Mathworks, “Constrained Nonlinear Optimization Algorithms”, Disponible Online en: <http://www.mathworks.com/help/optim/ug/constrained-nonlinear-optimization-algorithms.html>.
- [67] Mathworks, “Divide Data for Optimal Neural Network Training”, Disponible Online en: <http://www.mathworks.com/help/nnet/ug/divide-data-for-optimal-neural-network-training.html>.
- [68] Mathworks, “Unconstrained Nonlinear Optimization Algorithms”, Disponible Online en: <http://www.mathworks.com/help/optim/ug/unconstrained-nonlinear-optimization-algorithms.html>.
- [69] Meda-Campaña, J.A.; Rodríguez-Valdez, J.; Hernández-Cortés, T.; Tapia-Herrera, R.; Nosov, V., “Analysis of the Fuzzy Controllability Property and

- Stabilization for a Class of TS Fuzzy Models”, IEEE Transactions on Fuzzy Systems, vol. 23, no. 2, pp. 291-301, 2015.
- [70] Moler, C., “Numerical Computing with MATLAB”. Other Titles in Applied Mathematics, SIAM, 2004.
- [71] Mozelli, L. A; Palhares, R. M., “Stability Analysis of Takagi-Sugeno Fuzzy Systems via LMI: Methodologies Based on a New Fuzzy Lyapunov Function. Sba Controle & Automação, vol. 22, n.6, pp. 664-676, 2011.
- [72] Mutha, R. K.; Cluett, W. R.; Penlidis, A., “Nonlinear Model-Based Predictive Control of Nonaffine Systems”, Automatica, vol. 33, no. 5, pp. 907–913, 1997.
- [73] Nelles, O.; Isermann, R., “Identification of Nonlinear Dynamic Systems - Classical Methods versus Radial Basis Function Networks”, Proceedings of the 1995 American Control Conference, vol. 5, pp. 3786 – 3790, 1995.
- [74] Nelles, O., “Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models”, Springer, 2001.
- [75] Nijmeijer, H.; van der Schaft, A. J., “Nonlinear Dynamic Control Systems”, Springer-Verlag, 1990.
- [76] Norgaard, M.; Ravn, O.; Poulsen, N. K.; Hansen, L. K., “Neural Networks for Modelling and Control of Dynamic Systems”, Springer-Verlag, 2001.
- [77] Núñez, A.; Sáez, D.; Collado, J.; Ávila, F., “Toolbox de Identificación Difusa Takagi-Sugeno”, Departamento de Ingeniería Eléctrica, Universidad de Chile, 2013.
- [78] Ogata, K., “Ingeniería de Control Moderna”, Tercera Edición, Prentice Hall, 1998.
- [79] Patwardhan, S. C.; Madhavan, K. P., “Nonlinear Model Predictive Control Using Second-Order Model Approximation”, Industrial & Engineering Chemistry Research, vol. 32, no. 2, pp. 334–344, 1993.
- [80] Patwardhan, S. C.; Madhavan, K. P., “Nonlinear Predictive Control of an Exothermic CSTR Using Recursive Quadratic State Space Models”, Proceedings of the 4th IEEE Conference on Control Applications, pp. 967–972, 1995.
- [81] Patwardhan, S. C.; Madhavan, K. P., “Improved Techniques for the Development of Quadratic Perturbation Models”, Industrial & Engineering Chemistry Research, vol. 35, pp. 4281-4290, 1996.
- [82] Pekpe, K.M.; Cassar, J.P.; Chenikher, S., “Identification of MIMO Takagi-Sugeno Model of a Bioreactor”, IEEE International Fuzzy Systems Conference, FUZZ-IEEE, pp. 1-6, 2007.
- [83] Prandini, M; Bittanti, S.; Campi, M. C., “A Penalized Identification Criterion for Securing Controllability in Adaptive Control”, Journal of Mathematical Systems, Estimation, and Control, vol. 8, no. 4, pp. 1-19, 1998.
- [84] Proakis, J. G.; Manolakis, D. G., “Digital Signal Processing: Principles, Algorithms, and Applications”, Third Edition, Prentice-Hall, 1996.
- [85] Qin, H., “On the Controllability of a Nonlinear Control System”, Computers & Mathematics with Applications, vol. 10, no. 6, pp. 441–451, 1984.
- [86] Rajesh, R.; Kaimal, M., “T-S Fuzzy Model with Nonlinear Consequence and PDC Controller for a Class of Nonlinear Control Systems”, Applied Soft Computing, vol. 7, no. 3, pp. 772–762, 2007.

- [87] Rankovic, V.; Nikolic, I., “Model Predictive Control Based on the Takagi-Sugeno Fuzzy Model”, *Journal of Information, Control and Management Systems*, vol. 5, no. 1, pp. 101-110, 2007.
- [88] Rossiter, J. A., “Model-Based Predictive Control: A Practical Approach”, CRC Press, 2003.
- [89] Rumchev, V. G.; Adeane, J., “Reachability and Controllability of Time-Variant Discrete-Time Positive Linear Systems”, *Control and Cybernetics*, vol. 33, no. 1, pp. 85-94, 2004.
- [90] Sáez, D.; Zúñiga, R., “Cluster Optimization for Takagi & Sugeno Fuzzy Models and Its Application to a Combined Cycle Power Plant Boiler”, *Proceedings of the 2004 American Control Conference*, vol. 2, pp. 1776–1781, 2004.
- [91] Salahshoor, K.; Ahangari, I., “Fuzzy Identification of Nonlinear Hybrid Dynamic Systems Using Modified Potential Clustering”, *2011 2nd International Conference on Control, Instrumentation and Automation (ICCIA)*, pp. 1142–1147, 2011.
- [92] Sarimveis, H.; Bafas, G., “Fuzzy Model Predictive Control of Nonlinear Processes Using Genetic Algorithms”, *Fuzzy Sets and Systems*, vol. 139, pp. 59-80, 2003.
- [93] Sheikholeslam, F.; Shekaramiz, M., “Stability Criterion for Takagi-Sugeno Models”, *2011 Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, vol. 1, pp. 261 – 266, 2011.
- [94] Shook, D.S.; Mohtadi, C.; Shah, S.L., “Identification for Long-Range Predictive Control”, *IEE Proceedings-D Control Theory and Applications*, vol. 138, no. 1, pp. 75 – 84, 1991.
- [95] Silverman, L. M.; Meadows, H. E., “Controllability and Observability in Time-Variable Linear Systems”, *SIAM J. Control*, vol. 5, no. 1, pp. 64–73, 1967.
- [96] Skrjanc, I.; Blazic, S.; Agamennoni, O., “Interval Fuzzy Modeling Applied to Wiener Models With Uncertainties”, *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics*, vol. 35, no. 5, pp. 1092-1095, 2005.
- [97] Skrjanc, I.; Blazic, S.; Agamennoni, O., “Identification of Dynamical Systems with a Robust Interval Fuzzy Model”, *Automatica*, vol. 41, no. 2, pp. 327–332, 2005.
- [98] Slotine, J.J.E.; Li, W., “Applied Nonlinear Control”, Prentice-Hall, 1991.
- [99] Soderström, T.; Stoica, P., “System Identification”, Prentice Hall, 1989.
- [100] Solís, J.; Sáez, D.; Estévez, P., “Particle Swarm Optimization-based Fuzzy Predictive Control Strategy”, *2006 IEEE International Conference on Fuzzy Systems*, pp. 1866-1871, 2006.
- [101] Soltani, M.; Chaari, A.; Ben Hmida, F.; Gossa, M., “Modified Fuzzy Model Identification Clustering Algorithm for Liquid Level Process”, *2010 18th Mediterranean Conference on Control & Automation (MED)*, pp. 1151 – 1157, 2010.
- [102] Soltani, M.; Bessaoudi, T.; Chaari, A., “Affine Takagi-Sugeno Fuzzy Model Identification Based on a Novel Fuzzy c-Regression Model Clustering and Particle Swarm Optimization”, *2012 16th IEEE Mediterranean Electrotechnical Conference (MELECON)*, pp. 1067-1070, 2012.
- [103] Soni, Y. K.; Bhatt, R., “BF-PSO Optimized PID Controller Design using ISE, IAE, IATE and MSE Error Criteria”, *International Journal of Advanced Research in Computer Engineering & Technology*, vol. 2, no. 7, 2013.

- [104] Sontag. E. D., “Kalman’s Controllability Rank Condition: From Linear To Nonlinear”, SYCON - Rutgers Center for Systems and Control, Department of Mathematics, Rutgers University, New Brunswick.
- [105] Sousa, J. M.; Babuska, R.; Verbruggen, H.B., “Fuzzy Predictive Control Applied to an Air-Conditioning System”, *Control Engineering Practice*, vol. 5, no. 10, pp. 1395–1406, 1997.
- [106] Sugeno, M.; Yasukawa, T., “A Fuzzy Logic Based Approach to Qualitative Modeling”, *IEEE Transactions on Fuzzy Systems*, vol. 1, no. 1, 1993.
- [107] Takagi, T.; Sugeno, M., “Fuzzy Identification of Systems and Its Applications to Modeling and Control”, *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-15, no.1, pp.116-132, 1985.
- [108] Tanaka, K.; Sugeno, M., “Stability Analysis and Design of Fuzzy Control Systems”, *Fuzzy Sets and Systems*, vol. 45, no. 2, pp. 135-156, 1992.
- [109] Tanaka, K.; Hori, T.; Wang, H.O., “A Multiple Lyapunov Function Approach to Stabilization of Fuzzy Control Systems”, *IEEE Transactions on Fuzzy Systems*, vol. 11, no. 4, pp. 582–589, 2003.
- [110] Teng, F. C.; Tsoi, A. C., “Fuzzy Controller Based on MacVicar-Whelan Matrix”, 1994 IECON, 20th International Conference on Industrial Electronics, Control and Instrumentation, vol. 2, pp. 1343-1348, 1994.
- [111] Teslic, L.; Hartmann, B.; Nelles, O.; Skrjan, I., “Nonlinear System Identification by Gustafson–Kessel Fuzzy Clustering and Supervised Local Model Network Learning for the Drug Absorption Spectra Process”, *IEEE Transactions on Neural Networks*, vol. 22, no. 12, pp. 1941-1951, 2011.
- [112] Van den Bergh, F.; Engelbrecht, A.P., “A New Locally Convergent Particle Swarm Optimiser”, 2002 IEEE International Conference on Systems, Man and Cybernetics, vol. 3, 2002.
- [113] Van den Hof, P., “Closed-Loop Issues in System Identification”, *Annual Reviews in Control*, vol. 22, pp. 173–186, 1998.
- [114] Wang, L.-X., “Fuzzy Systems are Universal Approximators”, *IEEE International Conference on Fuzzy Systems*, pp. 1163-1170, 1992.
- [115] Wen Yu; Xiao Ou Li, “Fuzzy Identification Using Fuzzy Neural Networks With Stable Learning Algorithms”, *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 3, pp. 411–420, 2004.
- [116] Won Young Yang; Wenwu Cao; Tae-Sang Chung; John Morris, “Applied Numerical Methods Using MATLAB”, Wiley Interscience, 2005.
- [117] Xiao Wu; Jiong Shen; Yiguo Li; Lee, K.Y., “Stable Model Predictive Control Based on TS Fuzzy Model with Application to Boiler-Turbine Coordinated System”, 2011 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC), pp. 2981–2987, 2011.
- [118] Yuan-Chang Chang; Shun-Feng Su, “Controller Synthesis for Nonlinear T-S Fuzzy Model via the Control Lyapunov Function Approach”.
- [119] Zadeh, L.A., “Outline of a New Approach to the Analysis of Complex Systems and Decision Processes”, *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-3, no. 1, pp. 28 – 44, 1973.



- 
- [120] Zeng, X-J.; Singh, M. G., “Fuzzy Bounded Least-Squares Method for the Identification of Linear Systems”, *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, vol. 27, no. 5, pp. 624-635, 1997.
  - [121] Zhao, J.; Wertz, V.; Gorez, R., “A Fuzzy Clustering Method for the Identification of Fuzzy Models for Dynamic Systems”, *Proceedings of the 1994 IEEE International Symposium on Intelligent Control*, pp. 172 – 177, 1994.
  - [122] Zhitao Liu; Youyi Wang; Du, J., “Fuzzy Model Predictive Control of a Permanent Magnet Synchronous Motor in Electric Vehicles”, *2013 10th IEEE International Conference on Control and Automation (ICCA)*, pp. 604–608, 2013.
  - [123] Zuheir, A, “Fuzzy Model Predictive Control Algorithm Applied in Nuclear Power Plant”, *Information and Communication Technologies*, 2006. *ICTTA '06*. 2nd, vol. 1, pp. 1410–1415, 2006.

## 8. Anexos

### 8.1. Anexo A: Algoritmos de Optimización

A continuación se detallan los algoritmos utilizados en este trabajo para resolver el problema de optimización planteado en el Capítulo 4, con el fin de obtener modelos difusos controlables.

#### 8.1.1. Método Quasi-Newton

El método Quasi-Newton (QN) es un algoritmo basado en gradiente, por lo que resulta ser un método más eficiente cuando la función objetivo a minimizar,  $f(x)$ , es continua en su primera derivada [68]. Se basa en formular una aproximación cuadrática mediante el uso del gradiente,  $\nabla f(x)$ , y el Hessiano,  $H(f(x))$ , de la función objetivo  $f(x)$ . A diferencia del método de Newton, Quasi-Newton aproxima el Hessiano utilizando observaciones de  $f(x)$  y  $\nabla f(x)$ . Para ello se utiliza la fórmula dada por Broyden, Fletcher, Goldfarb y Shanno (BFGS), la cual resulta ser efectiva para propósitos generales. Dicha fórmula está dada por

$$H(k+1) = H(k) + \frac{q(k)q(k)^T}{q^T(k)s(k)} - \frac{H(k)s(k)s^T(k)H^T(k)}{s^T(k)H(k)s(k)}, \quad (8.1)$$

donde  $s(k) = x(k+1) - x(k)$  y  $q(k) = \nabla f(x(k+1)) - \nabla f(x(k))$ .

Posteriormente se realiza una búsqueda en la dirección  $d(k)$  dada por las siguientes ecuaciones:

$$d(k) = -H^{-1}(k) \cdot \nabla f(x(k)), \quad (8.2)$$

$$x(k+1) = x(k) + \mu \cdot d(k), \quad (8.3)$$

donde  $\mu$  es un escalar que representa el largo del paso. Se itera hasta que se cumpla algún criterio de convergencia o condición de término.

#### 8.1.2. Método Nelder-Mead

El método Nelder-Mead (NM), a diferencia de Quasi-Newton, utiliza sólo evaluaciones de la función objetivo, siendo un algoritmo más apropiado para problemas donde  $f(x)$  no es una función suave o presenta discontinuidades [68].

El algoritmo consiste en los siguientes pasos [116]:

- 1) Tomar tres puntos  $a, b, c$ , donde  $f(a) < f(b) < f(c)$ .
- 2) Si los tres puntos o su evaluación están lo suficientemente cerca, se declara  $a$  como la solución del problema, en otro caso se espera que el mínimo se encuentre en la dirección opuesta del peor punto evaluado,  $c$ .
- 3) Tomar  $e = m + 2(m - c)$ , con  $m = (a + b)/2$ . Si  $f(e) < f(b)$ , reemplazar  $c$  por  $e$ , en otro caso, tomar  $r = 2m - c$  y si  $f(r) < f(c)$ , reemplazar  $c$  por  $r$ . Si  $f(r) \geq f(b)$  entonces tomar  $s = (c + m)/2$ , y si  $f(s) < f(c)$ , reemplazar  $c$  por  $s$ .
- 4) En caso de que no se cumplan ninguna condición en el paso 3), reemplazar  $b$  y  $c$  por  $m$  y  $c_1 = (a + c)/2$  respectivamente. Reflejando que el mínimo podría estar alrededor de  $a$ .
- 5) Volver a iterar.

En la Figura 8.1 se observan los puntos asociados en el algoritmo.

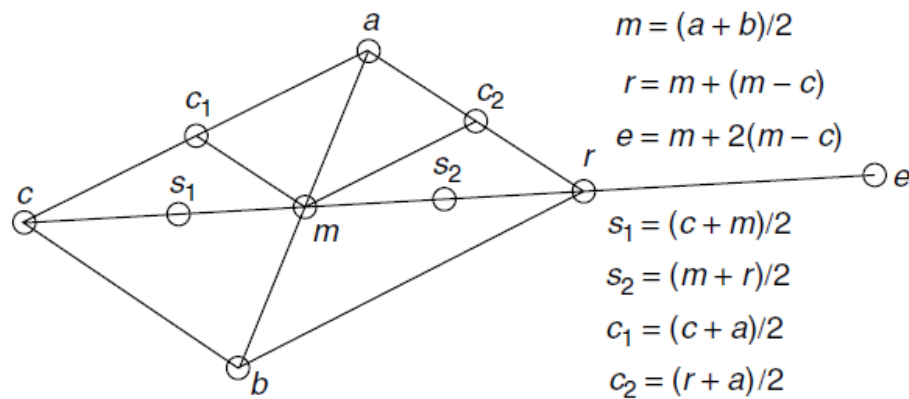


Figura 8.1: Puntos Utilizados en el Algoritmo Nelder-Mead [116].

### 8.1.3. Optimización por Enjambre de Partículas

La optimización por enjambre de partículas (*Particle Swarm Optimization*, PSO) es un algoritmo evolutivo basado en el comportamiento social de agrupaciones en la naturaleza, donde la evolución de los individuos o partículas en el enjambre considera tanto la información referente al historial del individuo mismo, denotando la componente cognitiva; como la información referente al resto de los individuos, denotando la componente social [48].

A diferencia de Quasi-Newton, y similar a Nelder-Mead, PSO no utiliza información del gradiente de la función objetivo, sino que se basa en la evaluación de un conjunto de posibles soluciones. El algoritmo tiene la ventaja de no necesitar información del gradiente, por lo que puede resolver problemas que no sean diferenciables y/o discontinuos. Además al utilizar múltiples soluciones tiene menor probabilidad de quedar atrapado en un óptimo local, sin embargo, no se asegura su convergencia al óptimo global. También requiere de múltiples evaluaciones de la función objetivo, lo que puede llevar a un alto costo computacional si la función a optimizar es compleja.

En el caso de los algoritmos evolutivos suele llamarse función de *fitness* a la función objetivo, donde el *fitness* corresponde a la evaluación de la partícula en dicha función. El algoritmo se detalla en el siguiente pseudo-código:

- 1) Inicializar los parámetros de PSO, como el tamaño del enjambre  $n$ , el número de iteraciones, y los parámetros asociados al algoritmo mismo.
- 2) Inicializar población con posiciones  $p(t)$  y velocidades  $v(t)$  aleatorias. Evaluar *fitness* del enjambre inicial. Se define  $PBest(t)$  como la mejor posición de cada partícula a lo largo de las iteraciones, es decir, para la  $i$ -ésima partícula se tiene  $PBest_i(t) = \operatorname{argmin} \{f(p_i(1)), \dots, f(p_i(t))\}$ ; y  $Best(t)$  corresponde a la mejor partícula a lo largo de todas las iteraciones.
- 3) Mientras no se cumpla algún criterio de término
  - a. Aumentar el número de iteraciones  $t = t + 1$
  - b. Se actualizan las velocidades y posiciones de cada partícula acorde a las siguientes expresiones:

$$v(t + 1) = wv(t) + c_1r_1(t)(Pbest(t) - p(t)) + c_2r_2(t)(Best(t) - p(t)), \quad (8.4)$$

$$p(t + 1) = p(t) + v(t + 1), \quad (8.5)$$

donde  $w$  es el factor de inercia;  $c_1, c_2$  corresponden a la constante cognitiva y social respectivamente;  $r_1(t), r_2(t)$  son número aleatorios. Además se acota la velocidad  $v(t + 1) \in [-v_{max}, v_{max}]$ . y posiciones  $p(t + 1) \in [p_{min}, p_{max}]$ .

- c. Evaluar *fitness* del nuevo enjambre. Actualizar  $PBest(t)$  y  $Best(t)$ .
- 4) Terminado el ciclo, la solución está dada por  $Best(t)$ .

PSO ha sufrido varias modificaciones que ayudan al algoritmo a mejorar su desempeño. Uno de ellas es variar el factor de inercia  $w$  a lo largo de las iteraciones, partiendo por un valor alto,  $w_{inicial}$ , y terminando con un valor bajo,  $w_{final}$ , variando linealmente a lo largo de las iteraciones. Esto ayuda a que el algoritmo explore nuevas soluciones al inicio del ciclo, y luego explote la región de posible solución al final.

Otra modificación incluida es la presentada en [112], donde se incorporan propiedades para la convergencia local del algoritmo PSO. Éste se basa en modificar la

ecuación de velocidad de la mejor partícula, debido a que en dicho caso se tiene  $v(t + 1) = wv(t)$ , por lo que la partícula se mueve sólo en el caso en que  $v(t)$  y  $w$  son distintos de cero, por lo que puede ocurrir una convergencia prematura.

Sea  $\tau$  el índice de la mejor partícula, vale decir,  $PBest_{\tau}(t) = Best(t)$ , luego se modifica la ecuación de velocidad de la forma

$$v_{\tau}(t + 1) = -p_{\tau}(t) + Best(t) + wv_{\tau}(t) + \rho(t)(1 - 2r_2(t)), \quad (8.6)$$

con  $r_2(t)$  un valor aleatorio.  $\rho(t)$  se inicializa en 1 y se calcula en cada iteración como

$$\rho(t + 1) = \begin{cases} 2\rho(t) & \text{Si } \#éxitos > s_c \\ 0.5\rho(t) & \text{Si } \#fracasos > f_c, \\ \rho(t) & \text{En otro caso} \end{cases} \quad (8.7)$$

donde  $\#éxitos$  ( $\#fracasos$ ) indica el número consecutivo de éxitos (fracasos). Un fracaso se define como  $f(Best(t)) = f(Best(t - 1))$ . Para asegurar que (8.7) está bien definida se deben cumplir las siguientes condiciones:

$$\#éxitos(t + 1) > \#éxitos(t) \rightarrow \#fracasos(t + 1) = 0, \quad (8.8)$$

$$\#fracasos(t + 1) > \#fracasos(t) \rightarrow \#éxitos(t + 1) = 0, \quad (8.9)$$

A su vez, los autores de la modificación recomienda utilizar  $f_c = 5$  y  $s_c = 15$ .

## 8.2. Anexo B: Control Predictivo Difuso

Dentro de las estrategias de control avanzadas destaca el control predictivo basado en modelos (*Model-Based Predictive Control*, MPC), una técnica de control óptimo en tiempo discreto que permite tratar con las restricciones físicas que usualmente tienen los procesos en la industria [17]. Este tipo de controladores usan un modelo para predecir el comportamiento de la planta dentro de un horizonte dado, y a partir de dichas predicciones genera una secuencia de acciones de control, las cuales son determinadas a partir de la minimización de una función objetivo que toma en consideración tanto el esfuerzo de control como el error de predicción.

Cuando se utiliza un modelo de predicción no lineal la estrategia pasa a denominarse *Nonlinear Model Predictive Control* (NMPC). Este tipo de controladores han sido probados en diversos estudios demostrando buenos resultados. Dada su habilidad para lidiar con las restricciones que pueda tener el sistema, ya sea en la salida o en la variable de entrada, este tipo de controladores ha aumentado su uso en la industria, siendo una alternativa viable incluso en problemas de modelación de tráfico [37].

Patwardhan & Madhavan hacen uso de esta técnica de control en su línea de investigación [79][80][81], donde en su último trabajo estudian el costo computacional que conlleva la resolución del problema de optimización cuyas restricciones están dadas por el modelo de predicción, el cual se traduce en ecuaciones diferenciales ordinarias no lineales. El uso de *solvers* de este tipo de ecuaciones puede resultar muy costoso computacionalmente. Alternativas como *orthogonal collocation* son métodos de resolución de ecuaciones diferenciales que tienen la ventaja de expresar la dinámica del sistema a través de ecuaciones algebraicas, de tal manera que la acción de control se puede obtener más rápido a través de algoritmos de optimización eficientes. Sin embargo, las no linealidades del proceso persisten, por lo que se requieren varios cálculos iterativos para encontrar una solución. Dado lo anterior, el uso de *solvers* directos u *ortogonal collocation* puede ser restrictivo a la hora de implementar estas estrategias *online* debido al tiempo requerido para resolver completamente el problema de optimización. Por otro lado, la linealización sucesiva del sistema requiere considerablemente menos cálculos, simplifica el problema de optimización y por lo tanto resulta ser una opción más atractiva si se contempla una implementación en tiempo real. No obstante, el uso de aproximaciones en torno a un punto de operación es válido sólo en un rango acotado, por lo que si se produce un cambio drástico en la planta de un instante a otro, ya sea debido a perturbaciones o desgaste de algún componente, la linealización pierde validez [33]. A su vez, utilizando como modelo de predicción las aproximaciones de segundo orden del proceso, Patwardhan & Madhavan demuestran mediante el Lema de Morse que dichos modelos mejoran el desempeño de la planta en lazo cerrado, aumentando la región de controlabilidad del sistema.

En [72] se estudia el uso de NMPC para modelos de predicción no lineales no afines (es decir, no lineales respecto a la variable manipulada). En vista de lidiar con el costo computacional proponen un método iterativo para aproximar la solución. De manera

similar a lo realizado por Patwardhan & Madhavan, este método se basa en la reinterpretación de la predicción de la salida como una serie de Taylor, es decir, se aproxima la dinámica del sistema por una serie de términos hasta determinado orden. Luego, por tratarse de una aproximación de la dinámica del sistema, se deriva una solución subóptima, pero con menor costo computacional. También se hace énfasis en que un mayor horizonte de predicción puede aumentar la robustez del controlador debido a que se toma en consideración el comportamiento de la planta en una ventana de tiempo mayor, lo que ayuda a anticipar comportamientos indeseados y controlarlos con anterioridad.

En [52] se plantea un NMPC basado en multimodelos. El enfoque abordado trata de generar un modelo de predicción a partir de un conjunto de modelos locales, los cuales son combinados mediante una red neuronal perceptrón multicapa (*Multi Layer Perceptron Neural Network*, MLP NN). Los modelos locales son de primer orden y pueden ser obtenidos mediante linealización, o bien utilizando datos de entrada-salida y filtros lineales. Por otro lado, la red neuronal utilizada para combinar los modelos consiste en tres capas: la capa de entrada, la capa oculta y la capa de salida. La capa de entrada posee  $m \times N_r$  neuronas debido a que considera un vector de entrada  $\eta(k) = [\hat{y}_1^T, \dots, \hat{y}_{N_r}^T]^T$ , con  $N_r$  la cantidad de modelos locales y  $m$  la cantidad de salidas. La capa oculta posee  $N_h$  neuronas, cuyo valor se puede modificar dependiendo del grado de precisión deseado. Por último, la capa de salida posee  $m$  neuronas que representan las salidas del sistema multimodelo. Cabe destacar que esta metodología es similar al planteamiento difuso, aunque una vez entrenada la red sus pesos quedan fijos, por lo que la combinación de la salida no varía, a diferencia de los modelos TS, donde la salida ponderada depende principalmente del grado de activación de las premisas en cada instante.

En [19] se estudia una solución subóptima al problema de optimización del NMPC para modelos del tipo  $\dot{x}(t) = f(x(t), u(t))$ . Para ello plantean el uso de geometría diferencial (álgebra de Lie) y de aproximaciones en serie de Taylor (hasta cualquier orden) de sistemas no lineales, donde se aproxima tanto la predicción del sistema como la referencia deseada, derivando en una expresión analítica de la acción de control óptima. Luego, dependiendo del orden de la aproximación, se pueden alcanzar cualquier grado de exactitud deseado. Cabe destacar que el planteamiento realizado es para sistemas sin restricciones y la función objetivo está enfocada sólo a minimizar el error cuadrático, dejando de lado el esfuerzo de control. Dada la expresión analítica derivada del análisis matemático del problema no se tiene que realizar optimización *online*, por lo que este enfoque es más apto para ser implementado en tiempo real incluso en sistemas con respuestas rápidas. Es importante recalcar que la presente metodología resulta ser estable en lazo cerrado dependiendo del tiempo de predicción, el grado relativo del sistema, y el orden la aproximación por serie de Taylor.

Dentro de la categoría NMPC se tiene el *Fuzzy Model Predictive Control* (FMPC), en cuyo caso se utilizan modelos difusos para predecir el comportamiento de la planta, demostrando mejorar características como la respuesta transiente del sistema en

comparación a modelos predictores lineales [44]. A su vez, se ha demostrado que generar un modelo de predicción a partir de métodos de identificación difusa permite diseñar una estrategia de control predictivo que logra bajos errores de seguimiento incluso en plantas prominentemente no lineales [87].

En [39] se postula una metodología basada en *Fuzzy Convolution Models*, los cuales corresponden a modelos de predicción discretos donde cada modelo local tiene una respuesta dada por la convolución discreta entre la respuesta al impulso y la entrada al sistema. En este trabajo se postula un diseño de control jerárquico, el cual resuelve el problema de optimización de manera iterativa y centrada en dos capas: nivel local y nivel global, es decir, se aborda el problema de optimización para cada uno de los modelos locales como también para la salida global predicha, siendo necesario un sistema de coordinación. La idea de esto es iterar sobre las diversas soluciones del problema de optimización y evaluar el error entre la salida predicha y la referencia, si dicho error es pequeño se termina el algoritmo, sino se pasa a la capa baja (nivel local) y se re-evalúan los modelos. Los autores destacan que este método de resolución del problema de optimización evita extensivos cálculos de optimización no lineal *online*, aumentando la factibilidad del uso de estas estrategias en tiempo real.

En [117] se plantea una metodología basada en modelos difusos TS con consecuencias lineales que permite asegurar la estabilidad del sistema al utilizar el FMPC. Para ello se calculan las acciones de control a partir de un conjunto de inecuaciones, asegurando que una función extendida difusa de Lyapunov (*Extended Fuzzy Lyapunov Function*, extended-FLF) cumpla las condiciones que aseguren estabilidad. De esta forma se garantiza tanto la estabilidad del sistema como el cumplimiento de las restricciones que éste pueda tener tanto en la salida como en la variable manipulada.

En [5] se propone el uso de modelos relacionales difusos, los cuales poseen 3 componentes principales: fusificación, la matriz relacional difusa y la defusificación. Las ecuaciones relacionales quedan definidas por un operador que, en este caso, corresponde a la composición suma-producto. A raíz de este tipo de modelo de predicción se genera una estrategia FMPC, la cual es comparada con métodos como PID y control por modos deslizantes (*Sliding Mode Control*, SMC). A raíz de un análisis comparativo se concluye que el FMPC es más robusto que estrategias de control más clásicas.

Djebrani & Abdessemed proponen en [23] el diseño de un controlador predictivo difuso basado en los modelos locales lineales que posee el modelo difuso de la planta real. A partir de cada consecuente lineal se deriva un controlador predictivo para cada regla, con los cuales posteriormente se obtiene una acción de control global calculada como la combinación de los controladores obtenidos para cada regla, generando una estrategia de control difusa similar al modelo difuso TS. La desventaja de este enfoque radica en el costo computacional asociado, debido a que es necesario resolver un problema de optimización por cada regla para obtener una acción de control óptima.



La resolución del problema de optimización del FMPC es un tema bastante abordado en la literatura, debido a que encontrar soluciones en tiempos acotados es fundamental para la aplicación de estas estrategias en tiempo real en diversos procesos de la industria. En [92] proponen el uso de algoritmos genéticos para resolver el problema, mientras que en [100] se utiliza PSO. Ambas herramientas permiten configurar el algoritmo para resolver el problema de optimización de manera relativamente acertada y con menor carga computacional, sin embargo, no se puede asegurar la convergencia al valor óptimo. Por otro lado, en [61] estudian el uso tanto de GA como del método *Branch & Bound* para resolver el problema de optimización. En el caso del *Branch & Bound* se tienen dos inconvenientes: primero, el tiempo computacional crece exponencialmente con el horizonte de control y el número de alternativas; y segundo, la discretización de las posibles acciones de control, la cual puede provocar sobrepasos y respuestas más lentas, por lo que los algoritmos evolutivos como GA y PSO aparecen como las principales cartas para mejorar los tiempos de cómputo.

Finalmente, en la literatura se encuentran varias aplicaciones en donde el FMPC permite controlar diversos procesos, como por ejemplo sistemas de aire acondicionado [105], motores en vehículos eléctricos [122], turbinas de calderas [117], plantas nucleares [123], e incluso en ámbitos de la medicina para el control del nivel de glucosa [3]. Esto demuestra la gran versatilidad que puede tener el FMPC en la industria, volviéndolo una herramienta poderosa para el control de sistemas.

En particular para un sistema SISO, el control predictivo difuso utiliza una función de costos cuadrática para medir el desempeño del sistema, dada por

$$J(z(k), \bar{u}(k)) = Q \sum_{i=1}^{N_p} e^2(k+i|k) + R \sum_{i=1}^{N_u} u^2(k+i-1), \quad (8.10)$$

donde  $z(k)$  son las premisas del modelo;  $e(k+i|k) = r(k+i) - y_{fuzzy}(k+i|z(k))$  representa el error en el instante  $k+i$  dada las condiciones en el instante  $k$ , con  $r(k+i)$  la referencia en el instante  $(k+i)$  e  $y_{fuzzy}(k+i|z(k))$  la predicción del modelo difuso en el instante  $(k+i)$ ;  $u(k+i-1)$  es la acción de control en  $(k+i-1)$ ;  $\bar{u}(k) = [u(k), \dots, u(k+N_u), \dots, u(k+N_p-1)]^T$ , donde  $u(k+i) = u(k+N_u)$  para  $i = N_u, \dots, N_p-1$ ;  $N_p$  y  $N_u$  denotan el horizonte de predicción y de control respectivamente, con  $N_p \geq N_u$ ;  $Q, R > 0$  son ponderadores escalares.

A su vez, la función objetivo puede ser escrita considerando la variación de la acción de control,  $\Delta u(k) = u(k) - u(k-1)$ , resultando

$$J(z(k), \Delta \bar{u}(k)) = Q \sum_{i=1}^{N_p} e^2(k+i|k) + R \sum_{i=1}^{N_u} \Delta u^2(k+i-1), \quad (8.11)$$

con  $\Delta\bar{u}(k) = [\Delta u(k), \dots, \Delta u(k + N_u), \dots, \Delta u(k + N_p - 1)]^T$ .

Sea  $\mathbb{Y}$ ,  $\mathbb{U}$  y  $\Delta\mathbb{U}$  el espacio factible para la salida, la entrada y la variación de la entrada respectivamente. Luego, el control predictivo difuso se puede formular a partir del siguiente problema de optimización:

$$\begin{aligned} & \min_{\bar{u}(k)} J(z(k), \bar{u}(k)) \\ \text{s. a. } & y_{fuzzy}(k + i|z(k)) = \sum_{r=1}^{N_r} h_r(k + i|z(k)) \cdot y_r(k + i|z(k)) \\ & y_{fuzzy}(k + i|z(k)) \in \mathbb{Y}, \quad u(k + i - 1) \in \mathbb{U}, \quad \Delta u(k + i - 1) \in \Delta\mathbb{U}, \quad i = 1, \dots, N_p, \end{aligned} \quad (8.12)$$

o bien

$$\begin{aligned} & \min_{\Delta\bar{u}(k)} J(z(k), \Delta\bar{u}(k)) \\ \text{s. a. } & y_{fuzzy}(k + i|z(k)) = \sum_{r=1}^{N_r} h_r(k + i|z(k)) \cdot y_r(k + i|z(k)) \\ & y_{fuzzy}(k + i|z(k)) \in \mathbb{Y}, \quad u(k + i - 1) \in \mathbb{U}, \quad \Delta u(k + i - 1) \in \Delta\mathbb{U}, \quad i = 1, \dots, N_p, \end{aligned} \quad (8.13)$$

dependiendo del tipo de función objetivo a utilizar.

Dado lo expuesto en (8.10)-(8.13), se tiene que el control predictivo es abordado como un problema de programación cuadrática, para lo cual existen diversos algoritmos de solución. Dentro de dichos algoritmos se encuentra la programación cuadrática secuencial (*Sequential Quadratic Programming*, SQP), métodos de punto interior (*Interior Point Methods*), y algoritmos evolutivos como el mismo PSO presentado en la Sección 8.1.3 [100], o algoritmos genéticos [92], donde estos dos últimos demuestran buenos resultados reduciendo el tiempo de cómputo. A continuación se detallan algunos de estos métodos de optimización.

## 8.2.1. Algoritmos de Solución

Para detallar los algoritmos se considera el siguiente problema general:

$$\begin{aligned} & \min_x f(x) \\ \text{s. a. } & h(x) = 0, \quad g(x) \leq 0, \end{aligned} \quad (8.14)$$

donde  $x$  es la variable,  $f(x)$  es la función objetivo,  $h(x)$  son las restricciones de igualdad y  $g(x)$  las restricciones de desigualdad.

### 8.2.1.1. Programación Cuadrática Secuencial

El principal objetivo de la optimización con restricciones es transformar el problema original en un subproblema que pueda ser resuelto iterativamente [66]. En particular, este método emplea el Lagrangiano

$$L(x, \lambda) = f(x) + \lambda^T h(x) + \mu^T g(x), \quad (8.15)$$

con  $\lambda$  y  $\mu$  denominados multiplicadores de Lagrange.

La principal idea detrás del algoritmo de programación cuadrática secuencial es formular un subproblema mediante la aproximación de segundo orden del Lagrangiano en (8.15) y linealizando las restricciones en torno a la  $k$ -ésima iteración  $x_k$ . De este modo el subproblema queda

$$\begin{aligned} \min_d \quad & \nabla f(x_k) + \frac{1}{2} d^T \nabla^2 L(x_k, \lambda_k) d \\ \text{s. a.} \quad & \nabla h(x_k)^T d + h(x_k) = 0 \\ & \nabla g(x_k)^T d + g(x_k) \leq 0. \end{aligned} \quad (8.16)$$

Al resolver (8.16) se actualiza el valor de  $x$  utilizando  $x_{k+1} = x_k + \alpha_k d$ , donde  $\alpha_k$  se determina mediante un algoritmo de búsqueda de línea tal que una función de mérito decrezca.

### 8.2.1.2. Método de Punto Interior

El método de punto interior busca transformar el problema de optimización mediante la inclusión de condiciones de barrera en la función objetivo [66]. El problema de optimización modificado se detalla en (8.17).

$$\begin{aligned} \min_{x,s} \quad & f_\mu(x, s) = f(x) - \mu \sum_i \ln(s_i) \\ \text{s. a.} \quad & h(x) = 0 \\ & g(x) + s = 0. \end{aligned} \quad (8.17)$$

Notar que existen tantas variables  $s_i$  como restricciones de inequidad tenga el problema original. Además se restringe que  $s_i > 0$  tal que las expresiones estén acotadas y bien definidas. Cabe destacar que el problema de optimización se simplifica al contener sólo restricciones de igualdad. Para resolver (8.17) se puede utilizar el método de paso directo, o bien el método del gradiente conjugado; para más detalles revisar [66].

Si bien los dos algoritmos detallados pueden ser utilizados para resolver el problema de optimización presentado en el control predictivo, el método de punto interior se ha vuelto más popular dentro de esta estrategia debido a que sus tasas de convergencia son más rápidas [88], sin embargo, lo anterior está ligado a la aplicación en que se utilicen los algoritmos. A su vez, es posible también configurar adecuadamente los parámetros del algoritmo de enjambre de partículas o algoritmos genéticos de tal forma de minimizar el tiempo de cómputo.

Una vez obtenido el vector de acciones de control futuras  $\bar{u}(k)$ , o el vector de variaciones de acciones de control futuras  $\Delta\bar{u}(k)$ , mediante la solución del problema de optimización en (8.12) o (8.13) según corresponda, utilizando alguno de los algoritmos anteriores, se aplica al proceso la acción de control actual solamente,  $u(k)$ .

Cabe destacar que el control predictivo posee dos parámetros,  $R$  y  $Q$ , acorde a la función objetivo en (8.10)-(8.11), por lo que se requiere de un proceso de sintonización para encontrar los parámetros óptimos que permitan alcanzar los objetivos de control deseados.

## 8.2.2. Sintonización de Controladores

Comúnmente se utiliza la dinámica del error de un sistema como principal indicador del desempeño de un controlador, determinando la calidad de la respuesta obtenida comparándola con la respuesta deseada del sistema. Dado lo anterior, existen variados índices que evalúan dicho desempeño, analizando el error acumulado durante la simulación.

Dentro de los criterios de desempeño utilizados para la optimización de los parámetros de un controlador se pueden encontrar principalmente la integral del error absoluto (*Integral Absolute Error*, IAE), la integral de error cuadrático (*Integral Squared Error*, ISE), la integral del tiempo por el error absoluto (*Integral Time-weighted Absolute Error*, ITAE), y la integral del tiempo por el error cuadrático (*Integral Time-weighted Squared Error*, ITSE) [103].

La principal diferencia entre el IAE/ISE y el ITAE/ITSE es que estos últimos ponderan el error por el tiempo, lo que otorga un menor peso a los grandes errores que pueden existir inicialmente en el sistema, mientras que se penaliza con un mayor valor los errores en estado estacionario. Por otro lado, la diferencia entre considerar el error absoluto y el error cuadrático es que este último busca una rápida disminución de los errores grandes, ponderando en menor medida los errores pequeños, en contraste a que el error absoluto busca disminuir el error en igual medida [64].

Dado que generalmente se busca obtener un error en estado estacionario pequeño, en este trabajo se opta por utilizar el ITAE como índice de desempeño del controlador,

debido a que se pondera con mayor medida los errores en estado estacionario, y además no se disminuye el efecto de errores pequeños, como lo hace el ITSE.

El criterio de desempeño ITAE se define como

$$ITAE = \int_0^{\infty} t \cdot |e(t)| dt, \quad (8.18)$$

donde  $e(t) = r(t) - y(t)$ , con  $r(t)$  la referencia e  $y(t)$  la salida de la planta.

Para llevar (8.18) a tiempo discreto se aproxima la integral por una sumatoria, de la forma

$$ITAE = \sum_{k=0}^{\infty} t(k) \cdot |e(k)| \Delta t(k), \quad (8.19)$$

Con  $\Delta t(k) = t(k) - t(k - 1)$ .

Luego, es posible plantear un problema de optimización que busque minimizar (8.19), con el fin de encontrar los parámetros adecuados del controlador. En el caso particular de utilizar un controlador predictivo, dicho problema se formula como

$$\min_{Q,R} \sum_{k=0}^{\infty} t(k) \cdot |r(k) - y(k, Q, R)|, \quad (8.20)$$

donde  $Q, R$  son los parámetros del controlador, los cuales influyen directamente en la salida del sistema  $y(t, Q, R)$ . Notar que  $t(k) = t(k - 1) + T_s$  cuando se utiliza un tiempo de muestreo fijo, por lo que  $\Delta t(k) = T_s$ , y al ser constante no influye en el problema de optimización.

Para resolver (8.20) se pueden utilizar diversos algoritmos de optimización. En particular, se plantea el uso de PSO para llevar a cabo esta tarea, debido a que permite evaluar una mayor cantidad de posibles soluciones simultáneamente, y además permite acotar de manera más intuitiva el tiempo de cómputo, gracias a la elección del número de partículas e iteraciones deseado.

Una vez detallado el control predictivo, la manera en que se puede sintonizar y los algoritmos que se pueden utilizar para obtener una acción de control óptima, en la siguiente sección se detallan algunas consideraciones en el diseño del controlador al momento de implementar esta estrategia en tiempo real, analizando principalmente el tiempo de cómputo que se requiere para obtener una acción de control.

## 8.2.3. Consideraciones en el Diseño de Controlador Predictivo

Es posible emplear diversos métodos para lograr implementar un controlador predictivo en la práctica, lo cual puede empeorar levemente su desempeño, pero seguir cumpliendo con el objetivo de control: llevar las variables a los puntos de operación deseados, o en su defecto, a una zona cercana a la referencia deseada.

En esta sección se plantean diversos métodos para poder implementar un controlador predictivo en tiempo real. Con el fin de ilustrar las consideraciones propuestas se emplea el reactor de estanque continuamente agitado, junto al controlador de mejor desempeño presentado en la Sección 4.2.1.2 (ver Tabla 4.11). A su vez, las pruebas se realizan en un computador con procesador Intel Core i5 de 2.27 GHz y 3 Gb de RAM.

### 8.2.3.1. Horizonte de Predicción

El horizonte de predicción ( $N_p$ ) utilizado en el planteamiento del controlador predictivo influye directamente en el tiempo de cómputo para obtener una acción de control, debido a que horizontes de predicción más largos involucran una mayor cantidad de variables, complejizando el problema de optimización.

Cabe destacar que la elección de  $N_p$  en el planteamiento está asociada a la calidad del modelo utilizado como predictor. Si dicho modelo presenta un error considerable al predecir el comportamiento de la planta en horizontes largos, esto repercute en el desempeño en lazo cerrado del controlador, pudiendo alcanzar incluso condiciones de inestabilidad [88].

Si bien es recomendable utilizar horizontes de predicción que logran capturar gran parte del comportamiento transiente de la planta, la predicción del comportamiento de plantas altamente inestables en largos horizontes de predicción debe ser abordada con cuidado, por lo que puede ser recomendable utilizar valores de  $N_p$  bajos en dichos escenarios para obtener un buen desempeño.

En resumen, la elección de un horizonte de predicción afecta en el desempeño en lazo cerrado del sistema, y dependiendo tanto de su valor como del modelo de predicción se pueden obtener mejores o peores respuestas en lazo cerrado.

En particular, para el caso de estudio se tienen los tiempo de cómputo expuestos en la Tabla 8.1, donde se observa que utilizar una menor cantidad de variables de optimización reduce considerablemente el esfuerzo computacional necesario para obtener una solución óptima. Cabe destacar que en ambos casos el sistema logra

converger a la referencia deseada con un bajo error en estado estacionario  $e_{ss}$ , sin embargo, el mejor desempeño se obtiene cuando  $N_p = 10$ .

**Tabla 8.1: Tiempos de Cómputo Variando  $N_p$ .**

$N_p$	<b>Tiempo Promedio [s]</b>	<b>Desviación Estándar del Tiempo [s]</b>	<b>Tiempo Máximo [s]</b>	$e_{ss}$
<b>1</b>	<b>0.0541</b>	<b>0.0244</b>	<b>0.1152</b>	<b>0.0018</b> <b>(0.3456 %)</b>
10	1.6525	1.9040	4.5345	7.1008e-04 (0.1396 %)

### 8.2.3.2. Algoritmo de Solución

Otro aspecto que influye en la eficiencia con que se determina una acción de control óptima es el algoritmo utilizado para resolver el problema de optimización planteado por el controlador predictivo. Según lo expuesto en la Sección 8.2.1, existen diversos algoritmos para resolver un problema de optimización con restricciones, como puede ser SQP, el método de punto interior, y PSO. Cabe recordar que este último puede reducir el tiempo de cómputo, sin embargo, no asegura la convergencia a alguna solución óptima, y por lo tanto puede afectar el desempeño en lazo cerrado del sistema en caso de que no se elijan los parámetros del algoritmo adecuadamente. En la Tabla 8.2 se enlistan los parámetros utilizados por PSO para resolver el problema de optimización del controlador predictivo, donde se tiene el cuidado de no empeorar la respuesta en lazo cerrado del controlador.

**Tabla 8.2: Parámetros PSO para Controlador Predictivo Difuso.**

<b>Parámetro</b>	<b>Valor</b>
Tamaño de la población	6
Número de iteraciones	4
$[p_{min}, p_{max}]$	[300,490]
$V_{max}$	190
$[c_1, c_2]$	[2,2]
$[w_{inicial}, w_{final}]$	[0.95,0.4]
$[f_c, s_c]$	[5,15]

En la Tabla 8.3 se presentan los resultados obtenidos al utilizar los distintos *solvers*, empleando un horizonte de predicción  $N_p = 1$ . Se observa que el algoritmo PSO permite ajustar sus parámetros de tal manera que el tiempo de cómputo se reduzca en comparación a SQP y punto interior, sin perjuicio de afectar la respuesta en lazo cerrado del controlador, y en particular, el error en estado estacionario  $e_{ss}$ . A su vez, PSO

presenta tiempos menos variables, debido a que el criterio de término en este caso se limita al número de iteraciones, demorándose prácticamente el mismo tiempo entre una iteración y otra, lo cual resulta una ventaja en comparación a los otros *solvers* empleados, donde la desviación estándar del tiempo de cómputo es mayor y en consecuencia se tiene mayor incerteza del esfuerzo computacional requerido para encontrar una solución al problema de optimización.

**Tabla 8.3: Tiempos de Cómputo Variando Algoritmos de Optimización.**

<i>Solver</i>	<i>Tiempo Promedio [s]</i>	<i>Desviación Estándar del Tiempo [s]</i>	<i>Tiempo Máximo [s]</i>	$e_{ss}$
SQP	0.0541	0.0244	0.1152	0.0018 (0.3456 %)
Punto Interior	0.1701	0.1146	0.4324	0.0018 (0.3456 %)
<b>PSO</b>	<b>0.0355</b>	<b>3.3795e-04</b>	<b>0.0376</b>	<b>0.0019</b> <b>(0.3662 %)</b>

### 8.2.3.3. Tiempo de Muestreo y Tiempo de Aplicación

El tiempo de muestreo utilizado en el sistema impone una cota para la resolución del problema de optimización del control predictivo, por lo que juega un rol importante al momento de diseñar este tipo de estrategias.

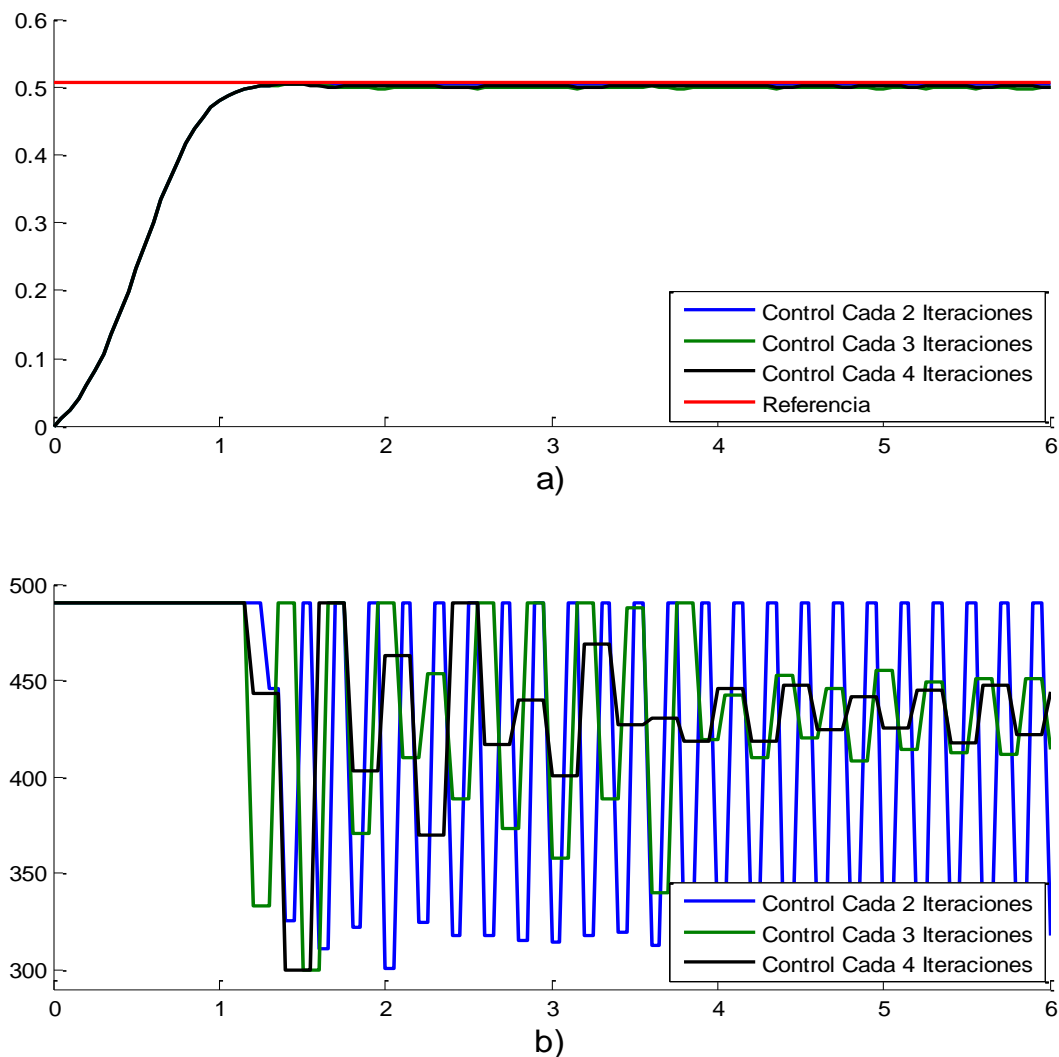
En caso de poder seleccionar la tasa de muestreo aplicada a la planta es posible variar su valor y obtener datos para generar modelos de predicción discretos acordes al tiempo de muestreo utilizado. Luego, si se escoge un tiempo que respete la relación impuesta por el teorema de Shanon (ver (2.9)), y se genera el modelo de predicción adecuado, es posible relajar la condición impuesta sobre el algoritmo de optimización utilizado.

Sin embargo, es posible que el valor del tiempo de muestreo en los procesos en la industria este establecido de antemano, por lo que no sería posible variar dicha tasa y generar un modelo de predicción adecuado. En vista de lo anterior, es posible variar el tiempo en que se aplica el controlador predictivo, es decir, no necesariamente resolver el problema de optimización en cada iteración, sino que mantener la señal constante por un tiempo mayor al tiempo de muestreo, tal que sea posible calcular nuevamente una acción de control óptima. Si bien lo anterior puede generar un empobrecimiento en el desempeño del controlador, en algunos casos cuando se tienen tiempos de muestreo demasiado altos en comparación al tiempo de respuesta de la planta es posible que el controlador mantenga un buen desempeño en lazo cerrado.

A modo de ejemplo, se emplea la estrategia antes descrita en el caso de estudio, donde se aplica la señal de control cada dos, tres y cuatro iteraciones, y no en cada



iteración como se realiza usualmente. En la Figura 8.2 a) se presenta el desempeño en lazo cerrado del controlador, donde se observan pequeñas variaciones de la salida en estado estacionario, lo cual repercute en el error obtenido, como se observa en la Tabla 8.4; sin embargo, el controlador cumple con el objetivo de llevar el valor de la salida a una región cercana a la referencia. A su vez, la Figura 8.2 b) presenta la entrada al sistema, donde se observa los distintos tiempos en que la acción de control permanece constante, dando mayor flexibilidad con respecto al tiempo para resolver el problema de optimización del control predictivo para ser aplicado en el instante siguiente. Cabe destacar que mientras mayor es el tiempo en que permanece constante la señal, menor es su variación, debido a que el controlador intenta mantener la entrada entorno a un valor promedio.



**Figura 8.2: Resultados Control Predictivo Difuso del CSTR Usando Distintos Tiempos de Aplicación. a) Salida  $y(t)$  vs Tiempo [s]. b) Entrada  $u(t)$  [K] vs Tiempo [s].**

**Tabla 8.4: Error en Estado Estacionario Usando Control Predictivo Difuso en el CSTR Usando Distintos Tiempos de Aplicación.**

<b>Tiempo de Aplicación</b>	<b><math>e_{ss}</math></b>
Cada 2 Iteraciones	0.0026 (0.5163 %)
Cada 3 Iteraciones	0.0084 (1.6491 %)
Cada 4 Iteraciones	0.0060 (1.1847 %)

## 8.3. Anexo C: Resultados del Proceso de Identificación

### 8.3.1. Modelos Difusos Controlables de 2do Orden CSTR

En las siguientes tablas se resumen los resultados obtenidos utilizando  $n_y = 2$ ,  $n_u = 1$ , para las configuraciones óptimas encontradas en las pruebas de la Sección 4.2.1.1.

Tabla 8.5: Resultados Identificación en Lazo Abierto Usando  $n_y = 2$ ,  $n_u = 1$ , Método QN/NM/PSO, CI's Conocidas e Identificación a un Paso.

Número de Reglas	RMSE Entrenamiento		RMSE Validación		RMSE Prueba	
	1 Paso	N Pasos	1 Paso	N Pasos	1 Paso	N Pasos
1	4.3444 e-04	0.1327	3.4421 e-04	0.1175	4.4019 e-04	0.1229
2	3.9506 e-04	0.1622	3.1868 e-04	0.1278	4.0058 e-04	0.1602
3	3.7584 e-04	0.4782	2.9371 e-04	0.1720	3.8709 e-04	0.2066
4	3.1691 e-04	0.1169	<b>2.4073 e-04</b>	0.1021	3.4077 e-04	0.1055
5	3.5936 e-04	0.1293	2.5158 e-04	0.0978	3.8466 e-04	0.1082

Tabla 8.6: Resultados Identificación en Lazo Abierto Usando  $n_y = 2$ ,  $n_u = 1$ , Método NM, CI's Conocidas e Identificación a N Pasos.

Número de Reglas	RMSE Entrenamiento		RMSE Validación		RMSE Prueba	
	1 Paso	N Pasos	1 Paso	N Pasos	1 Paso	N Pasos
1	4.3465 e-04	0.1327	3.4609 e-04	0.1213	4.4087 e-04	0.1237
2	3.9690 e-04	0.1444	3.1983 e-04	0.1205	4.0286 e-04	0.1348
3	4.5272 e-04	0.1030	4.0552 e-04	<b>0.0881</b>	4.6794 e-04	0.0942
4	4.9720 e-04	0.1040	3.9585 e-04	0.0963	4.5237 e-04	0.1172
5	9.1018 e-04	0.0901	6.4988 e-04	0.0904	7.1344 e-04	0.1321

**Tabla 8.7: Resultados Identificación en Lazo Cerrado Usando  $n_y = 2, n_u = 1$ , Método NM, CI's Conocidas e Identificación a un Paso.**

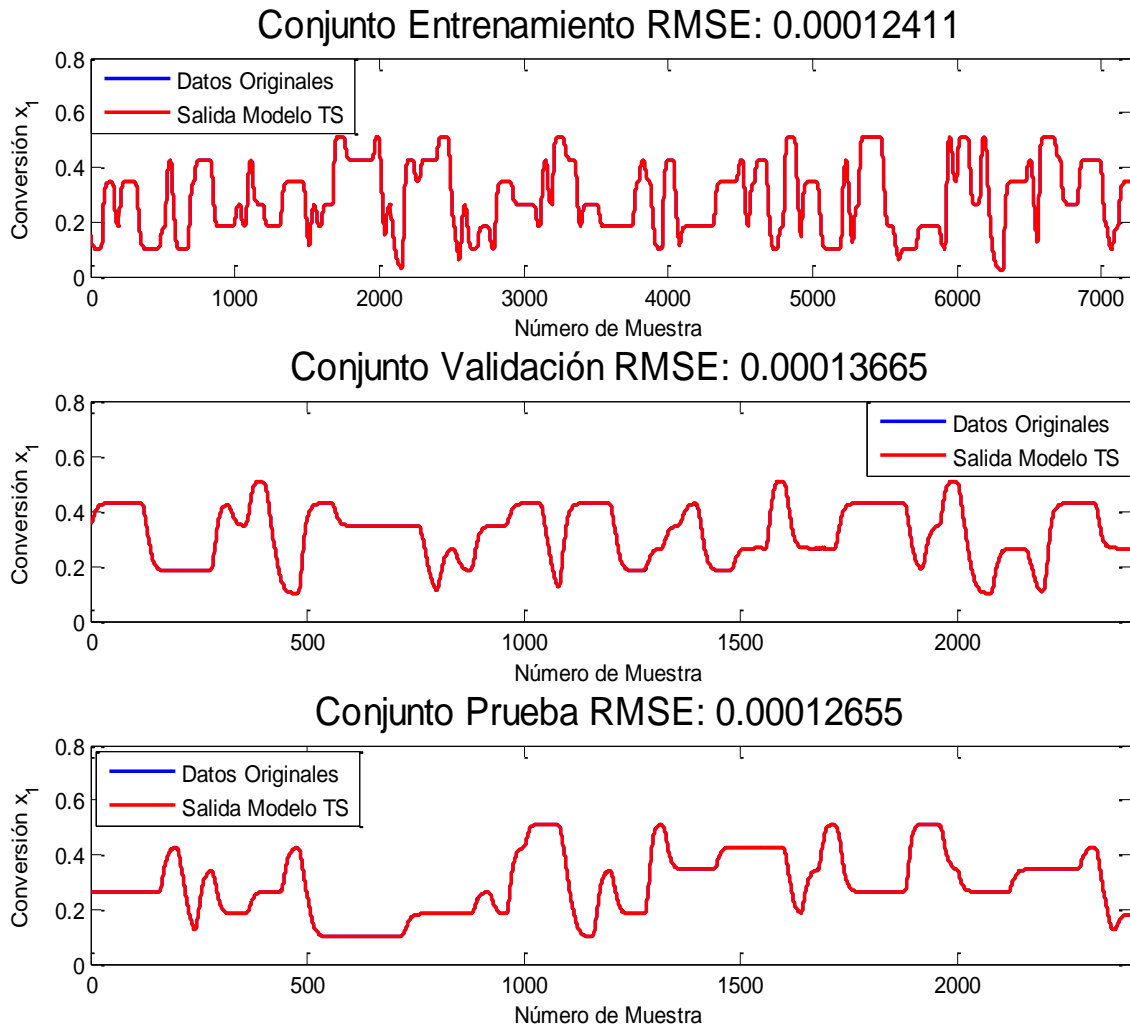
Número de Reglas	RMSE Entrenamiento		RMSE Validación		RMSE Prueba	
	1 Paso	N Pasos	1 Paso	N Pasos	1 Paso	N Pasos
1	4.3795 e-04	0.1046	4.3975 e-04	0.0827	4.1606 e-04	0.0941
2	3.5951 e-04	0.0658	3.4834 e-04	0.0642	3.2234 e-04	0.0642
3	3.6402 e-04	0.0790	3.5467 e-04	0.0761	3.3797 e-04	0.0696
4	3.4160 e-04	0.0669	3.3475 e-04	0.0714	3.0678 e-04	0.0838
5	2.4222 e-04	0.0329	<b>2.2255 e-04</b>	0.0314	2.2350 e-04	0.0294

**Tabla 8.8: Resultados Identificación en Lazo Cerrado Usando  $n_y = 2, n_u = 1$ , Método QN-PSO, CI's Conocidas e Identificación a N Pasos.**

Número de Reglas	RMSE Entrenamiento		RMSE Validación		RMSE Prueba	
	1 Paso	N Pasos	1 Paso	N Pasos	1 Paso	N Pasos
1	4.3795 e-04	0.1046	4.3975 e-04	0.0827	4.1606 e-04	0.0941
2	3.5951 e-04	0.0658	3.4834 e-04	0.0642	3.2234 e-04	0.0642
3	3.6402 e-04	0.0790	3.5467 e-04	0.0761	3.3797 e-04	0.0696
4	3.4160 e-04	0.0669	3.3475 e-04	0.0714	3.0678 e-04	0.0838
5	2.4222 e-04	0.0329	2.2255 e-04	<b>0.0314</b>	2.2350 e-04	0.0294

### 8.3.2. Modelo Difuso no Controlable CSTR

En la Figura 8.3 se observa el desempeño de predicción a un paso de un modelo difuso TS no controlable para la representación de la dinámica del CSTR. En este caso se consideran modelos locales autorregresivos  $y_r(z(k)) = \theta_{r,1}y(k - 1) + \dots + \theta_{r,5}y(k - 5)$ .



**Figura 8.3: Resultado Modelo no Controlable Identificación a un Paso con Datos en Lazo Cerrado. Arriba: Conjunto de Entrenamiento. Medio: Conjunto de Validación. Abajo: Conjunto de Prueba.**

## 8.4. Anexo D: Formas Canónicas de Modelos Lineales

Los sistemas dinámicos en general pueden ser representados tanto por ecuaciones entrada-salida, como también por ecuaciones en variables de estado. Luego, el análisis de las propiedades de un modelo puede ser abordado utilizando una u otra representación.

En particular para modelos lineales con parámetros constantes, es posible transformar un sistema desde su representación en variables de estado a su representación entrada-salida (función de transferencia,  $G(z)$ ) utilizando la transformación

$$G(z) = \frac{Y(z)}{U(z)} = C(zI - A)^{-1}B + D, \quad (8.21)$$

donde el modelo en variables de estado está descrito por

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) + Du(k). \end{aligned} \quad (8.22)$$

Por otro lado, el proceso de transformar un sistema en su representación entrada salida a la ecuación en variables de estados no siempre es directo. Sin embargo, existen las llamadas formas canónicas que permiten llevar un modelo de entrada-salida a variables de estado utilizando los mismos parámetros de la función de transferencia, pero considerando algunos supuestos. Estas representaciones se denominan forma canónica controlable y forma canónica observable [47].

Con el fin de detallar las representaciones canónicas se considera el siguiente modelo en función de transferencia:

$$G(z) = \frac{Y(z)}{U(z)} = \frac{\beta_0 z^n + \beta_1 z^{n-1} \dots + \beta_n}{z^n + \alpha_1 z^{n-1} - \dots - \alpha_n}, \quad (8.23)$$

### 8.4.1. Forma Canónica Controlable

Para derivar la forma canónica controlable considérese

$$\frac{\bar{X}(z)}{U(z)} = \frac{1}{z^n + \alpha_1 z^{n-1} - \dots - \alpha_n}, \quad (8.24)$$

lo que en el dominio del tiempo puede ser escrito como

$$\bar{x}(k+n) + \alpha_1 \bar{x}(k+n-1) + \dots + \alpha_n \bar{x}(k) = u(k). \quad (8.25)$$

A su vez, la salida del sistema puede ser escrita en términos de  $\bar{X}(z)$  como

$$Y(z) = (\beta_0 z^n + \beta_1 z^{n-1} \dots + \beta_n) \bar{X}(z), \quad (8.26)$$

lo que en el dominio del tiempo resulta

$$y(k) = \beta_0 \bar{x}(k+n) + \beta_1 \bar{x}(k+n-1) \dots + \beta_n \bar{x}(k). \quad (8.27)$$

Por lo tanto, el sistema en variables de estado puede ser escrito considerando (8.25) y (8.27) de la forma

$$\begin{aligned} x_1(k+1) &= x_2 \\ x_2(k+1) &= x_3 \\ &\vdots \\ x_n(k+1) &= -\alpha_n x_1(k) - \alpha_{n-1} x_2(k) - \dots - \alpha_1 x_n(k) + u(k) \end{aligned} \quad (8.28)$$

$$y(k) = (\beta_n - \beta_0 \alpha_n) x_1(k) + (\beta_{n-1} - \beta_0 \alpha_{n-1}) x_2(k) + \dots + (\beta_1 - \beta_0 \alpha_1) x_n(k) + \beta_0 u(k).$$

De (8.28) se identifican las matrices del modelo lineal en variables de estado:

$$\begin{aligned} A &= \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ -\alpha_n & -\alpha_{n-1} & -\alpha_{n-2} & \dots & -\alpha_1 \end{bmatrix}, & B &= \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \\ C &= [\beta_n - \beta_0 \alpha_n \quad \beta_{n-1} - \beta_0 \alpha_{n-1} \quad \dots \quad \beta_1 - \beta_0 \alpha_1], & D &= \beta_0 \end{aligned} \quad (8.29)$$

Cabe destacar que la solución de la ecuación de diferencias en (8.22) puede ser escrita como

$$x(n) = A^n x(0) + \mathbb{C}U, \quad U = [u(n), \dots, u(0)]^T, \quad (8.30)$$

donde  $\mathbb{C}$  es la matriz de controlabilidad del sistema (ver (3.5)). Luego, es posible introducir nuevas coordenadas mediante una transformación lineal  $T$ , tal que  $\tilde{x}(k) = Tx(k)$ , y  $\tilde{x}(k)$  corresponda a la forma canónica controlable. Asociando la transformación lineal con (8.30) se puede derivar que la matriz de controlabilidad del sistema en el nuevo sistema de coordenadas  $\tilde{x}(k)$  esta dada por

$$\tilde{\mathbb{C}} = T\mathbb{C}. \quad (8.31)$$

Por lo tanto, para llevar cualquier sistema descrito en variables de estado a su forma canónica controlable se debe aplicar la transformación

$$T = \tilde{C}C^{-1}, \quad (8.32)$$

donde  $\tilde{C}$  es la matriz de controlabilidad del sistema en su forma canónica controlable, y  $C$  es la matriz de controlabilidad en la representación de estados original. De lo anterior se concluye que un sistema puede ser escrito en su forma canónica controlable sólo si el sistema original es controlable, en caso contrario se violan los supuestos necesarios para describir el sistema mediante esta formulación.

### 8.4.2. Forma Canónica Observable

Para obtener la forma canónica observable es necesario reescribir (8.23) de la forma

$$Y(z) = \beta_0 U(z) - z^{-1}[\alpha_1 Y(z) - \beta_1 U(z)] - \dots - z^{-n}[\alpha_n Y(z) - \beta_n U(z)], \quad (8.33)$$

de donde se observa que el sistema puede ser expresado en variables de estado como

$$\begin{aligned} y(k) &= x_n(k) + \beta_0 u(k) \\ x_n(k+1) &= -\alpha_1(x_n(k) + \beta_0 u(k)) + \beta_1 u(k) + x_{n-1}(k) \\ x_{n-1}(k+1) &= -\alpha_2(x_n(k) + \beta_0 u(k)) + \beta_2 u(k) + x_{n-2}(k) \\ &\vdots \\ x_1(k+1) &= -\alpha_n(x_n(k) + \beta_0 u(k)) + \beta_n u(k), \end{aligned} \quad (8.34)$$

donde a partir de (8.34) se identifica

$$\begin{aligned} A &= \begin{bmatrix} 0 & 0 & \dots & 0 & -\alpha_n \\ 1 & 0 & \dots & 0 & -\alpha_{n-1} \\ 0 & 1 & \ddots & \vdots & -\alpha_{n-2} \\ \vdots & \ddots & \ddots & 0 & \vdots \\ 0 & \dots & 0 & 1 & -\alpha_1 \end{bmatrix}, & B &= \begin{bmatrix} \beta_n - \beta_0 \alpha_n \\ \beta_{n-1} - \beta_0 \alpha_{n-1} \\ \vdots \\ \beta_1 - \beta_0 \alpha_1 \end{bmatrix} \\ C &= [0 \quad \dots \quad 0 \quad 1], & D &= \beta_0. \end{aligned} \quad (8.35)$$

Cabe recordar que un sistema es observable si es posible estimar el estado del sistema dada una secuencia de acciones de control  $u(k)$  y una serie de datos de salida  $y(k)$ . A partir de (8.22) se tiene que

$$y(n) = CA^n x(0) + CCU + Du(n), \quad U = [u(n), \dots, u(0)]^T. \quad (8.36)$$

Sin pérdida de generalidad, sea  $u(k) = 0 \forall k$ , luego se tiene que



$$\begin{bmatrix} y(0) \\ y(1) \\ \vdots \\ y(n-1) \end{bmatrix} = \mathbb{O}x(0), \quad \mathbb{O} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}, \quad (8.37)$$

donde  $\mathbb{O}$  es la matriz de observabilidad del sistema.

Similar a lo realizado en la Sección 8.4.1, se puede aplicar una transformación lineal al sistema para llevarlo desde cualquier representación en variables de estado a la forma canónica observable. Para ello se tiene que la ecuación de salida de la transformación  $\tilde{x}(k) = Tx(k)$  esta dada por  $y(k) = CT^{-1}\tilde{x}(k) + Du(k)$ . Luego, la relación entre las matrices de observabilidad está dada por

$$\tilde{\mathbb{O}} = \mathbb{O}T^{-1}, \quad (8.38)$$

con  $\tilde{\mathbb{O}}$  la matriz de controlabilidad del sistema en la forma canónica observable y  $\mathbb{O}$  la matriz de observabilidad en la representación en variables de estado original.

Luego, la transformación lineal que se debe aplicar para llevar un sistema en variables de estado a la forma canónica observable es

$$T = \tilde{\mathbb{O}}^{-1}\mathbb{O}, \quad (8.39)$$

por lo que se debe cumplir que el sistema sea observable para poder aplicar esta transformación.

## 8.5. Anexo E: Estabilidad de Modelos Difusos

Una característica importante a estudiar una vez que se ha diseñado el modelo difuso es su estabilidad. De esta manera es posible conocer mejor el proceso analizado, entregando una noción de la capacidad del sistema de permanecer relativamente cercano a un punto de equilibrio, o en su defecto converger a él.

La estabilidad de un modelo difuso discreto con consecuencias lineales puede ser estudiada en el sentido de Lyapunov, donde una condición suficiente para garantizar esta propiedad puede ser obtenida en términos del teorema de Lyapunov. Para ello, considérese un modelo difuso dado por el conjunto de reglas del tipo

$$R_r: \text{ Si } z_1(k) \text{ es } MF_1^r \text{ y } \dots \text{ y } z_p(k) \text{ es } MF_p^r \text{ entonces } x^r(k+1) = A_r x(k), \quad (8.40)$$

donde  $x(k)$  es el estado del sistema;  $R_r$  denota la  $r$ -ésima regla del modelo difuso, con  $r \in \{1, \dots, N_r\}$ , y  $N_r$  el número total de reglas;  $x^r(k+1)$  es su consecuencia o modelo local;  $z(k) = [z_1(k), \dots, z_p(k)]$  es el vector de premisas en el tiempo  $k$ ; y  $MF_i^r$  es el conjunto difuso (función de pertenencia) de la  $i$ -ésima premisa correspondientes a la  $r$ -ésima regla.

Cabe destacar que se ignora el término asociada a la entrada  $u(k)$  debido a que para el estudio de la estabilidad de un sistema dinámico ésta no influye. Luego, la salida del modelo difuso en variables de estado se infiere como

$$x(k+1) = \sum_{r=1}^{N_r} h_r(z(k)) \cdot A_r x(k), \quad (8.41)$$

con  $h_r(z(k))$  el grado de activación normalizado definido como en (2.2)-(2.3).

Para derivar una condición de estabilidad para el sistema en (8.41), primero se enuncia el teorema de Lyapunov para sistemas discretos.

Si se tiene un sistema discreto  $x(k+1) = f(x(k))$ , tal que  $f(0) = 0$ , y además existe una función escalar  $V(x(k))$  tal que:

- $V(0) = 0$ .
- $V(x(k)) > 0, \forall x(k) \neq 0$ .
- $\Delta V(x(k)) = V(x(k)) - V(x(k-1)) < 0, \forall x(k) \neq 0$ .

Entonces el estado  $x(k) = 0$  es un punto de equilibrio asintóticamente estable.

Luego, si se utiliza la función de Lyapunov  $V(x(k)) = x^T(k)Px(k)$ , con  $P$  una matriz positiva definida y  $x(k)$  dado por el sistema difuso en (8.40)-(8.41), entonces se tiene

que el equilibrio es asintóticamente estable si existe una matriz común  $P$  tal que cumpla la siguiente condición:

$$A_r^T P A_r - P < 0, \quad \forall r \in \{1, \dots, N_r\}. \quad (8.42)$$

En [108] se plantea un teorema para asegurar la condición de estabilidad, el cual dice que si existe una matriz  $P$  tal que (8.42) se satisface, y si se asume que  $A_r$  es estable y no singular, para  $r = 1, \dots, N_r$ ; entonces  $A_r A_q$  es estable para  $r, q = 1, \dots, N_r$ . Por lo tanto, el teorema anterior afirma que si alguna de las matrices  $A_r A_q$  es inestable, entonces no existe una matriz  $P$  tal que (8.42) se satisface. Luego, para estudiar la estabilidad de un modelo difuso con consecuencias lineales, basta asegurar la estabilidad de cada modelo local, y también corroborar la estabilidad de todas las combinaciones de  $A_r A_q$ .

Cabe destacar que el teorema exige que las matrices  $A_r$  sean no singulares, por lo que si alguna de ellas lo es, entonces no se puede analizar el sistema por este método. Por otro lado, el análisis de la estabilidad se remite al estudio en la representación de variables de estado del sistema, por lo que se debe expresar cada sistema local en dicha representación. Luego, es necesario aplicar algunas de las formas canónicas presentes en la Sección 8.4 de anexos, con lo cual se debe asegurar con anterioridad si es posible desarrollar dichas representaciones.

Otros enfoques se pueden encontrar en la literatura especializada, como el propuesto el 2003 por Tanaka et al. [109] donde introducen un nuevo término: *Fuzzy Lyapunov Function*. Esta función está definida de manera similar a la salida de un modelo difuso TS tradicional, vale decir, considera la suma ponderada por los grados de activación de cada modelo lineal local. A partir de dicha definición se pueden obtener ciertas condiciones que aseguran la estabilidad del modelo en lazo abierto, además se plantea una relajación de éstas condiciones considerando las propiedades que tienen las funciones de pertenencia. Siguiendo una línea similar, en [71] se propone la mejora de la función de Lyapunov difusa al incluir términos asociados a la derivada de la función de pertenencia. De esta manera, al imponer las condiciones de estabilidad de Lyapunov se forma un set de inecuaciones que incluyen términos asociados a la segunda derivada de las funciones de pertenencia, afirmando que el uso de mayor información relacionada con la variación temporal del modelo difuso TS puede ser beneficioso para el análisis de estabilidad. Si bien estos enfoques sirven para estudiar la estabilidad de un modelo difuso, éstos no entregan una restricción explícita para asegurar dicha condición, por lo que no es posible realizar una implementación directa en un algoritmo.

Por otro lado, en [93] se propone abordar el análisis de estabilidad de la misma manera que lo propuesto por Tanaka & Sugeno en [108], es decir, se busca utilizar el método directo de Lyapunov, el cual es aplicado a cada modelo local de la base de reglas de la representación difusa con el fin de encontrar una matriz  $P$  común que satisfaga las condiciones en cada caso. El aporte en [93] se basa en proponer un procedimiento para encontrar dicha matriz  $P$ , esto debido a que tal tarea se vuelve compleja para sistemas con gran cantidad de reglas. A raíz del método propuesto se logran obtener nuevas condiciones para la estabilidad del modelo difuso. La principal desventaja de este

---

enfoque radica en que la metodología detallada sólo es válida para sistemas de segundo orden.