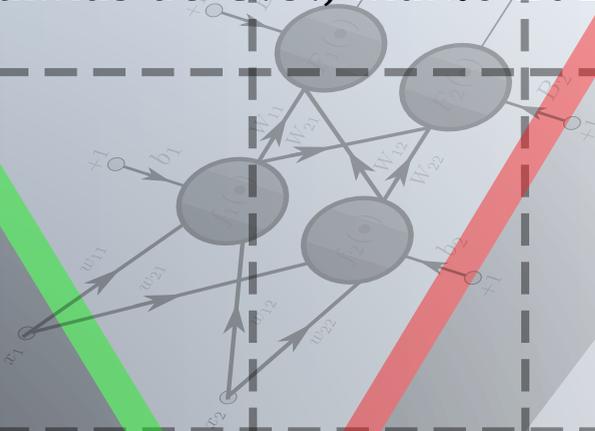
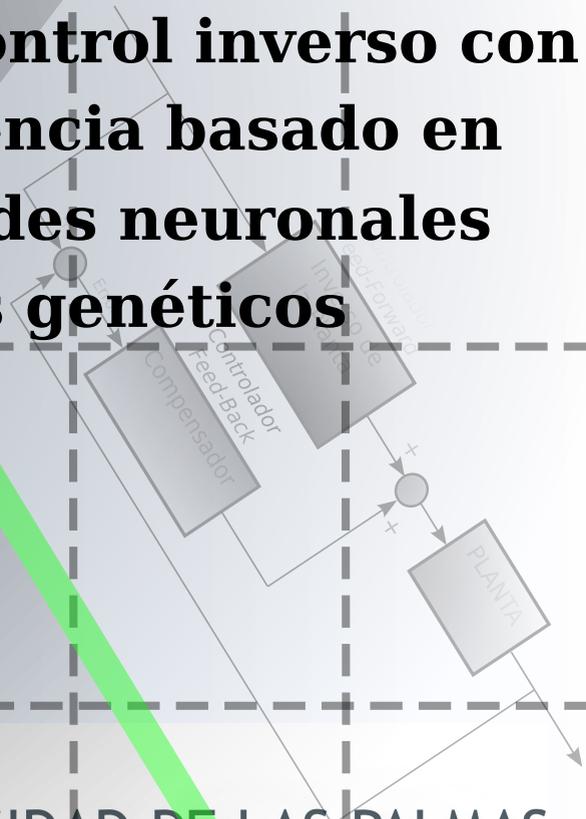


TESIS DOCTORAL
Carlos Betancor Martín
Las Palmas de G.C., marzo 2015



Aportaciones al control inverso con modelo de referencia basado en lógica borrosa, redes neuronales y algoritmos genéticos



**UNIVERSIDAD DE LAS PALMAS
DE GRAN CANARIA**

Departamento de Ingeniería Electrónica y Automática



**D ROBERTO ESPER-CHAÍN FALCÓN SECRETARIO DEL
DEPARTAMENTO DE INGENIERÍA ELECTRÓNICA Y
AUTOMÁTICA DE LA UNIVERSIDAD DE LAS PALMAS DE
GRAN CANARIA,**

CERTIFICA,

Que el Consejo de Doctores del Departamento en su sesión ordinaria de fecha de 7 de abril de 2015, tomó el acuerdo de dar el consentimiento para su tramitación, a la tesis doctoral titulada "Aportaciones al control inverso con modelo de referencia basado en lógica borrosa, redes neuronales y algoritmos genéticos" presentada por el doctorando D Carlos S. Betancor Martín y dirigida por los Doctores D. Aurelio Vega Martínez y D. Juan Antonio Montiel Nelson

Y para que así conste, y a efectos de lo previsto en el Artº 6 del Reglamento para la elaboración, defensa, tribunal y evaluación de tesis doctorales de la Universidad de Las Palmas de Gran Canaria, firmo la presente en Las Palmas de Gran Canaria, a 7 de abril de dos mil quince.



UNIVERSIDAD DE LAS PALMAS
DE GRAN CANARIA

Departamento: DEPARTAMENTO DE INGENIERÍA ELECTRÓNICA Y AUTOMÁTICA

Programa de Doctorado: INGENIERÍA DE TELECOMUNICACIÓN AVANZADA

Título de la Tesis

APORTACIONES AL CONTROL INVERSO CON MODELO DE
REFERENCIA BASADO EN LÓGICA BORROSA, REDES
NEURONALES Y ALGORITMOS GENÉTICOS

Doctorando

Fdo.: D. Carlos S. Betancor Martín

Director

Fdo.: Dr. D. Aurelio Vega Martínez

Director

Fdo.: Dr. D. Juan A. Montiel Nelson

Las Palmas de G.C., 26 de marzo de 2015

*A mi madre, María Encarnación,
y en memoria de mi padre, Carlos.
A mi mujer, Montserrat Bernal.
A mi hija, Montserrat Betancor.*

*“Nunca consideres el estudio como una obligación,
sino como una oportunidad para penetrar en
el bello y maravilloso mundo del saber“*

Albert Einstein (1879 – 1955)

Agradecimientos

Después de largos años de estudio, llega el momento final de hacer resumen y hacer una mención especial a aquellas personas que de una u otra forma me han dado consejos, fuerza, ánimo y apoyo para que esta tesis se llegara a culminar.

Quisiera agradecer al Catedrático Dr. D. Juan Antonio Montiel Nelson, por todo el esfuerzo que ha realizado para la concepción y buen desarrollo de este proyecto. Quisiera también agradecerle la inversión de tiempo que realizó para la buena elaboración de las publicaciones y documento de tesis. También gracias por todos los consejos recibidos. Sus continuos mensajes y su apoyo fueron fundamentales.

Al Dr. D. Aurelio Vega Martínez, por su inestimable apoyo y consejos a la hora de redactar el documento de esta tesis.

A mi compañero D. Carlos Javier Sosa por la ayuda y consejos dados. También a mi compañera D^a. Margarita Marrero Martín por sus aportes y por su interés en la evolución de mi trabajo.

Agradecer, por supuesto, el continuo apoyo y cariño durante tantos años de mi familia.

En especial, muchas gracias a mi mujer Montse, por su continuo ánimo y confianza en mí, por apoyarme, comprenderme y darme fuerzas. Su compañía, cariño y el amor dado en todo momento han sido fundamentales. Sólo ella sabe todo lo que he tenido que sacrificar para llegar al final de esta tesis.

Índice general

Índice general	I
Índice de tablas	V
Índice de figuras	VII
Acrónimos	XIII
Resumen	XV
1. Introducción	1
1.1. Planteamiento del problema y motivación	1
1.2. Objetivos de la investigación	2
1.3. Estructura del sistema de control	4
1.4. Metodología de diseño	6
1.5. Organización	10
2. Fundamentos	12
2.1. Control con redes neuronales	13
2.1.1. Introducción	13
2.1.2. Modelo de la neurona artificial	13
2.1.3. Tipos de redes neuronales artificiales	15
2.1.4. Control neuronal con modelo de referencia	16
2.1.5. Control neuronal inverso directo	16
2.2. Controladores borrosos	17
2.2.1. Los conjuntos borrosos	18
2.2.2. Estructura de un controlador borroso	19
2.2.3. Proceso en Takagi-Sugeno	24
2.2.4. PID borroso	25
2.2.5. PI borroso	25

2.3.	Modelado de la planta	27
2.3.1.	Introducción	27
2.3.2.	Modelo borroso Takagi-Sugeno	28
2.3.3.	Agrupamiento borroso: Gustafson-Kessel	29
2.3.4.	Índice VAF	32
2.4.	El algoritmo genético	32
2.4.1.	Conceptos básicos del algoritmo genético	32
2.4.2.	Representación, tamaño y población inicial	33
2.4.3.	Funciones de evaluación y aptitud	34
2.4.4.	Operadores genéticos	35
3.	Estado del arte	37
3.1.	El control inverso	37
3.1.1.	El concepto de inversión	38
3.1.2.	Realizabilidad del inverso	38
3.1.3.	Inversión de sistemas de fase mínima	40
3.1.4.	Inversión de sistemas de fase no mínima	41
3.1.5.	Algoritmos de inversión aproximada para sistemas de fase no mínima	42
3.2.	El control Feed Forward-Feed Back	44
3.2.1.	Introducción	44
3.2.2.	Aplicaciones	45
3.3.	Control basado en modelo de referencia	49
3.3.1.	Estructura por modelo de referencia	49
3.3.2.	Control inverso adaptativo con modelo de referencia	50
4.	Control de un motor de corriente continua	52
4.1.	El motor DC PM	53
4.2.	Identificación de la planta	54
4.3.	Modelo de referencia	56
4.4.	Diseño del controlador inverso	59
4.5.	Comportamiento del sistema controlado	61
4.6.	Diseño del controlador borroso	65
4.7.	Aportaciones	69
5.	Compensación de perturbaciones	71
5.1.	Perturbación de carga	72
5.2.	La función objetivo	74
5.3.	Modelo de rechazo de la perturbación	76
5.4.	Compensación de la perturbación del par de carga	78

5.4.1.	Introducción	78
5.4.2.	Diseño del PI borroso	80
5.4.3.	Modelo de rechazo elegido	82
5.4.4.	Parámetros del algoritmo genético	83
5.4.5.	Resultados obtenidos	88
5.5.	Sistema motor-generator	91
5.6.	Aportaciones	93
6.	Control de sistemas lineales	96
6.1.	Planta LTI de segundo orden	97
6.1.1.	Descripción e identificación de la planta	97
6.1.2.	Generación de la red neuronal	99
6.1.3.	Comportamiento del sistema controlado	99
6.1.4.	Estabilidad y robustez ante errores de modelado	103
6.2.	Planta LTI de tercer orden	104
6.2.1.	Descripción e identificación de la planta	104
6.2.2.	Comportamiento del sistema controlado	107
6.3.	Aportaciones	109
7.	Control de no linealidades continuas	111
7.1.	Sistema de segundo orden no lineal	113
7.1.1.	Sistema masa-resorte-amortiguador	113
7.1.2.	Resultados para diversas estrategias	114
7.1.3.	Identificación de la planta	120
7.1.4.	Diseño del controlador inverso	125
7.1.5.	Diseño del sistema supervisor	126
7.1.6.	Resultados obtenidos	128
7.1.7.	Cancelación de los errores en régimen permanente	135
7.2.	Control del nivel de líquido en tanques cónicos	140
7.2.1.	Introducción	140
7.2.2.	Modelado del tanque cónico	143
7.3.	Tanque cónico con descarga lenta	144
7.3.1.	Respuesta para diversas técnicas	144
7.3.2.	Identificación del tanque	146
7.3.3.	Diseño del controlador inverso	150
7.3.4.	Resultados obtenidos	152
7.3.5.	Estabilidad y robustez del sistema	154
7.3.6.	Comportamiento frente a perturbaciones	156
7.3.7.	Diseño de un controlador predictivo no lineal	157
7.4.	Tanque cónico con descarga rápida	163

7.4.1.	Respuesta con controladores PI	163
7.4.2.	Identificación del tanque	163
7.4.3.	Diseño del controlador inverso	165
7.4.4.	Resultados obtenidos	165
7.4.5.	Estabilidad y robustez del sistema	166
7.4.6.	Comportamiento frente a perturbaciones	169
7.5.	Aportaciones	169
8.	Control de una no linealidad discontinua	172
8.1.	Zona muerta	173
8.2.	Modelado de la zona muerta	176
8.3.	Motor con zona muerta simétrica	178
8.3.1.	Descripción del sistema	178
8.3.2.	Resultados con un GPC	178
8.3.3.	Identificación de la planta	179
8.3.4.	Diseño del controlador inverso	181
8.3.5.	Resultados obtenidos	182
8.3.6.	Entrenamiento de la red neuronal	184
8.4.	Motor con zona muerta asimétrica	188
8.4.1.	Descripción del sistema	189
8.4.2.	Resultados con PD y controlador adaptativo robusto	190
8.4.3.	Identificación de la planta	190
8.4.4.	Diseño del controlador inverso	194
8.4.5.	Resultados obtenidos	194
8.5.	Aportaciones	196
9.	Control de un convertidor conmutado DC/DC	199
9.1.	El convertidor DC/DC	200
9.2.	Modelo del convertidor conmutado reductor DC/DC	201
9.3.	Convertidor DC/DC reductor	203
9.3.1.	Diseño del sistema de control	204
9.3.2.	Simulaciones y resultados	206
9.4.	Aportaciones	211
	Conclusiones y líneas futuras	214
	Bibliografía	217

Índice de tablas

2.1. ESTRUCTURAS PID BORROSAS PRÁCTICAS	26
4.1. PARÁMETROS DEL ALGORITMO DE AGRUPAMIENTO	55
5.1. MÍNIMOS DE ISEOISRAE PARA 6000 EVALUACIONES	85
5.2. TÉCNICAS DE SELECCIÓN	86
5.3. TÉCNICAS DE CRUCE	86
5.4. MÍNIMOS DE ISEOISRAE VARIANDO EL MÉTODO DE CRUCE	88
5.5. MÍNIMOS DE ISEOISRAE VARIANDO LA PROBABILIDAD DE CRUCE	89
5.6. MÍNIMOS DE ISEOISRAE VARIANDO LA PROBABILIDAD DE MUTACIÓN	91
5.7. TIEMPO DE ESTABLECIMIENTO Y MÁXIMO SOBREPULSO . . .	94
6.1. PARÁMETROS PARA LA IDENTIFICACIÓN DE LA PLANTA DE SE- GUNDO ORDEN	98
6.2. PARÁMETROS PARA LA IDENTIFICACIÓN DE LA PLANTA DE TERCER ORDEN	105
7.1. PARÁMETROS PARA LA IDENTIFICACIÓN DEL SISTEMA NO LINEAL	122
7.2. VALORES FINALES PARA EL ALGORITMO DE AGRUPAMIENTO .	124
7.3. PARÁMETROS DEL ALGORITMO DE AGRUPAMIENTO	147
7.4. PARÁMETROS DEL ALGORITMO DE AGRUPAMIENTO	150
7.5. ÍNDICE DEL ORDEN PARA DIFERENTES RETRASOS m Y n . . .	160
7.6. NSSE PARA NOE Y DIFERENTE NÚMERO DE ITERACIONES . .	160
7.7. NSSE PARA NARX Y DIFERENTE NÚMERO DE ITERACIONES .	161
7.8. PARÁMETROS DEL ALGORITMO DE AGRUPAMIENTO	165
7.9. TIEMPO DE ESTABLECIMIENTO EN SEGUNDOS	166

8.1. PARÁMETROS PARA LA IDENTIFICACIÓN DEL MOTOR CON ZONA MUERTA SIMÉTRICA	180
8.2. VALORES FINALES ALCANZADOS EN RADIANES	184
8.3. VALORES MÍNIMOS DE IAE PARA DIFERENTES TASAS DE MUTACIÓN	186
8.4. VALORES MÍNIMOS DE IAE PARA DIFERENTES TASAS DE RE-INSERCIÓN	187
8.5. VALORES MÍNIMOS DE IAE PARA DIFERENTES METODOLOGÍAS DE CRUCE	187
8.6. VALORES MÍNIMOS DE IAE PARA DIFERENTES TASAS DE CRUCE	188
8.7. PARÁMETROS PARA LA IDENTIFICACIÓN DEL MOTOR CON ZONA MUERTA ASIMÉTRICA	193
9.1. PARÁMETROS PARA LA IDENTIFICACIÓN DEL CONVERTIDOR . .	204
9.2. TIEMPO DE ESTABLECIMIENTO Y SOBREPULSO	207
9.3. COMPARACIONES DE LOS ÍNDICES DE PRESTACIONES	207

Índice de figuras

1.1. Diagrama de bloques del sistema de control.	4
1.2. Control híbrido <i>Feed Forward-Feed Back</i>	5
1.3. Control por modelo de referencia.	5
1.4. Control inverso directo.	5
1.5. Fases y metodología de diseño del sistema de control.	7
2.1. Modelo de neurona estándar.	14
2.2. Perceptrón multicapa.	16
2.3. Control con modelo de referencia.	16
2.4. Control inverso directo.	17
2.5. Ejemplo de conjuntos borrosos.	18
2.6. Estructura típica de un controlador borroso.	19
2.7. Grados de pertenencia en el antecedente.	21
2.8. Implicación de Mamdani.	22
2.9. Conjunto borroso por el método del máximo.	23
2.10. Ejemplo con reglas Takagi-Sugeno.	25
2.11. Estructura del PI borroso.	26
3.1. Problema fundamental de inversión.	38
3.2. Control <i>Feed Forward</i>	44
3.3. Control híbrido <i>Feed Forward-Feed Back</i>	45
3.4. Control por modelo de referencia.	50
3.5. Control inverso adaptativo.	50
3.6. Control inverso adaptativo con modelo de referencia.	51
3.7. Control para la cancelación del ruido.	51
4.1. Señales para la identificación.	55
4.2. Resultado de la validación.	56
4.3. Sistema para calcular el modelo de referencia.	57
4.4. Sistema en lazo cerrado.	58

4.5. Respuesta del motor.	58
4.6. Diagrama de bloques del sistema de partida.	59
4.7. Sistema para el control en lazo abierto.	60
4.8. Sistema para el control en lazo cerrado.	60
4.9. Neurona que implementa a $u(k)$	61
4.10. Respuesta del sistema controlado.	62
4.11. Error que se produce en el seguimiento.	62
4.12. Respuesta con errores de modelado y 10 rad/s	63
4.13. Respuesta con errores de modelado con $T = 0.01 \text{ s}$	63
4.14. Señal de control con 10 rad/s para $T = 0.1 \text{ s}$ y $T = 0.01 \text{ s}$	64
4.15. Comportamiento para $T = 0.01 \text{ s}$ y $t_e = 0.05 \text{ s}$	65
4.16. Señal de control para $T = 0.01 \text{ s}$ y $t_e = 0.05 \text{ s}$	65
4.17. Estructura del PI borroso.	66
4.18. PI borroso con conjuntos triangulares.	67
4.19. PI borroso con conjuntos <i>gaussianos</i>	67
4.20. Zona en la que se activa cada una de las reglas.	68
4.21. Comportamiento para $T = 0.01 \text{ s}$ y $t_e = 0.05 \text{ s}$	69
5.1. PI borroso en la realimentación.	73
5.2. Estructura del PI borroso.	73
5.3. Posibles respuestas ante una perturbación en escalón.	74
5.4. Salida deseada y salida real.	76
5.5. Respuesta de los modelos (5.2) y (5.3).	77
5.6. Forma de modelar la perturbación de carga.	79
5.7. Comportamiento ante una perturbación de carga.	79
5.8. Superficie de control del PI borroso.	80
5.9. Conjuntos borrosos de las entradas y salida.	81
5.10. Región en la que se activa cada una de las reglas.	82
5.11. Función objetivo al variar la reinsertión.	84
5.12. Valores mínimos y medios de la función objetivo.	84
5.13. Función objetivo al variar la metodología de cruce.	87
5.14. Valores mínimos y medios para <i>reclin</i> y <i>xovdp</i>	87
5.15. Función objetivo con diferentes probabilidades de cruce.	88
5.16. Valores mínimos y medios al variar la probabilidad de cruce.	89
5.17. Mínimos para diferentes probabilidades de mutación.	90
5.18. Valores mínimos y medios al variar la mutación.	90
5.19. Rechazo de la perturbación para diversas funciones objetivos.	91
5.20. Respuesta final del sistema.	92
5.21. Respuesta ante la perturbación por el método CDM.	93
5.22. Respuesta ante la perturbación con el PI borroso.	93

6.1. Señales para la identificación.	98
6.2. Resultado de la validación.	99
6.3. Respuestas con dos controles neuroborrosos y un <i>lead-lag</i> . . .	100
6.4. Señal de salida para el sistema controlado.	101
6.5. Señal de control que se aplica a la planta.	101
6.6. Estructura base del sistema de control propuesto.	101
6.7. Salida del sistema controlado para t_e de 1 y 2 segundos.	102
6.8. Señal de control que se aplica a la planta.	103
6.9. Salidas para diferentes modelos de planta.	104
6.10. Señales para la identificación.	105
6.11. Sistema de control propuesto en cadena abierta.	106
6.12. Respuestas con dos controles neuroborrosos y un PID	107
6.13. Respuestas del sistema controlado.	108
6.14. Respuesta del sistema ante errores de modelado.	109
7.1. Sistema masa-resorte-amortiguador.	113
7.2. Control inverso directo	114
7.3. Control inverso directo con filtro paso-bajo	115
7.4. Control inverso mediante entrenamiento especializado	116
7.5. Control por linealización mediante realimentación	116
7.6. Control con un PID	117
7.7. Control PID con retroalimentación de la referencia	118
7.8. Control óptimo	118
7.9. Control con polo que cancela el cero	119
7.10. Control con polo sin cancelación del cero	119
7.11. Control predictivo no lineal	120
7.12. Control predictivo generalizado aproximado	121
7.13. Señales para la identificación.	121
7.14. VAF obtenido al variar el número de clústers.	123
7.15. VAF obtenido al variar el coeficiente de borrosificación.	123
7.16. Resultado de la validación para 3 clústers.	125
7.17. Integración de los subsistemas.	127
7.18. Caso en el que se activa la regla 1.	128
7.19. Reglas 2 y 3 activas y método del mínimo para la <i>and</i>	128
7.20. Reglas 2 y 3 activas y método del producto para la <i>and</i>	129
7.21. Respuestas obtenidas para 3, 6 y 10 clústers.	129
7.22. Errores para 3, 6 y 10 clústers.	130
7.23. Señales de control generadas para 3, 6 y 10 clústers.	130
7.24. Ampliación de la figura 7.23 entorno a los 350 s.	131
7.25. Respuestas para $T = 0.1$ s y $T = 0.2$ s.	132

7.26. Ampliación de la figura 7.25 en $t = 100$ s.	132
7.27. Ampliación de la figura 7.25 en $t = 400$ s.	133
7.28. Señales de control para $T = 0.1$ s y $T = 0.2$ s.	133
7.29. Ampliación de la figura 7.28 en $t = 400$ s.	134
7.30. Identificación con $T = 0.2$ s y control con $T = 0.1$ s.	134
7.31. Ampliación de la figura 7.30 en $t = 400$ s.	135
7.32. Señal de control para diferentes muestreos.	135
7.33. Ampliación de la figura 7.32 en $t = 400$ s.	136
7.34. Respuestas para $T = 0.2$ s.	136
7.35. Señales de control para $T = 0.2$ s.	137
7.36. Respuestas para diferentes muestreos.	137
7.37. Señales de control para diferentes muestreos.	138
7.38. Respuesta para 3 clústers y PI borroso con $T = 0.2$ s.	138
7.39. Error para 3 clústers y PI borroso con $T = 0.2$ s.	139
7.40. Señal de control para 3 clústers y PI borroso con $T = 0.2$ s.	139
7.41. Tanque cónico.	143
7.42. Respuesta del GPC para el punto P_1	145
7.43. Respuesta del GPC para el punto P_2	145
7.44. Respuesta con el control inverso directo	146
7.45. Señales para la identificación.	147
7.46. VAF al variar el .m con las señales de la figura 7.45.	148
7.47. Resultado de la validación para el modelo (7.14).	148
7.48. Señales para la identificación.	149
7.49. VAF al variar el .m con las señales de la figura 7.48.	151
7.50. Resultado de la validación para el modelo (7.15).	151
7.51. Respuesta con el modelo Takagi-Sugeno (7.14).	153
7.52. Respuesta con el modelo Takagi-Sugeno (7.15).	153
7.53. Respuesta con el modelo Takagi-Sugeno (7.15) y $T = 0.01$ s.	154
7.54. Respuesta al variar k_{out}	155
7.55. Respuesta al variar α en -1.4°	155
7.56. Respuesta al variar α en -10°	156
7.57. Respuesta al variar α en $+4^\circ$	156
7.58. Respuesta al variar α a 40° y 60°	157
7.59. Respuesta para perturbación entorno a $h = 0.2m$	157
7.60. Respuesta para perturbación entorno a $h = 0.4m$	158
7.61. Índice del orden para retrasos de entrada y salida iguales.	159
7.62. Índice del orden para diferentes retrasos de entrada y salida.	159
7.63. Validación para 6 neuronas y 1000 iteraciones.	161
7.64. Respuesta con 6 neuronas, $N = 7$, $N_u = 5$ y $\lambda = 0.01$	162

7.65. Respuesta con 9 neuronas, $N = 30$, $N_u = 20$ y $\lambda = 0.1$	162
7.66. Respuesta con 6 neuronas y $T = 1$ ms.	163
7.67. Respuesta con los dos PI	164
7.68. Respuesta del control inverso con modelo de referencia.	166
7.69. Respuesta al variar k_{out} un $\pm 50\%$	167
7.70. Respuesta al variar k_{out} entre 0 y 100 cm^2/s	167
7.71. Respuesta al variar R a 69.25 cm y H a 18 cm.	168
7.72. Respuesta al variar R a 9.25 cm y H a 128 cm.	168
7.73. Respuesta con perturbación entorno a $h = 5$ cm.	169
7.74. Respuesta con perturbación entorno a $h = 25$ cm.	170
8.1. Zona muerta en un sistema.	176
8.2. Función no lineal de la zona muerta.	176
8.3. Servosistema de posicionamiento con zona muerta.	177
8.4. Modelo empleado para el motor.	178
8.5. Respuesta del motor con un GPC	179
8.6. Señales para la identificación.	180
8.7. Señales para la identificación.	181
8.8. Respuestas a un cambio escalón unitario.	182
8.9. Respuestas en régimen permanente.	183
8.10. Señales de control para los controladores 1 y 3.	183
8.11. Respuesta al variar el ancho de la zona muerta.	184
8.12. Respuesta para varios cambios de ángulo.	185
8.13. Mínimos de IAE según el tamaño de la población.	186
8.14. Respuesta con la red neuronal sintonizada.	188
8.15. Respuesta ante errores de modelado.	189
8.16. Modelo empleado para el motor.	189
8.17. Error con un PD sin compensación de la zona muerta	190
8.18. Error con un compensador adaptativo robusto	191
8.19. Salida con el controlador adaptativo robusto	191
8.20. Señal de control con el controlador adaptativo robusto	192
8.21. Error para una perturbación de -0.4 V en el instante 2 s	192
8.22. Señales para la identificación.	193
8.23. Salidas del modelo de referencia y del sistema.	194
8.24. Error en el seguimiento de la trayectoria.	195
8.25. Señal de control para minimizar el error.	195
8.26. Error para incertidumbre de la zona muerta.	196
8.27. Respuesta para perturbación de -22 V a los 10 s.	197
8.28. Errores para diferentes valores de J y B	197

9.1. Convertidor conmutado reductor DC/DC.	201
9.2. Señal PWM.	202
9.3. Modelo para CCM y estado ON.	202
9.4. Modelo para CCM y estado OFF.	202
9.5. Modelo para DCM cuando la corriente es cero.	203
9.6. Señales para la identificación.	204
9.7. Respuesta para un cambio de 0 V a 6 V.	206
9.8. Evolución de los índices de prestaciones.	207
9.9. Respuesta para variaciones en la consigna y carga.	208
9.10. Respuesta con QPI y carga de 2 Ω	208
9.11. Respuesta con QPI y carga de 50 Ω	208
9.12. Respuesta con FC y carga de 2 Ω	209
9.13. Respuesta con FC y carga de 50 Ω	209
9.14. Respuesta con ANN y carga de 2 Ω	209
9.15. Respuesta con ANN y carga de 50 Ω	210
9.16. Respuesta para $R_L = 1 \Omega$ y $R_L = 0.7 \Omega$	210
9.17. Respuesta para $R_L = 1 \Omega$ y $R_L = 0.7 \Omega$	210
9.18. Respuesta para variaciones de referencia y de carga.	211
9.19. Rechazo a las perturbaciones de carga de la figura 9.18.	212
9.20. Corriente por la bobina I_L para el caso de la figura 9.18.	212

Acrónimos

ANN	<i>Artificial Neural Network</i>
CCM	<i>Continuous Conduction Mode</i>
CDM	<i>Coefficient Diagram Method</i>
COG	<i>Centre Of Gravity</i>
DC	<i>Direct Current</i>
DCM	<i>Discontinuous Conduction Mode</i>
DOF	<i>Degree Of Freedom</i>
DSM	<i>Direct Synthesis Method</i>
FB	<i>Feed Back</i>
FC	<i>Fuzzy Control</i>
FF	<i>Feed Forward</i>
FMID	<i>Fuzzy Modeling and IDentification</i>
FOPTD	<i>First Order Plus Time Delay</i>
GA	<i>Genetic Algorithm</i>
GPC	<i>Generalized Predictive Controller</i>
GSM	<i>Gain Scheduling Method</i>
IAE	<i>Integral of Absolute Error</i>
IMC	<i>Internal Model Controller</i>

ISE	<i>Integral Square Error</i>
ISEOISRAE	<i>Integral Square Error Or Integral Square Root of Absolute Error</i>
ITAE	<i>Integral of Time multiplied by Absolute Error</i>
ITSE	<i>Integral of Time multiplied by the Squared Error</i>
LTI	<i>Linear Time Invariant</i>
MIMO	<i>Multi Input Multi Output</i>
MOSFET	<i>Metal Oxide Semiconductor Field Effect Transistor</i>
MPC	<i>Model Predictive Control</i>
NEFCON	<i>NEural Fuzzy CONtrol</i>
NGPC	<i>Nonlinear Generalized Predictive Control</i>
NPZI	<i>Nonminimum-Phase Zeros Ignore</i>
NSSE	<i>Normalized Sum of Squared Errors</i>
PD	<i>Proportional Derivative</i>
PI	<i>Proportional Integral</i>
PID	<i>Proportional Integral Derivative</i>
PM	<i>Permanent Magnet</i>
PSO	<i>Particle Swarm Optimization</i>
PWM	<i>Pulse Width Modulation</i>
QPI	<i>Quadratic Performance Index</i>
SISO	<i>Single Input Single Output</i>
VAF	<i>Variance Accounted For</i>
ZMETC	<i>Zero-Magnitude-Error Tracking Controller</i>
ZPETC	<i>Zero-Phase-Error Tracking Controller</i>

Resumen

Debido al interés que existe actualmente en combinar el control de prealimentación —*Feed Forward* (FF)— con el clásico de realimentación —*Feed Back* (FB)—, el presente trabajo tiene como finalidad marcar una estrategia de diseño de un sistema que incorpora a estos dos tipos de control. El controlador FF implementa un inverso aproximado de la planta y el FB se encarga de complementar, cuando fuese necesario, la tarea del FF para hacer al sistema más robusto.

Para el diseño del FF se opta por representar a la planta con modelos locales lineales y para ello se modela de forma borrosa mediante el tipo Takagi-Sugeno, obtenido a partir del algoritmo de agrupamiento Gustafson-Kessel modificado. Este modelo borroso junto con un modelo de referencia, en el que se especifica el comportamiento en régimen transitorio y permanente deseado del sistema final, da lugar al inverso aproximado. Parte del trabajo de investigación se centra en buscar los parámetros del algoritmo de agrupamiento para, en el proceso de identificación, obtener el modelo Takagi-Sugeno que de lugar posteriormente a un inverso aproximado que sirva como controlador FF y el cual se implementa mediante redes neuronales.

El controlador FB se diseña como un controlador borroso, concretamente con un controlador proporcional integral —*Proportional Integral* (PI)— borroso. Su misión es hacer al sistema más robusto ante las incertidumbres de modelado de la planta y ante las perturbaciones en el sistema. Para llevar a cabo estas dos labores a la vez con un único PI borroso, minimizando el hardware necesario y reduciendo el coste computacional, el trabajo se centra en sintonizar de forma correcta a este PI borroso. Se muestra mediante diversas aplicaciones que el controlador diseñado lleva a cabo la labor encomendada de forma totalmente satisfactoria.

En ciertas aplicaciones, y con el objetivo de conseguir un mejor desempeño del sistema, se incluye un algoritmo genético —*Genetic Algorithm* (GA)— que, fuera de línea, sintoniza las ganancias del PI borroso y los pesos de la red neuronal. En este aspecto, el trabajo consiste en investigar cómo los distintos operadores involucrados en un GA afectan a la respuesta final.

Se aplica la metodología propuesta a diferentes tipos de plantas lineales y no lineales, realizando comparaciones con otras técnicas de control. Se observa la estabilidad y robustez del sistema ante la variación de diferentes parámetros y ante perturbaciones, resultando un sistema muy estable y robusto. Se concluye que el sistema y la metodología propuesta dan como resultado mejores prestaciones que otras técnicas más complejas publicadas.

Introducción

En este capítulo de introducción, y para explicar la motivación que ha promovido el desarrollo de este trabajo, se plantea inicialmente el problema objeto de estudio, para posteriormente exponer los objetivos que se persiguen en esta investigación. Para situar al lector desde el inicio, se presenta y explica el esquema de bloques del sistema de control que se va a diseñar y la metodología de diseño propuesta que se seguirá, enlazando cada una de las fases del diseño con las diferentes secciones de los capítulos que aparecen en este documento.

En el siguiente capítulo se expondrán algunos campos de conocimiento que entrarán a formar parte de la estructura del sistema y de la metodología propuesta.

El capítulo está organizado como sigue. En la sección inicial 1.1 se expone el problema objeto de estudio y la motivación. Se sigue, en la sección 1.2, aclarando los objetivos de este trabajo de investigación. La estructura del sistema de control se explica en la sección 1.3 y la metodología de diseño en la siguiente sección 1.4. Por último, se describe la organización de este documento de tesis en la sección 1.5.

Sección 1.1

Planteamiento del problema y motivación

En el problema del diseño de los sistemas de control está implícito el concepto de inversión. En general, podría decirse que todos los controladores generan de alguna forma una inversa del proceso, tan exacta como sea posible,

es decir, una inversa aproximada. Los mecanismos usados, para generar esta inversa aproximada, son los que marcan las diferencias entre los controladores obtenidos.

En el control inverso directo, el controlador debería ser un modelo inverso perfecto de la planta, de manera que la salida del sistema siguiera a la consigna, o referencia impuesta al sistema, de modo exacto. En la práctica, sin embargo, hay una serie de factores que hacen que obtener un inverso perfecto sea imposible, por lo que se recurre a obtener un inverso aproximado.

Por otra parte, es mejor seguir una versión suavizada de la consigna, como en el caso de un escalón. Esta versión suavizada se genera con un modelo de referencia, de forma que la salida del sistema controlado sea igual a la salida del modelo de referencia. Dicha respuesta deseada va a depender del tipo de proceso que se vaya a controlar y sus parámetros se establecen en función de las especificaciones deseadas en régimen permanente y transitorio del sistema final controlado.

Además, en un sistema controlado hay que tener en cuenta los errores de modelado de la planta y las perturbaciones. Ante estos dos factores, el sistema controlado deberá ser estable y robusto, minimizando el efecto producido en la salida por estos agentes adversos.

La motivación de este trabajo de investigación surge para unificar lo expuesto anteriormente en un sistema sencillo y que sea versátil a la hora de poder aplicarlo a un amplio espectro de tipos de plantas.

Sección 1.2

Objetivos de la investigación

En esta tesis se generará el inverso aproximado del proceso a partir de un modelo borroso de la planta a controlar y de un modelo de referencia. El modelo borroso se obtendrá utilizando un algoritmo de identificación por agrupamiento borroso, resultando un modelo borroso Takagi-Sugeno. De esta forma, se obtiene un modelado de la planta basado en modelos locales lineales y el inverso aproximado será obtenido mediante la integración de la inversión de cada modelo local. Por otro lado, el modelo de referencia también va a ayudar a conseguir el inverso aproximado, que mediante un método de diseño directo se implementará con una red neuronal, actuando como controlador de prealimentación. La red neuronal quedará a disposición para ser entrenada, si fuese necesario, de acuerdo a algún tipo de criterio, y así optimizar aún más la respuesta. Dicho entrenamiento se llevaría a cabo fuera de línea y correría a cargo de un algoritmo genético, siendo esta una solución ventajosa cuando

la plataforma disponible en tiempo real impide soluciones complicadas.

Otro de los objetivos de la tesis es que el sistema sea estable y robusto ante errores de modelado y ante perturbaciones. Para ello, y dependiendo del tipo de planta a controlar, se diseñará un controlador borroso que se situará en la cadena de realimentación, de forma que modifique la señal de control que se aplica a la planta procedente del controlador que está actuando como inverso aproximado. Para un correcto ajuste de algunos parámetros del controlador borroso se hará uso de los algoritmos genéticos. Se investigarán los parámetros asociados a la lógica borrosa del controlador borroso que mejores prestaciones ofrecen, así como los operadores involucrados, en las distintas fases del algoritmo genético, que dan mejores resultados para la correcta sintonización del controlador borroso.

Para observar el rendimiento del sistema propuesto, se escogen plantas que han sido analizadas y controladas con diferentes técnicas de control, que en muchos casos son más complejas, y así realizar las comparativas pertinentes y deducir las ventajas y desventajas del sistema propuesto.

La simplicidad, la exactitud, la estabilidad, la robustez y la versatilidad del sistema. Estas serían las características principales con las que se podrían resumir los objetivos con el enfoque que se propone en esta tesis para el diseño del sistema de control.

La simplicidad, se refiere tanto a la complejidad computacional como a su implementación hardware, de forma que sería válido para sistemas embebidos en tiempo real. Aunque existen aplicaciones que no requieren una exactitud elevada, sí que existen otras que demandan una elevada exactitud en el valor de la variable controlada. Existen técnicas de control que son incompatibles con estos dos principios de simplicidad y exactitud, como por ejemplo el control predictivo en el que se requiere un esfuerzo computacional elevado.

Por otra parte, la estabilidad es importante en el sentido de que la variable controlada permanezca con un valor limitado, para una entrada limitada, cuando se producen variaciones indeseadas en el sistema controlado. Todas las técnicas de control persiguen este objetivo y lo satisfacen para cierto rango de variación de parámetros en el sistema. Cuanto mayor sea ese rango, más estable será el sistema obtenido.

La robustez, sin embargo, es un indicativo de como el control se adapta a cambios en el sistema y sigue ofreciendo una salida muy parecida hasta antes de que se produjera el cambio. El control robusto es una de las técnicas que existen para conseguir este objetivo.

La versatilidad indica que la metodología que se propone sirva para un amplio espectro de tipos de plantas. Como es lógico, es imposible que una

misma técnica se pueda aplicar a todos los tipos de procesos existentes, pero sí es interesante que esa técnica sea útil para muchos tipos de plantas. En este caso, la metodología planteada se aplicará a procesos de una entrada y una salida. Se desarrollará y probará en plantas lineales continuas, plantas que poseen no linealidades continuas y suaves, también en las que tienen fuertes no linealidades continuas, con cierto tipo de no linealidades discontinuas y a un caso particular de planta no lineal y variante en el tiempo.

Sección 1.3

Estructura del sistema de control

El esquema de bloques del sistema de control a desarrollar en esta tesis es el mostrado en la figura 1.1. Consta de un *Controlador Inverso* de prealimentación —*Feed Forward* (FF)— que actuará como un inverso aproximado de la planta y un *Controlador Borroso* de realimentación —*Feed Back* (FB)— que complementará al controlador de prealimentación para cancelar ciertos efectos que no se han tenido en cuenta en el diseño del *Controlador Inverso*.

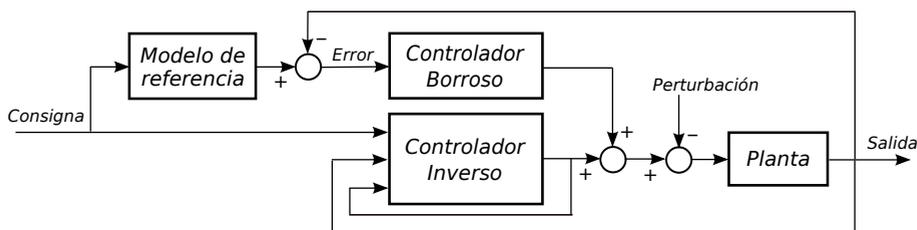


Figura 1.1: Diagrama de bloques del sistema de control.

Como se puede observar es un control híbrido FF-FB, que será tratado en el capítulo 3, sección 3.2 (figura 1.2). Sin embargo, tiene ciertas diferencias, ya que al controlador FF no sólo se le introduce la señal de consigna, sino que también se le proporciona información de la salida de la planta y de la señal de control. En este sentido es un esquema similar a un sistema de control basado en modelo de referencia, que será tratado en el capítulo 3, sección 3.3 (figura 1.3), pero en este caso la señal de error, resultante de la diferencia entre la salida del modelo de referencia y la salida de la planta, no se le aplica al controlador FF sino al controlador FB.

Debido a que el controlador FF se va a comportar como un inverso aproximado de la planta y además será implementado mediante redes neuronales, parte de la estructura planteada pertenece al esquema de control inverso direc-

to con redes neuronales, que se explicará en el capítulo 2, sección 2.1.5 (figura 1.4).

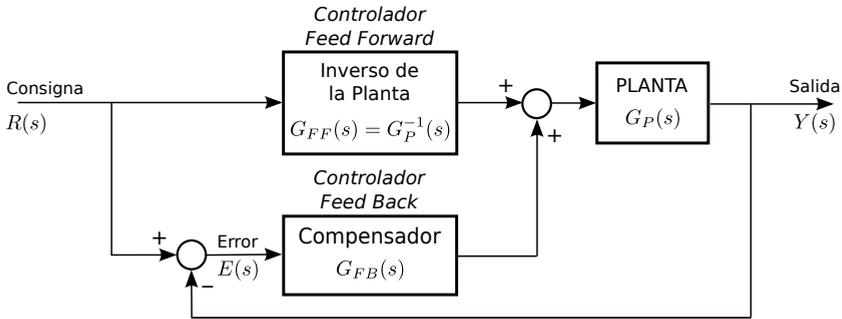


Figura 1.2: Control híbrido *Feed Forward-Feed Back*.

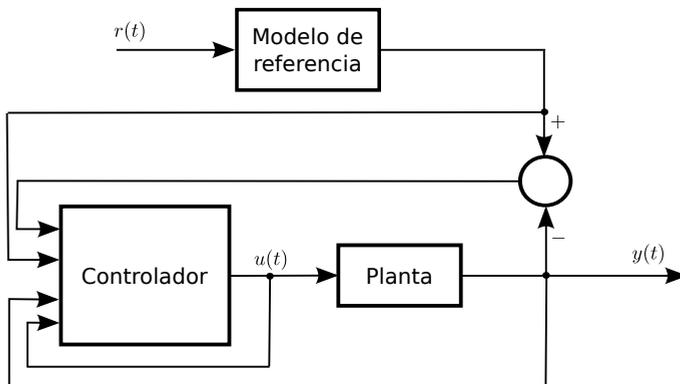


Figura 1.3: Control por modelo de referencia.

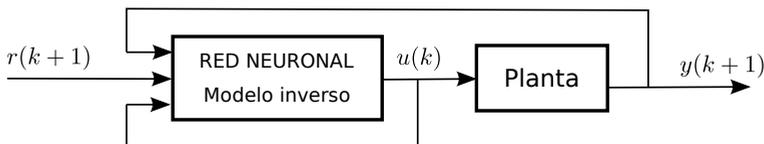


Figura 1.4: Control inverso directo.

Por lo tanto, de la estructura propuesta del sistema de control se puede decir que es una combinación entre un híbrido FF-FB, un sistema de control basado en modelo de referencia y un control inverso directo con redes neuronales. Hay que añadir además, que en el bloque que aparece como *Controlador Inverso*,

hay un sistema de supervisión, que será necesario cuando la planta esté representada por más de un modelo local, de forma que integre las distintas salidas de cada neurona en una única.

Con respecto al controlador FB, será implementado mediante lógica borrosa y por lo tanto será un controlador borroso. Concretamente, para minimizar la complejidad de este controlador se elegirá un PI borroso que será presentado en el capítulo 2, sección 2.2.5.

El *Controlador Inverso* se diseñará en función del modelo de la *Planta* y del *Modelo de referencia*, el cual contiene la información sobre el comportamiento dinámico final deseado del sistema. El *Controlador Borroso* tendrá como función disminuir la señal *Error* en cada instante. Este error se produce principalmente por las incertidumbres en el modelado de la planta y por las perturbaciones que puedan aparecer en el sistema.

Como se verá en las aplicaciones a partir del capítulo 4, con el controlador FF será suficiente para conseguir la respuesta deseada en ciertos casos, sin tener que introducir el control FB, a pesar de que se produzcan errores de modelado y perturbaciones.

Sección 1.4

Metodología de diseño

En la figura 1.5 se puede observar de forma gráfica los distintos pasos y las distintas fases que conformarán el sistema final controlado. Además, se ha incorporado las referencias a los capítulos y/o secciones que tratan el tema en cuestión, tanto en lo relativo a los conceptos básicos como al diseño.

Para aplicar el método propuesto es necesario tener un modelo matemático de la planta a controlar. Para conseguir dicho modelo matemático se puede recurrir al modelado de forma analítica, mediante las ecuaciones que describen el comportamiento de la planta, o a la identificación, aplicando señales de prueba de forma empírica y a partir de los vectores de entrada y salida obtener una descripción del modelo.

Eligiendo la identificación, de la planta física sería interesante y de gran ayuda el conocer el máximo número de características, o sea, conocer su comportamiento dinámico ante diferentes tipos de situaciones y conocer ciertos parámetros, como puede ser el rango de valores máximos de entrada, facilitando el proceso de identificación posterior. Según las señales de entrada que se empleen se obtendrá un modelo aproximado de la planta, que contendrá la información más completa posible sobre su dinámica, a partir del cual se diseñará el controlador.

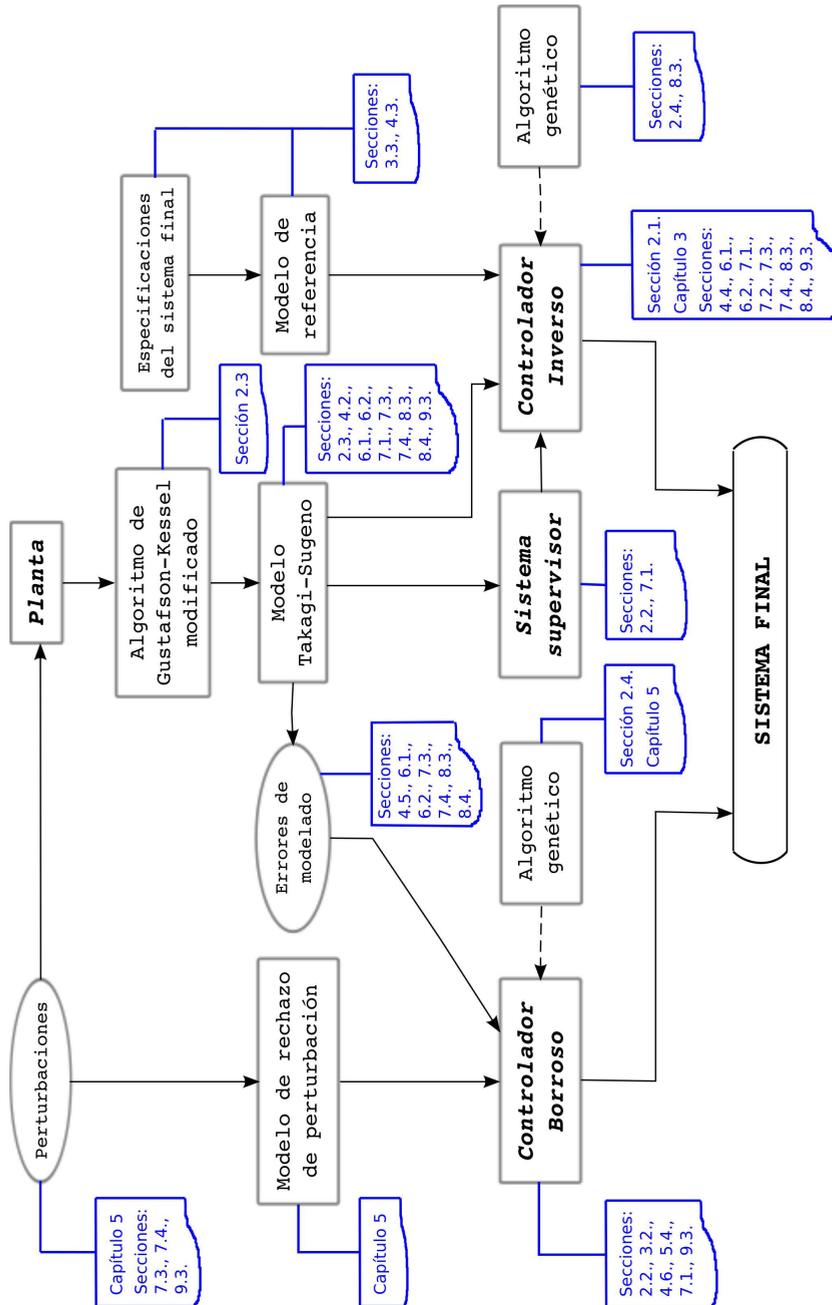


Figura 1.5: Fases y metodología de diseño del sistema de control.

Una vez elegidas las señales de entrada, se miden las salidas obtenidas, constituyendo los vectores de entrada-salida. Con estos vectores se aplica algún algoritmo para obtener un modelo matemático que describa de la forma más exacta posible el comportamiento dinámico de la planta. Para este diseño se ha escogido un algoritmo de identificación por agrupamiento borroso, concretamente el algoritmo de Gustafson-Kessel modificado, dando como resultado un modelo Takagi-Sugeno. Se podrían obtener muchos modelos Takagi-Sugeno que modelarán a la planta real, dependiendo de las señales de entrada usadas y de los parámetros introducidos en el algoritmo de identificación, pero como se verá en las aplicaciones, algunos modelos obtenidos, aunque sean menos exactos, darán mejores resultados en la respuesta final del sistema. Tanto el algoritmo de Gustafson-Kessel modificado como el modelo Takagi-Sugeno, serán explicados en el capítulo 2, en su sección 2.3 que está dedicada al modelado de la planta.

Una vez que se tiene un modelo Takagi-Sugeno, hay que encontrar un inverso aproximado para que el sistema controlado responda a las especificaciones de respuesta impuestas. Con la ayuda de un modelo de referencia, que se diseña a partir de las características en régimen transitorio y permanente del sistema final deseado, se realiza un proceso matemático para obtener dicho inverso aproximado, el cual constituirá al *Controlador Inverso*. Este proceso es importante y de la infinidad de modelos Takagi-Sugeno que se podrían deducir según los parámetros que se inserten en el algoritmo de agrupamiento, se podrían obtener infinidad de modelos inversos aproximados, pero muchos darán lugar a un sistema donde la señal de control se haga inestable, otros no se podrán implementar y del resto algunos no darán los resultados deseados. Por lo tanto, es muy importante establecer las pautas a seguir para formar el inverso aproximado que será implementado mediante una red neuronal.

La idea del control basado en modelo de referencia se expone en el capítulo 3, sección 3.3, y mediante un ejemplo de aplicación se explica como calcularlo en el capítulo 4, sección 4.3. Lo relativo a la teoría de inversión y a la obtención de los inversos aproximados se trata en la sección 3.1 del capítulo 3, pero, en general, en este capítulo se reflejan diversos aspectos del control inverso. El cálculo del controlador inverso, a partir del modelo de referencia, se explica en las diferentes aplicaciones. Por ejemplo, en la sección 4.3, del capítulo 4, se calcula el controlador inverso para la aplicación del control de velocidad de un motor de corriente continua e imán permanente. Tal como se comentó anteriormente, el controlador inverso será implementado mediante una red neuronal. Por ello, en la sección 2.1, del capítulo 2, se trata el campo de las redes neuronales y su aplicación al control.

El modelo Takagi-Sugeno de plantas no lineales consta de varios modelos locales, por lo que al hacer el cálculo del controlador inverso se genera una neurona por cada modelo local, es decir, el controlador inverso constará de varias neuronas con varias salidas. De esta forma, se hace necesario la inclusión de un sistema de supervisión que integre todas las salidas en una. Debido a que los antecedentes del modelo Takagi-Sugeno son borrosos, el sistema de supervisión se ha diseñado mediante lógica borrosa. Lo relacionado con este campo se expone en el capítulo 2, sección 2.2. Más concretamente, en la sección 2.2.3 se explica lo relativo al modelo Takagi-Sugeno. Por su parte, cómo se realiza el diseño de este sistema supervisor se explica mediante un ejemplo en el capítulo 7, en su sección 7.1.5.

El diseño del sistema de control estaría finalizado si no se produjeran perturbaciones en el sistema y si no hubieran errores de modelado. Sin embargo, en un sistema siempre están presentes estos factores adversos, los cuales no se han tenido en cuenta en el diseño del *Controlador Inverso*. Para mitigar los efectos que producirían, se diseña mediante lógica borrosa el *Controlador Borroso* que se incorpora en la cadena de realimentación. Por lo tanto, el sistema final constará de dos controladores: el controlador inverso de prealimentación y el controlador borroso de realimentación. Esta estructura de control se analiza en el capítulo 3, en su sección 3.2.

El diseño del controlador borroso, para compensar los errores de modelado, se muestra mediante un ejemplo de aplicación en el capítulo 4, sección 4.6, en base a los errores de modelado expuestos en la sección anterior 4.5. Este controlador borroso servirá de base para el resto de aplicaciones de los capítulos posteriores. Para el rechazo de las perturbaciones se parte del controlador borroso diseñado para las incertidumbres de modelado y se añade el requisito de que el sistema final debería responder, ante la presencia de una perturbación, lo más parecido posible a un *Modelo de rechazo de perturbación*, el cual se fija en función de la dinámica de la planta. Todo lo relacionado con las perturbaciones y el modelo de rechazo de perturbación se abarca en el capítulo 5, donde además se muestra un ejemplo de diseño para el rechazo de las perturbaciones de carga en el control de la velocidad angular de un motor de corriente continua e imán permanente. Tal como se indicó anteriormente, lo relacionado con el control borroso se plasma en la sección 2.2.

Para optimizar la respuesta del sistema, tanto en el seguimiento de la consigna como en el rechazo de las perturbaciones, se incorpora un algoritmo genético. Dependiendo del tipo de planta a controlar, el algoritmo genético sintonizará de forma más fina algunos parámetros del sistema, pudiendo actuar sobre las ganancias del *Controlador Borroso* y/o los pesos de la red

neuronal del *Controlador Inverso*. Los conceptos vinculados con los algoritmos genéticos son tratados en el capítulo 2, sección 2.4. En cuanto al diseño, en el capítulo 5 se realiza un estudio en profundidad sobre los parámetros que hay que tener en cuenta a la hora de sintonizar el algoritmo genético para que encuentre valores óptimos de las ganancias del controlador borroso. En el capítulo 8, sección 8.3.6, se realiza el estudio cuando el algoritmo genético debe sintonizar los pesos de la red neuronal.

Sección 1.5

Organización

Esta memoria de tesis está organizada como sigue. En este capítulo 1 se ha explicado la motivación que ha promovido el desarrollo de este trabajo, en base al problema objeto de estudio, y los objetivos que se persiguen en esta investigación. Además, se presentó y explicó el esquema de bloques del sistema de control objeto de desarrollo y la metodología de diseño propuesta.

En el siguiente capítulo 2 se exponen los fundamentos de algunos campos de conocimiento que entran a formar parte de la estructura y metodología propuesta. El control con redes neuronales, los controladores borrosos, el modelado de la planta y los algoritmos genéticos son los campos de conocimiento tratados, los cuales serán de gran utilidad y darán soporte metodológico al siguiente capítulo.

En el capítulo 3 se expone el estado del arte directamente relacionado con las propuestas desarrolladas en esta tesis. El control inverso, el control *Feed Forward-Feed Back* y el control basado en modelo de referencia son las técnicas tratadas en este capítulo.

La serie dedicada a aplicaciones comienza en el capítulo 4, enfocándose al control de la velocidad angular de un motor de corriente continua e imán permanente. Se diseñará tanto el controlador inverso como el controlador de realimentación, de forma que queden compensadas las incertidumbres de modelado.

En el capítulo 5 se extiende el diseño del controlador borroso, diseñado en el capítulo anterior, para que, aparte de compensar las incertidumbres de modelado, pueda corregir y minimizar los efectos debidos a perturbaciones de carga. Además, se desarrollará un algoritmo genético que se encargará de sintonizar las ganancias del controlador borroso.

Las plantas lineales son el objetivo del capítulo 6. Se comienza con una planta de segundo orden, diseñando el controlador inverso y discutiendo cómo mejorar el comportamiento del sistema controlado. También se estudia su es-

tabilidad y robustez. Lo mismo se hace para una planta de tercer orden que al discretizarla se convierte en una planta de fase no mínima, resultando más dificultoso su control.

Las no linealidades se empiezan a tratar en el capítulo 7. Se comienza con las no linealidades continuas y suaves. En este caso se trata de un sistema mecánico masa-resorte no lineal-amortiguador. Se diseña el control inverso y se comparan los resultados obtenidos con otras técnicas de control. Además, se incorpora el controlador borroso para los errores de modelado. La segunda aplicación consiste en el control de nivel de líquido en un tanque cónico, tratándose de un proceso que ofrece fuertes no linealidades continuas. Se diseña el control inverso para dos tanques y se comparan los resultados con otras estrategias de control. Además, para uno de los tanques, se diseña un controlador predictivo no lineal para realizar una comparativa adicional. También se estudia la estabilidad y robustez del sistema final.

Una aplicación de sistema con no linealidad discontinua es tratada en el capítulo 8. Se trata de la zona muerta que aparece en muchos procesos. Se aplica el control inverso a dos casos y se comparan las prestaciones con otras técnicas propuestas en la literatura. Para el primer ejemplo se hace uso de un algoritmo genético para sintonizar de forma más fina a la red neuronal. Como en los capítulos anteriores se observa la estabilidad y robustez del sistema.

En el capítulo 9, se trata el caso del control de convertidores reductores conmutados DC/DC —*Direct Current* (DC). Estos circuitos son no lineales y variantes en el tiempo, pero se puede analizar y realizar el diseño a partir de un modelo promedio. Se aplica el control inverso a un ejemplo de convertidor, al que se le han aplicado diferentes estrategias de control, y se diseña además el controlador borroso para el rechazo de las perturbaciones de carga. Como siempre, se hace un estudio de la estabilidad y robustez del sistema resultante.

Para finalizar, un resumen de las conclusiones y líneas futuras cierra la memoria.

Fundamentos

Una vez que en el capítulo anterior se centró al lector sobre la problemática a resolver, en este capítulo se expondrán algunos campos de conocimiento que entran a formar parte de la estructura y metodología propuesta. Debido a que el controlador FF se implementará mediante una red neuronal, se comienza explicando lo que es una red neuronal y de los tipos más comunes, para dar a continuación una visión sobre el control neuronal. Posteriormente, se abarca el campo relativo al control borroso, incluyendo conceptos de lógica borrosa, debido a que el controlador FB se implementará como controlador borroso. Se sigue explicando conceptos del modelado, centrándonos en el modelo Takagi-Sugeno y en la técnica de modelado por agrupamiento borroso, ya que de esta forma es como se obtendrá el modelo de la planta para posteriormente diseñar el controlador FF. El último campo de interés es el de los algoritmos genéticos ya que en este trabajo serán los encargados de entrenar al controlador FF y FB cuando sea necesario.

Los conceptos aquí mostrados son de gran utilidad y dan soporte metodológico al siguiente capítulo, en el que se expondrá el estado del arte.

El capítulo está organizado como sigue. En la siguiente sección 2.1, se aborda el control con redes neuronales. Los controladores borrosos son el objeto de estudio en la sección 2.2. El modelado de la planta se expone en la sección 2.3, incidiendo en la identificación borrosa y explicando el algoritmo de Gustafson-Kessel. Para finalizar, en la sección 2.4, se abordan los algoritmos genéticos, explicando los conceptos relacionados con esta temática y los operadores involucrados que más frecuentemente se utilizan.

Control con redes neuronales

Debido a que el controlador inverso será implementado mediante redes neuronales, en esta sección se hará una breve descripción de lo que son y de los tipos que serán usados en esta tesis, para posteriormente describir dos métodos de control directamente relacionados con el esquema de control final. No se trata de hacer una descripción exhaustiva de las redes neuronales sino de los principios básicos. Existe una amplia bibliografía especializada sobre el tema que se puede consultar [1–5].

2.1.1. Introducción

El control de sistemas no lineales es un área importante de aplicación de las redes neuronales, aunque también se pueden aplicar a sistemas lineales. El problema del control con redes neuronales se puede enfocar de dos formas:

1. Métodos de diseño directo: significa que el controlador es una red neuronal que debe ser entrenada. Esta solución es ventajosa cuando la plataforma disponible en tiempo real impide soluciones complicadas.
2. Métodos de diseño indirecto: en este caso el diseño está basado en un modelo de red neuronal del sistema a ser controlado, de modo que el controlador no es una red neuronal.

El entrenamiento de la red neuronal se puede ver como un problema de optimización no lineal, en el que se intenta encontrar una representación óptima de la red neuronal para minimizar una función objetivo.

Para esta tesis se ha escogido el método de diseño directo, optando por una solución simple y realizando el entrenamiento de la red *offline*.

2.1.2. Modelo de la neurona artificial

Se denomina neurona a un dispositivo simple de cálculo que a partir de un vector de entrada proporciona una única respuesta o salida. Los elementos que constituyen a una neurona i son los siguientes:

- Entradas $x_j(t)$: procedentes de un sensor o de otra neurona.
- Pesos sinápticos w_{ij} : representan la intensidad de interacción entre cada neurona presináptica j y la postsináptica i .

- Regla de propagación: proporciona el valor del potencial postsináptico, $h_i(t)$, de la neurona i en función de sus pesos y entradas. La función más habitual es de tipo lineal, que se basa en la suma ponderada de las entradas con los pesos sinápticos (2.1).

$$h_i(t) = \sum_j w_{ij} \cdot x_j(t) \quad (2.1)$$

- Función de activación: proporciona el estado de activación actual, $a_i(t)$, en función del estado de activación anterior y de su potencial postsináptico actual. Sin embargo, en muchos modelos se considera que el estado actual de la neurona no depende de su estado anterior. Las funciones de activación más empleadas son la identidad, escalón, lineal a tramos, sigmoidea, gaussiana y sinusoidal.
- Función de salida: proporciona la salida actual, $y_i(t)$, en función de su estado de activación. Frecuentemente la función de salida es la identidad, $y_i(t) = a_i(t)$.

Con frecuencia se añade al conjunto de pesos de la neurona un parámetro adicional b_i , que se puede denominar polarización (*bias*) o umbral (*threshold*), y que se resta del potencial postsináptico. Por lo tanto, el modelo de neurona estándar (figura 2.1) queda definido en (2.2).

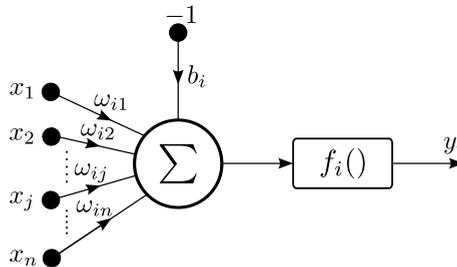


Figura 2.1: Modelo de neurona estándar.

$$y_i(t) = f_i \left(\sum_j w_{ij} \cdot x_j(t) - b_i \right) \quad (2.2)$$

Se denomina arquitectura a la topología de conexionado de una red neuronal. En un sistema neuronal artificial los nodos se conectan por medio de sinapsis,

determinando el comportamiento de la red. En general, las neuronas se suelen agrupar en unidades estructurales que se denominan capas.

Cuando se construye un sistema neuronal, se parte de una determinada arquitectura de red, estableciéndose habitualmente los pesos sinápticos iniciales de forma aleatoria. Para que la red resulte operativa es necesario entrenarla, constituyendo el aprendizaje, el cual se puede llevar a cabo modificando los pesos sinápticos o la arquitectura de la red.

No se hará una descripción de los algoritmos existentes para el entrenamiento de las redes neuronales, ya que no serán aplicados en el trabajo que se presenta en esta tesis.

2.1.3. Tipos de redes neuronales artificiales

Dependiendo del modelo de neurona concreto que se utilice, de la arquitectura de conexión y del algoritmo de aprendizaje, surgirán distintos tipos de redes neuronales. A continuación se presenta la descripción de las redes neuronales que tendrán relación con las usadas en esta tesis para implementar al controlador FF. Para una descripción detallada del resto de redes neuronales puede consultar las referencias [2, 3].

Red neuronal lineal adaptativa

La red neuronal lineal adaptativa —*ADaptive LInear NEuron* (ADALINE)— es un modelo unidireccional, compuesta por una capa sensorial o de entrada, con n neuronas, y otra capa de salida, con m neuronas. La salida se puede expresar como en (2.3).

$$y_i(t) = f\left(\sum_{j=1}^n w_{ij} \cdot x_j(t) + b_i\right), \quad 1 \leq i \leq m \quad (2.3)$$

Las neuronas de entrada no realizan ningún cómputo y la función de activación de las neuronas de la capa de salida es de respuesta lineal, es decir, la función de activación es la identidad: $y_i(t) = h_i(t)$.

Perceptrón multicapa

Un perceptrón multicapa —*Multi-Layer Perceptron* (MLP)— se obtiene al añadir capas intermedias (figura 2.2). La estructura más común es la que tiene una capa oculta, con funciones de activación de tipo sigmoideo y neuronas de salida lineales, constituyendo un aproximador universal de funciones.

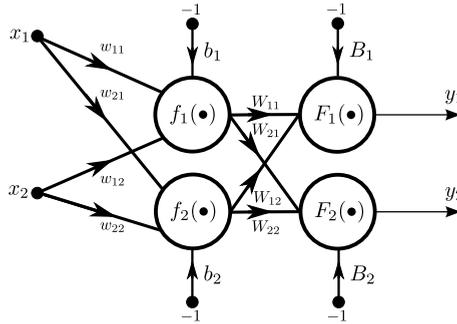


Figura 2.2: Perceptrón multicapa.

Redes de neuronas recurrentes

Las redes de neuronas recurrentes se caracterizan porque pueden aparecer en la red conexiones de una neurona a ella misma, conexiones entre neuronas de una misma capa o conexiones de las neuronas de una capa a la capa anterior.

2.1.4. Control neuronal con modelo de referencia

En este esquema (figura 2.3), el funcionamiento deseado del sistema en lazo cerrado se especifica a través de un modelo de referencia. El sistema de control intenta hacer que la salida de la planta, $y_p(t)$, coincida con la salida del modelo de referencia, $y_m(t)$. El error en esta estructura se utiliza como señal de entrenamiento de la red que actúa como controlador.

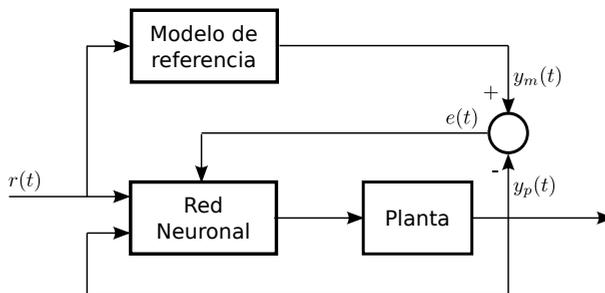


Figura 2.3: Control con modelo de referencia.

2.1.5. Control neuronal inverso directo

Cuando las redes neuronales fueron propuestas originalmente para controlar sistemas no lineales desconocidos, uno de los primeros métodos fue entrenar a

la red neuronal para actuar como el inverso del sistema y usarla como controlador. Aunque en la sección 3.1 se explicará lo relativo al control inverso, aquí se aplica al caso en el que el controlador es una red neuronal. Se asume que la planta puede ser descrita mediante la ecuación (2.4).

$$y(k+1) = g[y(k), \dots, y(k-n+1), u(k), \dots, u(k-m)] \quad (2.4)$$

La función g , que representa al modelo directo de la planta, debe ser invertible y debido a ello se puede obtener el modelo inverso y la red generar la señal de control más reciente $\hat{u}(k)$ según se expresa en (2.5). Si la red representa el inverso exacto, la señal de control generada $\hat{u}(k)$ llevará a la salida del sistema en el instante $(k+1)$ al valor de la consigna en ese instante (figura 2.4).

$$\hat{u}(k) = \hat{g}^{-1}[y(k+1), \dots, y(k-n+1), u(k-1), \dots, u(k-m)] \quad (2.5)$$

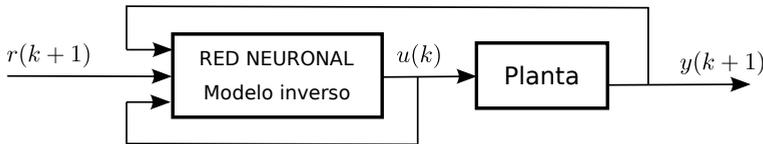


Figura 2.4: Control inverso directo.

Hay dos técnicas para establecer el modelo inverso: método *offline* conocido como entrenamiento general y método *online* llamado entrenamiento especializado. Aquí no se dará una explicación de estos métodos puesto que no serán de aplicación en el trabajo presentado. Para una descripción detallada puede consultar la referencia [4].

Sección 2.2

Controladores borrosos

La lógica borrosa o lógica difusa (*fuzzy logic*) es una rama de la inteligencia artificial que se funda en el concepto de que todo es cuestión de grado, permitiendo describir y gobernar a un sistema por medio de reglas de sentido común. La lógica borrosa se fundamenta en los denominados conjuntos borrosos (*fuzzy sets*), los cuales fueron introducidos por primera vez en 1965 [6]. Teniendo en cuenta que tanto el controlador de realimentación como el modelado de la planta se basarán en la lógica borrosa, en este apartado se tratará esta temática.

2.2.1. Los conjuntos borrosos

Sea X un universo de discurso, del cual A es un subconjunto, es decir: $A \subseteq X$. En teoría clásica de conjuntos cualquier elemento x perteneciente a X pertenece o no pertenece al subconjunto A , sin que exista ninguna otra posibilidad al margen de estas dos. La función de pertenencia va a indicar de forma numérica si un elemento x pertenece o no a un subconjunto A , asignando a cada elemento x del universo de discurso un 0 si no pertenece o un 1 si pertenece. La función de pertenencia se representa de la forma (2.6), siendo $\mu_A(x)$ el grado de pertenencia del elemento x al conjunto A .

$$\mu_A(x) = \left\{ \begin{array}{ll} 0, & \text{si } x \notin A \\ 1, & \text{si } x \in A \end{array} \right\} \quad (2.6)$$

La mayoría de los fenómenos que encontramos cada día son imprecisos, es decir, tienen implícito un cierto grado de borrosidad en la descripción de su naturaleza. Así el conjunto “*Personas altas*” es un conjunto borroso, puesto que los elementos que lo integran no están definidos con exactitud, es decir, existen elementos de los cuales no podemos decir que pertenecen o no pertenecen al conjunto de forma rotunda.

En la lógica borrosa el grado de pertenencia puede tomar todos los valores reales comprendidos en el intervalo $[0, 1]$. Por ejemplo, sea $X = \text{Altura normal de una persona adulta}$ el universo de discurso donde se definen tres conjuntos borrosos: $A = \text{baja}$, $B = \text{mediana}$ y $C = \text{alta}$ (figura 2.5). Según las funciones de pertenencia asignadas, hay regiones en las que se considera baja y mediana a la vez, entre 1.6 m y 1.7 m, o mediana y alta, entre 1.8 m y 1.9 m.

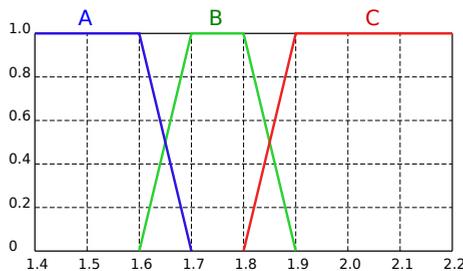


Figura 2.5: Ejemplo de conjuntos borrosos.

La mayoría de las veces, la asignación de las funciones de pertenencia se basa en la experiencia. En general, es preferible usar funciones de pertenencia simples, ya que simplifican muchos cálculos y no se pierde exactitud, debido a que precisamente se está definiendo un concepto borroso.

Los conjuntos borrosos se pueden operar entre sí del mismo modo que los conjuntos clásicos. Si tenemos dos conjuntos borrosos $A \subset X$ y $B \subset Y$, y un par $(x, y) \in X \times Y$, el conectivo *y* (*and*) se puede implementar mediante la relación borrosa $\mu_{AND}(x, y) = \min(\mu_A(x), \mu_B(y))$, o mediante cualquier *t-norma*. El conectivo *o* (*or*) se define habitualmente como la relación $\mu_{OR}(x, y) = \max(\mu_A(x), \mu_B(y))$, o mediante cualquier *s-norma*.

El operador de implicación *entonces* (*then*) puede definirse de muchas maneras en base a *t-normas* y *s-normas*. Dos ejemplos de las más usadas son el mínimo de Mamdani [7] y el producto de Larsen [8].

2.2.2. Estructura de un controlador borroso

El control borroso —*Fuzzy Control* (FC)— es una rama importante de la teoría de conjuntos borrosos, constituyendo una aplicación a la automatización de procesos y pudiéndose considerar como una clase independiente y a su vez como una variedad de controladores no lineales.

Un controlador borroso típicamente infiere los consecuentes de un conjunto más o menos grande de reglas simples, pudiéndose realizar en paralelo este proceso de razonamiento, lo que permite que incluso controladores relativamente complejos puedan realizar la inferencia borrosa en un tiempo de cálculo mínimo. Hay que señalar que la aplicación de la lógica borrosa en el control de procesos no está en absoluto reñida con el empleo de otras técnicas de control convencionales.

La estructura típica de un controlador borroso se muestra en la figura 2.6 y consta de los bloques necesarios para llevar a cabo el razonamiento borroso. A continuación se describe la función de cada bloque.

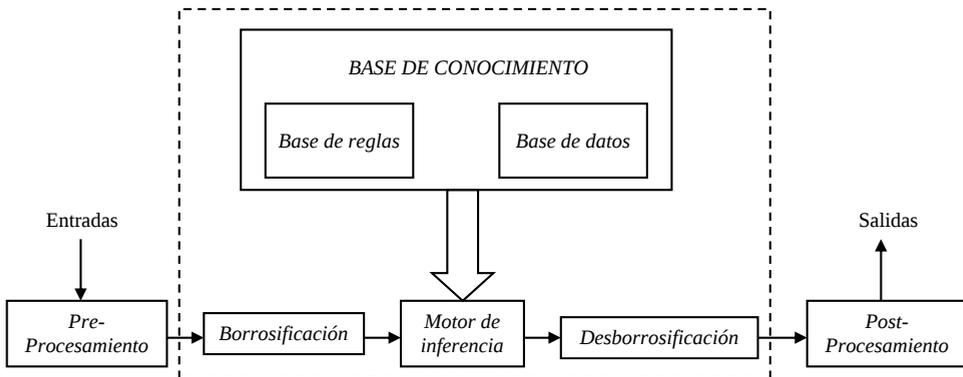


Figura 2.6: Estructura típica de un controlador borroso.

Pre-Procesamiento

Su misión es acondicionar las señales antes de entrar en el sistema borroso. Algunos ejemplos de pre-procesamiento son: muestreo y cuantificación, normalización, derivación y/o integración, sintonía de parámetros y filtrado.

Borrosificación

En la borrosificación (*fuzzyfication*) se recogen los valores reales (*crisp*) entregados por la etapa de pre-procesamiento y se asignan grados de pertenencia a cada entrada y para cada conjunto borroso.

Base de conocimiento

La base de conocimiento está formada por la base de reglas y la base de datos, la cual contiene la información sobre las funciones de pertenencia de las variables de entrada y de salida.

Si un problema puede ser resuelto en términos de razonamiento aproximado, un experto podrá transmitir sus conocimientos mediante un conjunto de reglas de la forma (2.7), siendo X_i las entradas, U_i las salidas, A los conjuntos borrosos de las entradas y B los conjuntos borrosos de las salidas. La base de reglas de la forma (2.7) es la de tipo Mamdani, porque fue él quien las usó en el primer controlador borroso [7].

$$\left. \begin{array}{l} \text{if } (X_1 \text{ is } A_{j,X_1}) \text{ and } \dots \text{ and } (X_n \text{ is } A_{j,X_n}) \\ \text{then } (U_1 \text{ is } B_{j,U_1}) \text{ and } \dots \text{ and } (U_m \text{ is } B_{j,U_m}) \end{array} \right\} \quad (2.7)$$

Una forma alternativa de la base de reglas es la de tipo Tagaki-Sugeno (2.8), en la que los consecuentes de cada regla son funciones arbitrarias, f_{j,U_k} , generalmente lineales, del espacio de entrada.

$$\left. \begin{array}{l} \text{if } (X_1 \text{ is } A_{j,X_1}) \text{ and } \dots \text{ and } (X_n \text{ is } A_{j,X_n}) \\ \text{then } (U_1 \text{ is } f_{j,U_1}(X_1, \dots, X_n)) \text{ and } \dots \\ \dots \text{ and } (U_m \text{ is } f_{j,U_m}(X_1, \dots, X_n)) \end{array} \right\} \quad (2.8)$$

Motor de inferencia

La inferencia da lugar a un conjunto borroso de salida por cada regla, determinándose primero la función de pertenencia de los antecedentes y posteriormente la de los consecuentes. Para la función de pertenencia de los antecedentes, hay que elegir entre las *t-normas* y las *s-normas* disponibles para

el conectivo *and* y *or*, respectivamente. Para el conectivo *and* se suele usar el método del mínimo y el método del producto. Para la función de pertenencia de los consecuentes, hay que aplicar algún método que evalúe el operador de implicación *then*, siendo el mínimo de Mamdani y el producto de Larsen los más usados. Por ejemplo, sea un controlador borroso con dos entradas, X_1 y X_2 , y tres conjuntos borrosos definidos para cada entrada, $\{A_1, A_2, A_3\}$ para X_1 y $\{A_4, A_5, A_6\}$ para X_2 , tal como muestra la figura 2.7. Sea $X_1 = -0.2$ y $X_2 = 0.4$ y la regla: “*if* (X_1 is A_1) *and* (X_2 is A_5) *then* ... “. El grado de pertenencia obtenido para cada entrada es: $\mu_{A_1}(X_1) = 0.2$ y $\mu_{A_5}(X_2) = 0.6$ (figura 2.7).

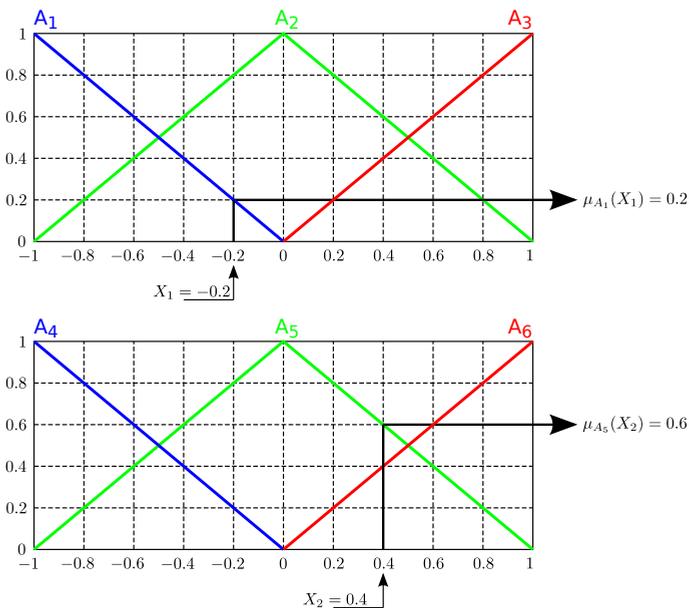


Figura 2.7: Grados de pertenencia en el antecedente.

Por el método del mínimo se obtendría el valor $\alpha = 0.2$ y por el método del producto el valor $\alpha = 0.12$. Si los conjuntos borrosos para la variable de salida son los mostrados en línea discontinua en la figura 2.8 y la regla es “*if* (X_1 is A_1) *and* (X_2 is A_5) *then* U is B_1 ”, por el método de Mamdani se obtendría el conjunto borroso de la figura 2.8 (para $\alpha = 0.2$). El método de Larsen generaría un conjunto borroso producto de α y $\mu_{B_1}(U)$.

Lo normal es que exista más de una regla, por lo que es necesario realizar la operación de agregación de los conjuntos borrosos obtenidos para cada regla. Los métodos de agregación más usados en control son el máximo, la suma y el producto, los cuales realizan dichas operaciones en cada punto para todos los

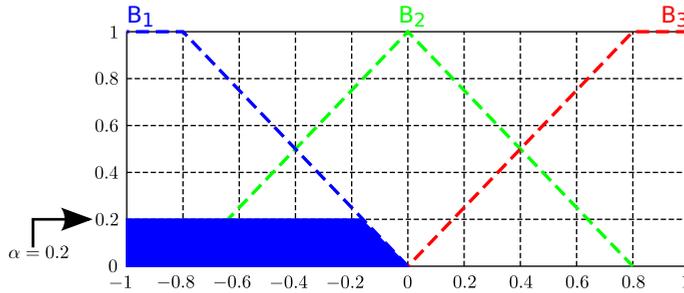


Figura 2.8: Implícación de Mamdani.

conjuntos borrosos. Por ejemplo, si el sistema anterior consta de dos reglas, dadas en (2.9), después de aplicar la agregación por el método del máximo, se obtendría el conjunto borroso mostrado en la figura 2.9.

$$\left. \begin{array}{l} \text{if } (X_1 \text{ is } A_1) \text{ and } (X_2 \text{ is } A_5) \text{ then } (U \text{ is } B_1) \\ \text{if } (X_1 \text{ is } A_2) \text{ and } (X_2 \text{ is } A_6) \text{ then } (U \text{ is } B_2) \end{array} \right\} \quad (2.9)$$

Los métodos de inferencia más conocidos y usados en la práctica, utilizando la base de reglas tipo Mamdani, son:

1. Método de Mamdani (*mínimo-máximo*): se utiliza el método mínimo para la implícación y en la agregación usa el método máximo.
2. Método de Larsen (*producto-máximo* o *producto-suma*): se utiliza el método producto para la implícación y en la agregación usa el método máximo o suma.

Desborrosificación

La desborrosificación (*defuzzification*) convierte el conjunto borroso obtenido a la salida del proceso de inferencia en un valor concreto (*crisp*). Existen diferentes estrategias de desborrosificación, no habiendo ningún procedimiento sistemático para su elección. En la práctica, la más usada es la del centro de gravedad —*Centre Of Gravity* (COG)—, en la que la salida es la abscisa que coincide con el centro de gravedad del conjunto borroso. Este es el más eficiente de todos los métodos, ya que proporciona variaciones suaves y continuas de la señal de control.

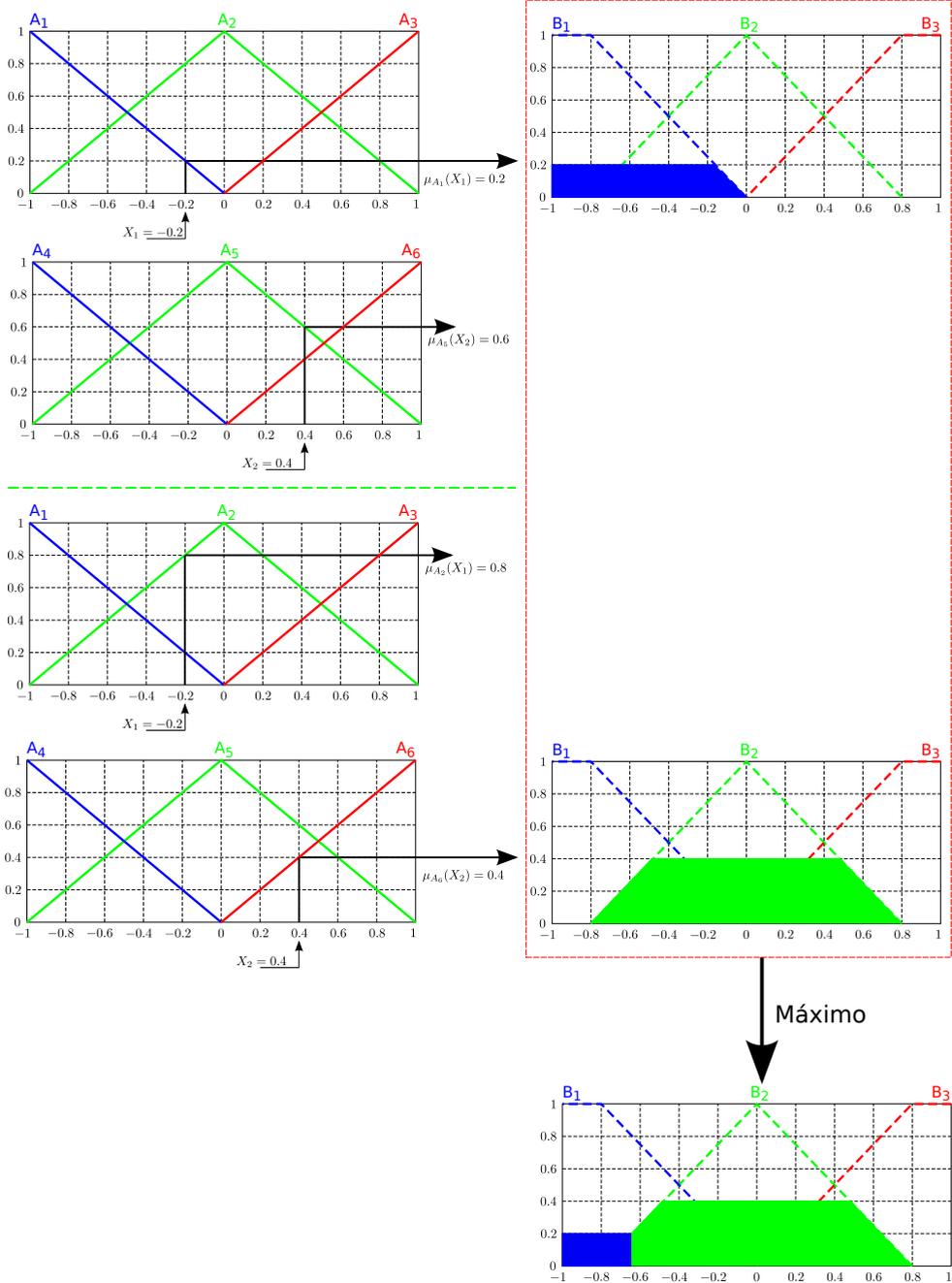


Figura 2.9: Conjunto borroso por el método del máximo.

Post-Procesamiento

Acondiciona las señales de salida del controlador borroso al proceso físico. Algunos ejemplos son: desnormalización o escalado, integración de la señal de salida y sintonía de parámetros.

2.2.3. Proceso en Takagi-Sugeno

Para la base de reglas tipo Takagi-Sugeno, en el proceso de agregación se incluyen los consecuentes de cada regla en el universo de discurso final y en la desborrosificación los métodos más usados son:

1. Método de la media ponderada (*weighted average, wtaver*): el valor de la salida y se calcula según (2.10).

$$y = \frac{\sum \omega_i \cdot y_i}{\sum \omega_i} \quad (2.10)$$

2. Método de la suma ponderada (*weighted sum, wtsum*): la salida y viene dada por (2.11).

$$y = \sum \omega_i \cdot y_i \quad (2.11)$$

En las expresiones (2.10) y (2.11), y_i es el valor de y para la regla i , siendo ω_i el peso de y_i . Por ejemplo, en la figura 2.10 se observa el caso de las tres reglas dadas en (2.12). Aquí el consecuente es $y_i = b_i$, siendo b_i el número de la regla. Usando el método del mínimo para la implicación: $\omega_1 = 0$ (ya que no se activa la regla R_1), $\omega_2 = 0.24$ y $\omega_3 = 0.76$. En la agregación aparecen los consecuentes de las reglas R_2 y R_3 , es decir, $y_2 = 2$ e $y_3 = 3$. En la desborrosificación, con el método *wtaver*, se obtiene el valor final $y = 2.76$, calculado en (2.13).

$$\left. \begin{array}{l} R_1 : \text{If } y(k-1) \text{ is } A_{11} \text{ and } y(k-2) \text{ is } A_{12} \text{ and} \\ \quad \text{and } u(k-1) \text{ is } A_{13} \text{ then } y(k) = 1 \\ R_2 : \text{If } y(k-1) \text{ is } A_{21} \text{ and } y(k-2) \text{ is } A_{22} \text{ and} \\ \quad \text{and } u(k-1) \text{ is } A_{23} \text{ then } y(k) = 2 \\ R_3 : \text{If } y(k-1) \text{ is } A_{31} \text{ and } y(k-2) \text{ is } A_{32} \text{ and} \\ \quad \text{and } u(k-1) \text{ is } A_{33} \text{ then } y(k) = 3 \end{array} \right\} \quad (2.12)$$

$$y = \frac{\omega_2 \cdot y_2 + \omega_3 \cdot y_3}{\omega_2 + \omega_3} = \frac{0.24 \cdot 2 + 0.76 \cdot 3}{0.24 + 0.76} = 2.76 \quad (2.13)$$

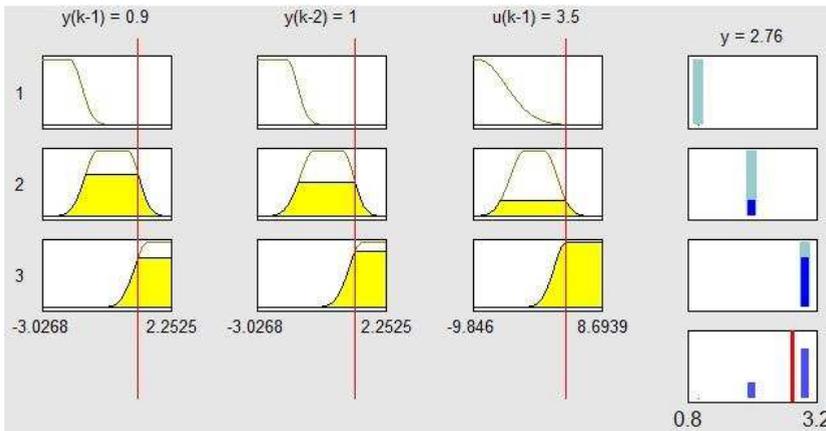


Figura 2.10: Ejemplo con reglas Takagi-Sugeno.

2.2.4. PID borroso

La mayoría de los trabajos de investigación sobre controladores Proporcional Integral Derivativo —*Proportional Integral Derivative* (PID)— borrosos se enfocan hacia los controladores de dos entradas propuestos por Mamdani [7], de forma que las primeras aplicaciones fueron con estructuras PI borrosas. La determinación de la estructura del controlador borroso y la sintonización del sistema es la tarea más dura, pero es el problema fundamental para unas prestaciones óptimas. En la mayoría de los casos, los procedimientos de prueba y error con simulación mediante algún software es el método que aplica el diseñador para la sintonización del controlador borroso.

El primer nivel de sintonización es la determinación de la base de conocimiento del sistema, esto incluye la estructura PID borrosa, la base de reglas, las funciones de pertenencia de los antecedentes y consecuentes y su distribución, el mecanismo de inferencia y la estrategia de desborrosificación. El segundo nivel es la sintonización de los factores de escala y otras ganancias usadas en la construcción de la estructura.

Además de la estructura del tipo PI de Mamdani, se han desarrollado otras estructuras, siendo los controladores PID borrosos prácticos más simples los mostrados en la tabla 2.1, donde e es el error.

2.2.5. PI borroso

Dentro de las estructuras PID borrosas prácticas, este trabajo se centrará en las estructuras que implementan al PI. Con ello se simplificará la base de

TABLA 2.1: ESTRUCTURAS PID BORROSAS PRÁCTICAS

Entradas	Tipo
e	P
$e, de/dt$	PD
e	PI
$e, de/dt$	PI
e	PID
$e, de/dt$	PID
$e, de/dt, d^2e/dt^2$	PID

conocimiento y como consecuencia el hardware necesario.

La respuesta del PI clásico ideal en tiempo continuo es la dada en (2.14), siendo K_P la constante proporcional, K_I la constante integral, $e(t)$ el error y $u(t)$ la señal de control.

$$u(t) = K_P \cdot e(t) + K_I \cdot \int e(t) dt \quad (2.14)$$

En (2.14) las entradas son el error y la integral del error. Sin embargo, la experiencia dicta que es más difícil escribir las reglas para la acción integral, además de los problemas debidos al efecto *windup* del integrador. Por lo tanto, para un controlador PI borroso se propone como solución que las entradas sean el error y su derivada y la salida la derivada de la señal de control, tal como se expresa en (2.15). Para obtener la señal de control basta con realizar la integral a la salida. La figura 2.11 muestra la estructura propuesta para el controlador PI borroso y es la que se usará en esta tesis.

$$\frac{du(t)}{dt} = K_P \cdot \frac{de(t)}{dt} + K_I \cdot e(t) \quad (2.15)$$

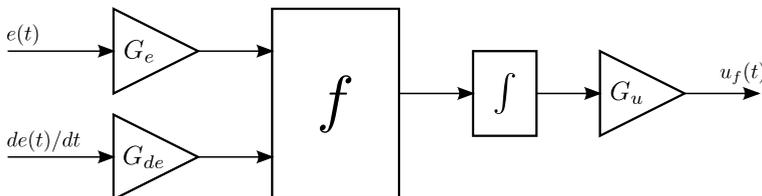


Figura 2.11: Estructura del PI borroso.

La salida del controlador de la figura 2.11 es la dada en (2.16), siendo en general f una función no lineal.

$$u_f(t) = G_u \cdot \int f\left(e(t) \cdot G_e, \frac{de(t)}{dt} \cdot G_{de}\right) \quad (2.16)$$

Sección 2.3

Modelado de la planta

Para el diseño del controlador inverso es necesario disponer de un modelo de la planta. Dicho modelo deberá estar expresado en forma de ecuaciones en diferencias que definirán el comportamiento del proceso. En esta sección se describirá la opción que se ha elegido para conseguirlo.

2.3.1. Introducción

El modelado es un paso previo imprescindible en el diseño de sistemas de control y para obtenerlo existen dos métodos:

1. *Modelado teórico*: realizado de forma analítica, planteando las ecuaciones necesarias. El problema de este enfoque es que se requiere un buen conocimiento del proceso.
2. *Identificación del sistema*: método experimental que permite obtener el modelo a partir de datos reales recogidos de la planta bajo estudio. Este enfoque resulta generalmente el más directo y el que permite obtener el modelo a más corto plazo.

Para tratar de modelar sistemas complejos, se puede recurrir a su descomposición en un conjunto más o menos grande de submodelos que tienen un rango de validez limitado y que se denominan modelos locales. El modelo global del sistema se obtendrá a través de la integración de los modelos locales, utilizando alguna técnica que permita la selección del modelo local en función de la situación en la que se encuentra el sistema. Con este enfoque se dispone de una técnica simple e intuitiva para el modelado de procesos complejos. Uno de los problemas que surgen a la hora de utilizar esta técnica es la elección del número de modelos locales y la identificación de los distintos parámetros de los modelos locales utilizados. Una de las alternativas es la utilización de datos experimentales y utilizar algún método basado en algoritmos genéticos, redes neuronales, plantillas o técnicas de agrupamiento (*clustering*).

Hay sistemas que si se excitan con las mismas variables de entrada en dos instantes diferentes, no producen las mismas respuestas. Aparece, pues, la necesidad de considerar la existencia de sistemas de parámetros variables con el tiempo. Este hecho introduce dos importantes variantes en la identificación del proceso a llevar a cabo:

1. Fuera de línea (*offline*): cuando se tiene la seguridad de que no se van a producir cambios a lo largo del tiempo en los parámetros del modelo considerado y es suficiente un modelo simple que sea capaz de reproducir con fidelidad algunas de las características esenciales del proceso.
2. En línea (*online*): cuando se desea reproducir más fielmente el comportamiento del proceso.

La consideración de una u otra posibilidad determina la necesidad de aplicar unas u otras teorías y algoritmos de identificación. Para una información más detallada puede consultar bibliografía existente sobre este tema, como por ejemplo [9–11].

Esta tesis se centrará únicamente en el control de plantas invariantes en el tiempo o de plantas variantes en el tiempo de las que se pueda identificar un modelo promediado. Por ello, se optará por el método de identificación fuera de línea y la técnica de agrupamiento borroso (*fuzzy clustering*), aplicando el algoritmo de Gustafson-Kessel [12] modificado [13]. La elección del agrupamiento borroso como método de identificación es debido a que los algoritmos de agrupamiento borroso son los más adecuados para la identificación borrosa [14].

2.3.2. Modelo borroso Takagi-Sugeno

Para la aproximación de una gran clase de sistemas no lineales, Takagi y Sugeno introdujeron en 1985 un modelo borroso basado en reglas [15]. La identificación usando modelos borrosos Takagi-Sugeno es una herramienta efectiva para la aproximación de sistemas dinámicos no lineales basada en la información suministrada por los datos de entrada-salida, mediante la interpolación de modelos locales lineales. Este modelo consiste en reglas de la forma (2.17), donde x es la variable de entrada (el antecedente), y_i es la variable de salida (el consecuente), R_i es la regla i -ésima, K es el número de reglas y A_i es el conjunto borroso del antecedente de la regla i -ésima. El antecedente se suele expresar como una combinación de proposiciones de la forma (2.18).

$$R_i : \text{If } x \text{ is } A_i \text{ then } y_i = f_i(x), \quad i = 1, 2, \dots, K \quad (2.17)$$

$$R_i : \left. \begin{array}{l} \text{If } x_1 \text{ is } A_{i1} \text{ and } x_2 \text{ is } A_{i2} \text{ and } \dots \text{ and } x_P \text{ is } A_{iP} \\ \text{then } y_i = f_i(x), \quad i = 1, 2, \dots, K \end{array} \right\} \quad (2.18)$$

Las funciones del consecuente f_i tienen una estructura que permanece igual en todas las reglas y sólo varían los parámetros de esta. Una forma simple y útil en la práctica es la lineal afín, $y_i = a_i^T \cdot x + b_i$, dando lugar a reglas con consecuentes que se pueden expresar de manera sencilla en forma de un modelo lineal de primer orden (2.19), llamado modelo Takagi-Sugeno afín, cuyas propiedades las investigó Rovatti [16].

$$R_i : \text{If } x \text{ is } A_i \text{ then } y_i = a_i^T \cdot x + b_i, \quad i = 1, 2, \dots, K \quad (2.19)$$

Cuando $a_i = 0$, los consecuentes en (2.19) son funciones constantes, obteniéndose el modelo *Singleton* (2.20).

$$R_i : \text{If } x \text{ is } A_i \text{ then } y_i = b_i, \quad i = 1, 2, \dots, K \quad (2.20)$$

2.3.3. Agrupamiento borroso: Gustafson-Kessel

El agrupamiento borroso (*fuzzy clustering*) es una herramienta que se puede usar para obtener de forma simultánea las funciones de pertenencia del antecedente con los modelos lineales locales del consecuente.

Los datos son organizados en una matriz Z por concatenación de una matriz que contiene los regresores y un vector que contiene la información de los regresandos. Por ejemplo, un sistema de la forma (2.21), al que se le hicieron N medidas, es organizado en la matriz Z (2.22), donde los regresores estarían constituidos por las cuatro primeras filas y el regresando por la fila última. El vector en la k -ésima columna de la matriz Z será denotado como z_k y contiene la información completa del sistema en el instante k .

$$y(k) = F(y(k-1), y(k-2), u(k-1), u(k-2)) \quad (2.21)$$

$$Z = \begin{bmatrix} y(2) & y(3) & \cdots & y(N-1) \\ y(1) & y(2) & \cdots & y(N-2) \\ u(2) & u(3) & \cdots & u(N-1) \\ u(1) & u(2) & \cdots & u(N-2) \\ y(3) & y(4) & \cdots & y(N) \end{bmatrix} \quad (2.22)$$

La mayoría de los algoritmos de agrupamiento borroso están basados en la optimización de la función objetivo *c-means* (2.23).

$$J(Z; U; V) = \sum_{i=1}^c \sum_{k=1}^N (\mu_{ik})^m \|z_k - v_i\|_A^2 \quad (2.23)$$

En (2.23):

- Datos que deben ser clasificados: $Z = \{z_1, z_2, \dots, z_N\}$.
- Matriz de partición borrosa de Z : $U = [\mu_{ik}]$.
- Vector de centroides a determinar: $V = [v_1, v_2, \dots, v_c]$.
- Grado de pertenencia de cada dato z_k al centro del prototipo de *clúster*¹ v_i : μ_{ik} .
- Norma², de la distancia del producto interno³ al cuadrado, determinada por la elección de la matriz A (2.24).

$$D_{ikA}^2 = \|z_k - v_i\|_A^2 = (z_k - v_i)^T \cdot A \cdot (z_k - v_i) \quad (2.24)$$

- Exponente que determina la borrosidad de las clases resultantes (cuanto mayor es m , más borrosos son los conjuntos identificados): $m \in (1, \infty)$.

El valor de la función de coste (2.23) es una medida ponderada del error cuadrático que se comete al representar las c clases por los prototipos v_i . La minimización de (2.23) es un problema de optimización no lineal que puede ser resuelto de muchas formas, pero la más habitual es la conocida como algoritmo *Fuzzy C-means*. El problema más importante de este algoritmo es que las clases identificadas tienen forma hiper-elipsoidal, cuando lo deseable para una posterior aplicación a control es que dichas clases tengan una estructura afín o lineal. Existen muchas extensiones y modificaciones al algoritmo *Fuzzy C-means* que pueden ser clasificados en tres grandes grupos:

- Algoritmos que utilizan una medida de la distancia adaptativa, es decir, una norma diferente para cada clase. Esto posibilita la detección de clases de datos con estructuras, tamaños y formas, diferentes.
- Algoritmos basados en prototipos lineales, con la norma constante y prototipos variables.
- Algoritmos basados en prototipos no lineales. Este tipo de algoritmos no tienen aplicación al caso de identificación de modelos locales lineales.

¹El término *clúster* es la castellanización del término en inglés *cluster*, el cual significa en su traducción literal al castellano: racimo, conjunto, grupo o cúmulo. En este documento se usará *clúster*.

²El operador norma determina la longitud de un vector.

³También conocido como producto escalar o producto punto.

El algoritmo de Gustafson-Kessel es un algoritmo que emplea una norma de distancia adaptativa y se suele usar para la obtención de modelos borrosos, detectando *clústers* de diferentes contornos geométricos en un conjunto de datos, dado que las clases hiper-elipsoidales que busca detectan de forma bastante correcta los comportamientos cuasi-lineales de los diversos regímenes de funcionamiento. El algoritmo de Gustafson-Kessel extiende el algoritmo básico *Fuzzy C-means* eligiendo una norma diferente A_i para cada clase (2.25) en lugar de la A constante (2.24).

$$D_{ik}^2_{A_i} = \|z_k - v_i\|_{A_i}^2 = (z_k - v_i)^T \cdot A_i \cdot (z_k - v_i) \quad (2.25)$$

Para obtener una solución viable, A_i debe ser limitada de alguna forma, siendo la más habitual fijar su determinante, lo que es equivalente a optimizar la forma de la clase manteniendo su volumen constante (2.26), con ρ_i constante para cada clase.

$$|A_i| = \rho_i, \quad \rho > 0 \quad (2.26)$$

Si F_i (2.27) es la matriz de covarianzas⁴ de la clase i , la expresión que se obtiene con los multiplicadores de Lagrange para A_i está dada en (2.28).

$$F_i = \frac{\sum_{k=1}^N (\mu_{ik})^m (z_k - v_i)(z_k - v_i)^T}{\sum_{k=1}^N (\mu_{ik})^m} \quad (2.27)$$

$$|A_i| = [\rho_i \cdot \det(F_i)]^{1/n} \cdot F_i^{-1} \quad (2.28)$$

La sustitución de las ecuaciones (2.27) y (2.28) en (2.25) da una norma de distancia Mahalanobis [17] entre z_k y v_i , donde la covarianza es ponderada por los grados de pertenencia en U . Para la elección de m se aplica lo mismo que en el caso del algoritmo *Fuzzy C-means*. Sin conocimiento previo, los volúmenes de los *clústers* ρ_i se fijan en el valor de uno para cada *clúster*. Una desventaja del algoritmo Gustafson-Kessel es que debido a la restricción de (2.26), sólo puede encontrar *clústers* de volúmenes aproximadamente iguales, siendo el *clúster* formado un hiperelipsoide.

Aunque el algoritmo de Gustafson-Kessel detecta los comportamientos cuasi-lineales bastante bien, presenta problemas numéricos debido a que la matriz F_i puede llegar a ser singular, y por lo tanto no se puede invertir, de modo que la matriz A_i (2.28) no puede ser calculada. Esto se produce cuando el número de muestras de datos es pequeño o cuando los datos dentro de un *clúster* están

⁴Matriz que contiene la covarianza, relación o dependencia entre dos variables, entre los elementos de un vector.

correlacionados linealmente. Babuška [13, 18] propuso una solución a este problema aplicando una técnica para calcular F_i . Sin embargo, como resultado se puede obtener un modelo borroso Takagi-Sugeno pobre debido a que cuando el número de datos en un *clúster* es demasiado bajo, los *clústers* tienen poca relación con la distribución real de los datos. Para resolver estos problemas, Babuška [13, 18] propuso un algoritmo de Gustafson-Kessel modificado y el paquete de herramientas (*toolbox*) obtenido es el de modelado e identificación borrosa —*Fuzzy Modeling and IDentification* (FMID)— [19] para MATLAB®.

2.3.4. Índice VAF

El índice VAF —*Variance Accounted For* (VAF)— es a menudo utilizado para evaluar la calidad de un modelo identificado, mediante la comparación de la salida verdadera del sistema real con la salida de dicho modelo. El VAF, en porcentaje, entre dos señales y_1 e y_2 se calcularía según se indica en (2.29), donde *var* es la varianza. De esta forma, dos señales iguales tendrían un VAF igual al 100 %, siendo inferior si las señales difieren.

$$VAF = \left[1 - \frac{\text{var}(y_1 - y_2)}{\text{var}(y_1)} \right] \cdot 100 \% \quad (2.29)$$

Sección 2.4

El algoritmo genético

El algoritmo genético —*Genetic Algorithm* (GA)— es un algoritmo de búsqueda y optimización cuyo mecanismo de funcionamiento está basado en el proceso de evolución natural, es decir, en la resolución de problemas inspirada en la naturaleza y por lo tanto, parten del principio darwiniano de reproducción y supervivencia de los individuos más aptos.

En esta tesis se hará uso del algoritmo genético para sintonizar ciertos parámetros del controlador borroso y en algunos casos para sintonizar los pesos de la red neuronal que constituye el controlador inverso. Por ello, en este apartado se introducirán los conceptos relativos al algoritmo genético y su principio de operación.

2.4.1. Conceptos básicos del algoritmo genético

Los conceptos básicos que se manejan en la teoría del GA son:

- *Individuo o cromosoma*: un individuo determina una potencial solución del problema que se pretende resolver mediante el GA.
- *Gen, locus y alelo*: un cromosoma está compuesto por un conjunto de genes, siendo el *locus* la posición del gen en el cromosoma y el *alelo* cada uno de los valores que puede tomar un gen.
- *Genotipo y fenotipo*: el *genotipo* es la descripción genética del individuo, mientras que el *fenotipo* es la forma en que se expresa el genotipo.
- *Población*: conjunto de individuos con los que se trabaja en el GA.

En líneas generales, la estrategia operativa de un GA consiste en partir de una población inicial que se evalúa y se obtiene el grado de adaptación de cada individuo al entorno. A partir de esta situación inicial se realizan una serie de iteraciones en cada una de las cuales se simula la creación de una nueva generación de individuos. Este proceso consiste en aplicar los operadores genéticos de selección, cruce y mutación sobre los individuos, hasta llegar finalmente a una población que, si el algoritmo converge adecuadamente, estará compuesta por buenos individuos, siendo el mejor de éstos la solución que ofrece el algoritmo. Un proceso de los llamados elitistas siempre mantendrá en la población al mejor individuo alcanzado hasta el momento. La gran utilidad que tienen los GA como estrategias de resolución de problemas se basa en los excelentes resultados experimentales que producen.

2.4.2. Representación, tamaño y población inicial

Lo primero que se debe hacer es elegir el tipo de representación que se va a usar para el cromosoma. En lugar de considerar la representación binaria, que se ve severamente limitada en varios dominios, se puede tener como cromosoma un vector de números reales. Cualquier cambio en la representación lleva asociado un cambio en los operadores genéticos y debe permitir que éstos produzcan individuos válidos y que sean fáciles de evaluar. Para un conocimiento más detallado sobre esta temática se puede consultar [20, 21].

El tamaño de la población es un parámetro importante. Si el tamaño de la población es muy grande se realizará una exploración del espacio de búsqueda más rápida en términos de generaciones, sin embargo, el tiempo necesario para obtener una nueva generación será mayor, dado que se estarán procesando individuos innecesarios. Si el tamaño de la población es muy pequeño podría llegar a darse el caso de que la población converja rápidamente a una solución que no sea lo suficientemente buena. Existen estudios sobre la influencia del tamaño de la población como el realizado en [22].

En cuanto a la población inicial, es la principal fuente de material genético para el algoritmo y se debe generar de forma que contenga cromosomas que estén bien dispersos por el espacio de soluciones. Si no se dispone de algún tipo de información, la manera más simple de cumplir con este objetivo es elegir cromosomas al azar. Sin embargo, los agrupamientos generados son de muy baja calidad y lleva demasiadas generaciones al GA llegar a buenas soluciones.

2.4.3. Funciones de evaluación y aptitud

El uso de los conceptos de función de evaluación y función de aptitud se suelen utilizar como sinónimos. Sin embargo, ambos conceptos son diferentes, y es conveniente hacer una distinción entre ellos. La función de evaluación, o función objetivo f_i , indica que tan buena es la solución obtenida tras la aplicación de un conjunto de parámetros, independientemente de la aplicación de esta función sobre otro conjunto de parámetros, siendo la función a minimizar o maximizar. Para un sistema de control la función objetivo puede ser un índice de funcionamiento que es una medida cuantitativa del funcionamiento de un sistema. Lo interesante de los índices de funcionamiento es que se puedan aplicar con facilidad, destacando los siguientes:

1. Integral del cuadrado del error —*Integral Square Error* (ISE)—:

$$f_{ISE} = \int_0^T e^2(t) dt \quad (2.30)$$

2. Integral de la magnitud absoluta del error —*Integral of Absolute Error* (IAE)—:

$$f_{IAE} = \int_0^T |e(t)| dt \quad (2.31)$$

También está la integral del tiempo multiplicada por el error al cuadrado —*Integral of Time multiplied by the Squared Error* (ITSE)—, que es una variante del ISE, y la integral del tiempo multiplicada por el error absoluto —*Integral of Time multiplied by Absolute Error* (ITAE)—, que es una variante del IAE.

Por su parte, la función de aptitud (*fitness function*) F_i , siempre está definida con respecto al resto de los individuos de la población y nos indica que tan buena o mala es una solución comparada con el resto de las soluciones obtenidas hasta el momento, transformando el valor de la función objetivo en una medida de aptitud relativa (2.32), es decir, la función aptitud nos indica la probabilidad de supervivencia de un individuo.

$$F_i = g(f_i) \quad (2.32)$$

La evaluación de esta función no debe tener un coste excesivo, pero, por otra parte, la función aptitud debe discriminar bien la calidad de los individuos. Al igual que existen diferentes funciones de evaluación, también existen diferentes funciones de aptitud. Una función de aptitud usada normalmente en el algoritmo genético es la mostrada en (2.33).

$$F_i = \frac{f_i}{f} = \frac{f_i}{\frac{\sum f_i}{N}} = \frac{N \cdot f_i}{\sum f_i} \quad (2.33)$$

2.4.4. Operadores genéticos

Selección

La selección es el primer paso, y una parte fundamental del funcionamiento del GA, donde se empieza a manipular el material genético de la población inicial para realizar el proceso de selección de candidatos a reproducirse. Este proceso de selección suele realizarse de forma probabilística, es decir, aún los individuos menos aptos tienen una cierta probabilidad de sobrevivir, de forma que los individuos que tengan un mayor valor de la función de aptitud tendrán mayor probabilidad de ser seleccionados.

Existen diferentes métodos de selección, siendo los más usuales los métodos de selección proporcionales, selección por jerarquías y selección por torneo. Los métodos de selección proporcionales describen a un grupo de esquemas de selección, originalmente propuesto por Holland [23], en los cuales se eligen individuos de acuerdo a su contribución de aptitud con respecto al total de la población. La selección por jerarquías, propuesta por Baker [24], evita la convergencia prematura de las técnicas de selección proporcional. La selección mediante torneo, propuesta por Wetzel [25], es similar a la de jerarquías, pero es computacionalmente más adecuada para implementarse en paralelo. Para una descripción detallada de los métodos de selección se pueden consultar las referencias [26–28].

Cruce

Tras realizar la selección se realiza el cruce que consiste en la sustitución de un conjunto de genes de un padre por los genes correspondientes del otro padre para generar un nuevo hijo. La forma de realizarlo depende de la representación elegida para los cromosomas.

La forma más sencilla de hacer el cruce entre dos individuos es el denominado cruce en un punto. Esta técnica fue propuesta por Holland [23] y fue muy popular durante muchos años. Hoy en día, sin embargo, no suele usarse mucho en la práctica debido a sus inconvenientes. De Jong [29] fue el primero en implementar un cruce de n puntos, como una generalización del cruce de un punto. El valor $n = 2$ es el que minimiza los efectos disruptivos del cruce y de ahí que sea usado con gran frecuencia.

Mutación

La mutación es un proceso aleatorio donde un alelo de un gen es reemplazado por otro, actuando para recuperar el material genético que podría perderse a través de la acción de selección y cruce [30]. Cuando la representación no es binaria, la mutación puede ser llevada a cabo seleccionando de forma aleatoria el nuevo valor dentro de un rango permitido. Wright [31] y Janikow y Michalewicz [32] demuestran cómo con codificación real se puede tener ventaja de las tasas de mutación más altas que con codificación binaria. De hecho, Tate y Smith [33] sostienen que para codificaciones más complejas que la binaria, las tasas altas de mutación pueden ser deseables y necesarias.

Reinserción

Tras la selección, cruce y mutación de la población de los padres se genera la población de los hijos cuya aptitud debe ser determinada. Los hijos deben ser reinsertados en la población de los padres para producir una nueva población, pudiéndose utilizar alguno de los siguientes esquemas:

- *Reinserción pura*: se reemplaza la población completa por la nueva.
- *Reinserción uniforme*: se genera una nueva población reemplazando de forma aleatoria a los padres por la descendencia.
- *Reinserción basada en la aptitud*: la nueva población se genera reemplazando a los padres menos competentes por la descendencia más apta.
- *Reinserción por torneo invertido*: se utiliza en combinación con el método de selección por torneo.

Estado del arte

Una vez que en el capítulo anterior se expusieron los fundamentos relativos a la temática involucrada en esta tesis, en este capítulo se expone el estado del arte directamente relacionado con las propuestas desarrolladas. Se comienza con el control inverso, explicando el concepto de inversión y la realizabilidad de dicho inverso. Se continúa exponiendo diferentes técnicas y algoritmos para obtener un inverso aproximado de sistemas de fase mínima y no mínima. A continuación se trata el control *Feed Forward-Feed Back* mostrando su estructura y dando una visión sobre el estado del arte de publicaciones actuales sobre este tema. Como tercer campo se aborda el control basado en modelo de referencia, explicando su significado y estructuras.

En el siguiente y sucesivos capítulos se comienza la serie de aplicaciones, llevando lo expuesto hasta ahora al control de plantas con diferentes características dinámicas.

El capítulo está organizado como sigue. La sección 3.1 se dedica al control inverso. Se continúa en la sección 3.2 con el control *Feed Forward-Feed Back*. Finalmente, el control basado en modelo de referencia es el que se trata en la sección 3.3.

Sección 3.1

El control inverso

En el problema del diseño de los sistemas de control, está implícito el concepto de inversión. Podría decirse que todos los controladores generan un inverso

tan exacto como sea posible del proceso, de manera que los controladores van a diferir unos de otros según la técnica usada para generar este inverso aproximado.

El control basado en modelo inverso ha sido ampliamente recogido en la literatura, donde se ha demostrado sus ventajas a la hora de optimizar ciertos parámetros del sistema controlado, mejorando su rendimiento. En las referencias [34–54] se incluyen numerosos ejemplos ilustrativos de aplicación de la teoría del inverso.

3.1.1. El concepto de inversión

El concepto de inversión surge cuando se quiere que la salida de una planta siga a una señal de referencia que se impone al sistema. La correspondencia entre la entrada $u(t)$ y la salida $y(t)$ de un proceso se puede describir mediante la relación $y(t) = g(u(t))$, $t \in \mathfrak{R}$ —relación de pertenencia a los números reales—, siendo $g(\circ)$ un operador generalizado y por lo tanto podría ser no lineal y variante en el tiempo. Según la figura 3.1, si se requiere que la salida $y(t)$ siga a una señal de referencia arbitraria y acotada $r(t)$, siendo $g(\circ)$ estable, la ley de control debe ser de la forma dada en (3.1).

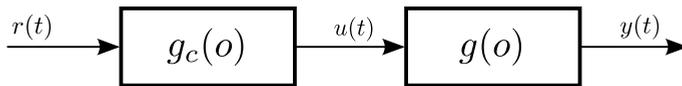


Figura 3.1: Problema fundamental de inversión.

$$g_c(\circ) = g^{-1}(\circ) \quad (3.1)$$

Según (3.1), el controlador $g_c(\circ)$ debe realizar la operación inversa de la planta o proceso $g(\circ)$, y esta ecuación sintetiza la idea básica de inversión. Sin embargo, existen razones que hacen físicamente irrealizable este inverso ideal y, por lo tanto, el controlador implementa un inverso aproximado del proceso.

3.1.2. Realizabilidad del inverso

Los factores que hacen que el controlador $g_c(\circ)$ de la ecuación (3.1) sea o no físicamente realizable son:

- **Estabilidad.** En la figura 3.1 se requiere que $g^{-1}(\circ)$ sea estable para que la señal de control $u(t)$ esté acotada.

- **Causalidad.** El controlador dado en (3.1) debe ser físicamente realizable y para ello debe ser causal. Por ejemplo, en las plantas con retardo de transporte el inverso es físicamente irrealizable.
- **Actuación limitada.** Relativo tanto a la amplitud como a la velocidad máxima de variación de la señal $u(t)$ en los actuadores reales, imposibilitando la obtención de un inverso perfecto.
- **Incertidumbre.** Debido a los errores de modelado de la planta, el inverso ideal de un modelo no implica un inverso perfecto de la planta real.

Por ejemplo, considere una planta $G(s)$ y un controlador $G_c(s)$, lineales e invariantes en el tiempo —*Linear Time Invariant (LTI)*—, modelados según (3.2).

$$\left. \begin{aligned} Y(s) &= G(s) \cdot U(s) \\ U(s) &= G_c(s) \cdot R(s) \end{aligned} \right\} \quad (3.2)$$

La planta se puede expresar de forma genérica tal y como se define en (3.3), donde:

- $n \geq m$: como mínimo debe haber el mismo número de polos (n) que de ceros (m).
- $t_d \geq 0$: el tiempo de retardo de transporte es positivo o nulo.
- $\Re\{p_l\} < 0$: la parte real de los polos debe ser negativa para que la planta sea estable.

$$G(s) = K \cdot \frac{\prod_{i=1}^m (s - c_i)}{\prod_{l=1}^n (s - p_l)} \cdot e^{-t_d \cdot s} \quad (3.3)$$

La inversión ideal se producirá cuando la función de transferencia del controlador sea igual a la inversa de la función de transferencia de la planta: $G_c(s) = G^{-1}(s)$. Con esta elección, para que la salida de la planta sea igual a la referencia impuesta al sistema, se tiene que cumplir que:

- $G(s)$ sea una descripción perfecta de la planta. Esto no es posible ya que siempre existirá incertidumbre en dicha descripción.
- $G_c(s)$ sea estable y realizable. Será estable si la parte real de los ceros de $G(s)$ es negativa ($\Re\{c_i\} < 0$) y será causal, y por lo tanto realizable, si el tiempo de retardo es nulo ($t_d = 0$) y el número de polos es igual al de ceros ($m = n$).

Por lo tanto, no se podrá conseguir la inversión ideal ya que:

- En los sistemas reales un cambio abrupto en la entrada nunca genera un cambio abrupto en la salida, lo que implica que el número de ceros es inferior al número de polos ($m < n$).
- En todo sistema real hay un retardo, aunque sea pequeño.
- Algunos sistemas tienen ceros fuera de la región de estabilidad.

Aparte de todo lo anterior, una planta real es siempre no lineal, siendo la descripción lineal una aproximación en una región de operación. Por ejemplo, la saturación en la entrada de una planta es una no linealidad que siempre aparece, limitando el rango de la señal de entrada. Por lo tanto, un objetivo más realista es encontrar un inverso aproximado que sea estable, realizable, robusto y genere señales de control con límites aceptables.

Según lo discutido hasta aquí, el inverso perfecto no puede ser logrado y es necesario generar un inverso aproximado, existiendo algunas técnicas generales para encontrarlo. A continuación se describirán algunas de estas técnicas, basándose en el hecho de que la planta es estable, lo cual es cierto para la mayoría de los procesos industriales. Si el sistema de interés es inestable, se puede realimentar para estabilizarlo. También se considerará únicamente el caso de sistemas de una entrada y una salida —*Single Input Single Output (SISO)*—.

3.1.3. Inversión de sistemas de fase mínima

Si la planta es de fase mínima¹, una solución para obtener un inverso aproximado puede consistir en agregar un filtro, $F(s)$, con la función de transferencia dada en (3.4), a la inversa teórica de la planta [55].

$$F(s) = \frac{1}{(\lambda \cdot s + 1)^n} \quad (3.4)$$

En (3.4), λ y n se eligen de forma que $G_c(s)$ resulte en una función de transferencia propia. Por ejemplo, si se supone la planta dada en (3.5), el inverso sería de la forma (3.6), que no es implementable físicamente. Si se añade a $G^{-1}(s)$ un filtro $F(s)$, con $n = 1$, quedaría el controlador expresado en (3.7), que sí es realizable físicamente y será estable si el polo ($s = -1/\lambda$) se sitúa en el semiplano izquierdo del plano S .

$$G(s) = \frac{5}{(8 \cdot s + 1)} \quad (3.5)$$

¹Un sistema es de fase mínima si todos los ceros se sitúan en el semiplano izquierdo del plano S o dentro del círculo unidad en el plano Z .

$$G^{-1}(s) = \frac{(8 \cdot s + 1)}{5} \quad (3.6)$$

$$G_c(s) = \frac{(8 \cdot s + 1)}{5} \cdot \frac{1}{(\lambda \cdot s + 1)} \quad (3.7)$$

Si la planta viene especificada de manera discreta, se pueden añadir retardos puros para construir un controlador causal, de forma que la salida siga a la entrada con el número de retardos que se hayan incorporado. Por ejemplo, si se discretiza la planta (3.5), con periodo de muestreo $T = 0.1$ s, se obtiene la planta discretizada (3.8), y el controlador podría quedar como el dado en (3.9). En este caso se ha añadido un retardo (z^{-1}), por lo que la salida seguirá a la entrada con un retardo de 0.1 s.

$$G(z) = \frac{0.0621}{z - 0.9876} \quad (3.8)$$

$$G_c(z) = \frac{z - 0.9876}{0.0621} \cdot \frac{1}{z} \quad (3.9)$$

3.1.4. Inversión de sistemas de fase no mínima

Para los sistemas de fase no mínima se complica la inversión de la dinámica, incluso para sistemas SISO LTI, puesto que los ceros de fase no mínima del sistema original pasan a ser polos inestables en el sistema inverso. Por ejemplo, en el sistema discreto (3.10), los ceros están en $z = 0$ y $z = -2$, constituyendo un sistema de fase no mínima, y su inversa sería inestable, puesto que contendría un polo fuera del círculo unidad.

$$G(z) = \frac{z \cdot (z + 2)}{z^2 - z + 0.75} \quad (3.10)$$

En el sistema LTI (3.11), $B_m(z)$ es el polinomio del numerador que contiene a los ceros de fase mínima y $B_n(z)$ el polinomio que contiene a los ceros de fase no mínima.

$$G(z) = \frac{B(z)}{A(z)} = \frac{B_m(z) \cdot B_n(z)}{A(z)} \quad (3.11)$$

En la literatura se han propuesto diversos esquemas para construir el inverso aproximado [42], que se engloban en dos grupos:

- Inversión exacta: usa de forma directa el inverso exacto aunque sea inestable.
- Inversión aproximada: usa una aproximación estable del inverso exacto inestable.

El inverso exacto causal del sistema (3.11) es el dado en (3.12).

$$\tilde{G}^{-1}(z) = \frac{A(z)}{z^q \cdot B_m(z) \cdot B_n(z)} \quad (3.12)$$

Debido a que el número de raíces del polinomio $A(z)$ no siempre es igual al número de raíces del producto del polinomio $B_m(z)$ por el polinomio $B_n(z)$, se requieren q unidades de retardo para asegurar una implementación causal de $\tilde{G}^{-1}(z)$. Usando (3.12), como el inverso exacto, producirá que la salida $Y(z)$ (figura 3.1) sea igual a la referencia deseada $R(z)$ con un retardo de q muestras (3.13).

$$Y(z) = \frac{1}{z^q} \cdot R(z) = \frac{1}{z^q} \cdot \frac{N(z)}{D(z)} \quad (3.13)$$

El sistema descrito en (3.13) es internamente inestable cuando $G(z)$ es de fase no mínima, produciendo que la señal $u(t)$ (figura 3.1) crezca sin acotar. En algunas aplicaciones se puede producir una señal $u(t)$ útil bajo un intervalo de tiempo reducido, como consecuencia de que la señal $r(t)$ sea de duración muy breve o que los ceros de fase no mínima estén muy cercanos al círculo unidad.

Cuando se necesita una señal $u(t)$ acotada, para cualquier intervalo, algunos algoritmos de inversión exacta usan las respuestas de la condición inicial de $G(z)$ y $\tilde{G}^{-1}(z)$, lo que se denomina precarga de la condición inicial [56,57]. De esta estrategia surgen dos métodos básicos: transferencia del estado de $G(z)$ a un estado inicial en un número de muestras determinadas y el método de preactuación infinita. Ambos métodos y las referencias asociadas se analizan detalladamente en [42].

3.1.5. Algoritmos de inversión aproximada para sistemas de fase no mínima

El inverso del sistema (3.11) se puede expresar como en (3.14), donde el término $\tilde{B}_n^{-1}(z)$ denota que el inverso del polinomio que contiene a los ceros de fase no mínima no es un inverso exacto y, por lo tanto, el inverso del sistema $\tilde{G}^{-1}(z)$ tampoco es exacto. Además, se añaden q muestras de retardo para mantener la causalidad.

$$\tilde{G}^{-1}(z) = \frac{A(z) \cdot \tilde{B}_n^{-1}(z)}{z^q \cdot B_m(z)} \quad (3.14)$$

Los algoritmos de inversión de fase no mínima, con inversión aproximada, se diferencian en la forma en la que obtienen $\tilde{B}_n^{-1}(z)$ y las q muestras de retardo.

Mientras que hay diferentes técnicas de inversión que son estables, las que son más comunes encontrar en la literatura y que mejores prestaciones ofrecen son:

- Controlador de seguimiento con error de fase cero —*Zero-Phase-Error Tracking Controller* (ZPETC)— [58].
- Controlador de seguimiento con error de magnitud cero —*Zero-Magnitude-Error Tracking Controller* (ZMETC)— [59].
- Aproximación en series de Taylor no causal [60].
- Ignorar a los ceros de fase no mínima —*Nonminimum-Phase Zeros Ignore* (NPZI)— [41].

El polinomio $B_n(z)$ se puede expresar como sigue:

$$B_n(z) = b_{nn} \cdot z^n + b_{n(n-1)} \cdot z^{n-1} + \dots + b_{n1} \cdot z + b_{n0} \quad (3.15)$$

En la técnica ZPETC se retiene la dinámica de los ceros de fase no mínima, convirtiendo el polinomio inestable $B_n(z)$ en uno estable (3.16).

$$\left. \begin{aligned} \tilde{B}_n^{-1}(z) &= \frac{B_n^f(z)}{(B_n(1))^2} = \\ &= \frac{b_{n0} \cdot z^n + b_{n1} \cdot z^{n-1} + \dots + b_{n(n-1)} \cdot z + b_{nn}}{(B_n(1))^2} \end{aligned} \right\} \quad (3.16)$$

Para una implementación causal de $\tilde{G}^{-1}(z)$, el orden del retardo q debe ser igual a: $O(A(z) \cdot B_n^f(z)) - O(B_m(z))$.

Por otro lado, la técnica ZMETC transforma los ceros de fase no mínima en polos estables de la inversa aproximada (3.17), siendo el orden de q : $O(A(z)) - O(B_m(z) \cdot B_n^f(z))$.

$$\tilde{B}_n^{-1}(z) = \frac{1}{b_{n0} \cdot z^n + b_{n1} \cdot z^{n-1} + \dots + b_{n(n-1)} \cdot z + b_{nn}} \quad (3.17)$$

El método de la aproximación en series de Taylor no causal busca aproximar $\tilde{B}_n^{-1}(z)$ según la ecuación (3.18), siendo m_T el orden de la aproximación de la serie y la secuencia α_i se deduce de la serie de Taylor.

$$\tilde{B}_n^{-1}(z) = \frac{\sum_{i=0}^{m_T} \alpha_i \cdot z^i}{B_n(1) \cdot \sum_{i=0}^{m_T} \alpha_i} \quad (3.18)$$

La aproximación estable más simple es la técnica NPZI, en la que se ignoran los ceros de fase no mínima en el modelo del sistema y se hace el ajuste adecuado en la ganancia. Es la aproximación menos precisa, pero la más simple cuando hay limitaciones en los recursos a la hora de implementar el controlador. $\tilde{B}_n(z)$ se reduce a $B_n(1)$ y, para que $\tilde{G}^{-1}(z)$ sea causal, el retardo q debe ser igual al orden de $A(z)$ menos el orden de $B_m(z)$: $O(A(z)) - O(B_m(z))$. Esta aproximación es la equivalente a la aproximación en series de Taylor de orden cero.

Sección 3.2

El control Feed Forward-Feed Back

En el enfoque basado en la inversión el controlador es un inverso del modelo de la planta. Con el fin de mejorar las prestaciones que puede ofrecer este tipo de control, en este apartado se añade realimentación, de forma que el sistema de control estará constituido por dos controladores.

3.2.1. Introducción

A un sistema constituido por un único controlador inverso se le denomina de alimentación hacia adelante —*Feed Forward* (FF)—, o de prealimentación, y se usa en muchas aplicaciones para seguir las transiciones de la consigna (*setpoint*) o para realizar el seguimiento de trayectorias deseadas (figura 3.2), como por ejemplo en las tareas de control punto a punto en robótica. Con el esquema de control FF no se tiene control cuando hay errores de modelado de la planta o cuando se producen perturbaciones en el sistema. Para estos casos es necesario recurrir a un esquema de control con dos grados de libertad, que combina la prealimentación con la realimentación —*Feed Back* (FB)—.

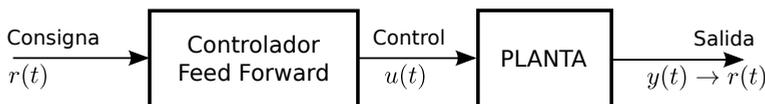
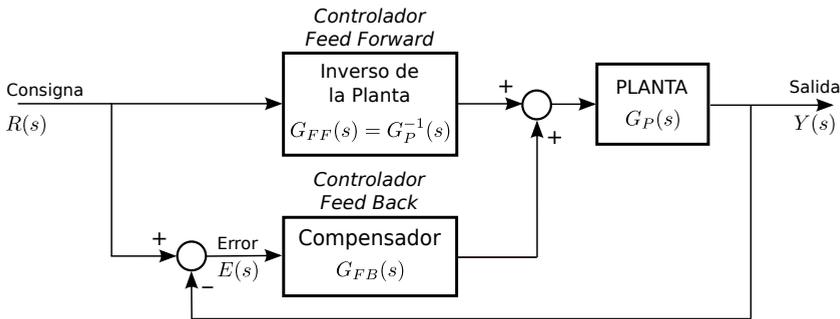


Figura 3.2: Control *Feed Forward*.

En la figura 3.3 se muestra la estructura de un sistema de control híbrido que combina a un controlador típico con estructura FB, $G_{FB}(s)$, y a un controlador con estructura FF, $G_{FF}(s)$. El controlador FF se encargará de seguir a la referencia $r(t)$ y el FB de compensar los errores de modelado de la planta y las perturbaciones que aparezcan en el sistema.

Figura 3.3: Control híbrido *Feed Forward-Feed Back*.

En esta tesis, la implementación del inverso para el controlador FF se realizará mediante redes neuronales (*neural networks*) que fueron tratadas en la sección 2.1. Por otra parte, para el controlador FB se ha optado por un control borroso —*Fuzzy Control* (FC)—, que se describió en la sección 2.2.

Otra arquitectura posible es aquella en la que el controlador FF implementa el inverso de un sistema en lazo cerrado en el que se ha incluido al controlador FB. Esta estructura no será objeto de estudio, ya que no será de aplicación en el trabajo presentado en esta tesis.

También se han propuesto estructuras FF-FB, en la que la parte FF tiene el cometido de cancelar los efectos de las perturbaciones en el sistema y de los errores de modelado de la planta, mientras que la parte FB tiene el cometido tradicional de control de la planta para seguir a la consigna. Este tipo de estructuras tampoco se tratarán en este trabajo. Como referencias iniciales sobre este tipo de estructuras se pueden consultar [61–63]. También se pueden consultar trabajos de investigación y desarrollo relacionados con esta temática como por ejemplo [64–78].

3.2.2. Aplicaciones

Dadas las ventajas que ofrece el control FF, se ha aplicado y se sigue aplicando con regularidad en muchas áreas. A continuación se exponen referencias recientes de investigación y aplicación de este tipo de control.

Los sistemas de fase no mínima han sido objeto de investigación y prueba de diferentes técnicas de control. En [79] se diseña un FF de múltiples entradas y múltiples salidas —*Multi Input Multi Output* (MIMO)—, y de respuesta al impulso finita —*Finite Impulse Response* (FIR)—, para sistemas que tengan ceros de fase no mínima en la planta. Los ceros de la planta se usan para conformar las señales de referencia, mientras que los polos de la planta se

usan para diseñar el controlador FF. Otro ejemplo de control de sistemas de fase no mínima se encuentra en [49] donde para el control de un generador diésel se diseña un controlador FF basado en el modelo inverso del sistema realimentado con un control predictivo basado en modelo.

El control de posición es un área muy importante y se han desarrollado y se siguen investigando técnicas para la mejora de este control. Para controlar la posición de un servo sistema de fase no mínima, en [80] se usa un control ZPETC FF. La misma técnica ZPETC se aplica en [81] para el control de posicionamiento de una mesa XY con un controlador FF adaptativo, incluyéndose además un modelo de referencia en el sistema. También para el control de posicionamiento de una mesa XY, en [82], se comparan tres técnicas FF ZPETC, extendiendo el algoritmo básico ZPETC. En [83] también se trata el control de una mesa XY mediante ZPETC con FF adaptativo. Para el control de posicionamiento con exactitud de una máquina de ensamblado en [84] se diseña un esquema de control FF-FB de velocidad y aceleración. Para sistemas de nanoposicionamiento MIMO en [85] se propone un esquema de dos grados de libertad usando una arquitectura combinada FF-FB y en [86] para una herramienta de litografía de obleas. En [87] se plantea la regulación de la posición de una válvula de recirculación de los gases de escape —*Exhaust Gas Recirculation* (EGR)—. Para ello se propone un control híbrido compuesto de un elemento reset de primer orden —*First Order Reset Element* (FORE)— [88] junto con una acción FF adaptativa. En este caso el objetivo de la acción FF es proporcionar rechazo asintótico de las perturbaciones actuando en la entrada de la planta. Otro ejemplo de aplicación del control FF, en este caso para el control de una válvula electromagnética, se encuentra en [89].

Para el control de seguimiento de acimut de un radar espía, los autores de [90] diseñan un sistema de control FF-FB. Para un manipulador móvil en [91] se usa un controlador robusto para eliminar cualquier efecto de perturbaciones presentes en el sistema y un FF para mejorar las prestaciones de seguimiento de la trayectoria. En algunos campos es necesario calcular la diferencial de una señal como, por ejemplo, en el control de movimiento de alto rendimiento. En [92] se propone implementar el diferenciador de seguimiento no lineal mediante un sistema FF mejorando la estimación de esa diferencial. Para sistemas lineales de parámetros variantes se aplica el control FF en [93] y también en [94] para el problema de seguimiento en un motor síncrono de imán permanente.

Los motores paso a paso también son objeto de control en [95]. Para ello se usa un controlador proporcional integral derivativo —*Proportional Integral Derivative* (PID)— como control FB en combinación con un filtro paso bajo

FF para garantizar la corriente deseada en un motor paso a paso de imán permanente. En el caso del control de motores de corriente continua —*Direct Current* (DC)— sin escobillas de pequeño tamaño, y para reducir el consumo de potencia, los autores de [96] proponen usar un sistema de control de corriente FF. Otros ejemplos de aplicación en motores los encontramos en [97], para el control de motores síncronos de imán permanente interior, o para el control numérico en [98], o en [99] para la compensación del par de carga en un motor síncrono de imán permanente.

Para el control preciso de un piezo actuador de dos grados de libertad —*Degree Of Freedom* (DOF)—, y dado que presenta una fuerte histéresis y acoplamiento entre los dos ejes, los autores de [100] presentan un esquema FF-FB. La parte FF se encarga de compensar la histéresis y la parte FB, con estructura de control por modelo interno, para mejorar las prestaciones del piezo actuador. De la misma forma, en el control de un *piezocantilever* —actuador piezoeléctrico construido por una viga empotrada— de 2-DOF, dedicado a tareas de microensamblamiento y micromanipulación, en [101] se usa un controlador FF, para compensar la no linealidad y la vibración del transductor. En [102] se presenta un esquema de control FF-FB para actuadores piezoeléctricos en aplicaciones de escaneado con trayectoria triangular de alta velocidad. Sobre actuadores piezoeléctricos también se usa en [103] un controlador FF para eliminar el comportamiento no lineal debido a la histéresis y para mejorar aún más la compensación se usa un PID como controlador FB. Otro ejemplo de control aplicado a un tubo piezoeléctrico se encuentra en [104]. El control de un actuador de polímero con histéresis se realiza en [105] implementando un controlador FF como el inverso del modelo del actuador.

Otra área de aplicación es el control de temperatura. Por ejemplo, en [106] se usa un control FF adaptativo para compensar las variaciones de la temperatura ambiente en las zonas de un recinto debidas, por ejemplo, a los cambios de ocupación. Para el control de temperatura del vapor principal en un sistema térmico, en [107] se presenta una nueva estrategia de control FF combinando un control PID convencional con una máquina de soporte vectorial. Otro ejemplo de aplicación de control de temperatura se encuentra en [108].

El campo de la robótica también ha sido objeto de estudio para analizar las ventajas y desventajas de diferentes técnicas de control. Para aumentar el confort en la interacción entre un humano y un exoesqueleto robótico, en [109] se diseña un controlador de impedancia con una compensación FF basada en modelo. En [110], y basado en las estructuras del sistema nervioso y los principios de funcionamiento, un FF, un FB y un mecanismo de adaptación se han integrado y probado en un brazo robótico antropométrico. En la generación

de trayectorias para robots humanoides se hace uso de la combinación FF-FB en [111]. En [112] se optimiza el diseño de un controlador FF para minimizar la sensibilidad para inexactitudes en el modelado de un brazo robótico en el plano horizontal. Usando esquemas FF adaptativos, en [113] se realiza el control de insectos micro robots para el vuelo vertical y de mantenimiento.

Un controlador FF-FB se diseña en [114] para motores de carga homogénea y encendido por compresión —*Homogeneous Charge Compression Ignition* (HCCI)— [115].

En electrónica de potencia también se ha empleado el control FF. Para el control de un inversor trifásico con transformador se propone un control FF en [116]. Concretamente, para el control de la corriente de salida el controlador FF elimina las influencias de las perturbaciones de carga. Las características distintivas de la propuesta son el error nulo en régimen permanente y la rápida respuesta dinámica. El control de convertidores es otro campo donde se han aplicado y se siguen aplicando estrategias basadas en el control FF. Por ejemplo, en [117] se diseña un controlador FF para un convertidor AC/DC de forma que proporcione una mejor corriente de línea y una mejor regulación de la tensión de salida. En lo que respecta a los convertidores para la carga de baterías también se ha aplicado el control FF, como en [118]. Más ejemplos de aplicación a convertidores se pueden encontrar en [119–122]. También para convertidores DC/DC se ha aplicado en [123, 124]. Otros ejemplos de investigación en la aplicación del control FF en electrónica de potencia para inversores se pueden encontrar en [125–127] o en rectificadores con factor de potencia elevado en [128].

El control de nivel de líquido es muy importante en múltiples aplicaciones. En [129] se diseña un controlador PID FF para el control de nivel de agua en una caldera.

El control de vibración activa —*Active Vibration Control* (AVC)— también ha sido objetivo del control FF. Un ejemplo se encuentra en [130], donde se diseña un compensador FF basado en modelo formulado como el diseño de un controlador óptimo H_∞ de orden reducido. Para el problema de rechazo de las vibraciones en una suspensión activa, en [131] se propone el diseño de un controlador FF y FB óptimos.

El ahorro energético es un tema importante hoy en día. Para aumentar la eficiencia energética en un sistema de iluminación mediante LED, en [132] se propone el diseño de una red neuronal FF como controlador. El control de campos de colectores solares es otro tema de interés. Relativo a este tema en [133] se plantea el diseño de un controlador proporcional integral —*Proportional Integral* (PI)— más un FF para el control de un campo de colectores solares

distribuido. En turbinas de viento es aplicado en [134, 135].

La ayuda a personas discapacitadas o con enfermedades como el Parkinson se puede lograr mediante el control de un *joystick* que controla una silla de ruedas. Para ello, en [136] se propone el entrenamiento de una red neuronal que hará las veces del inverso del modelo del *joystick*.

Otras aplicaciones peculiares en las que se han incorporado el control FF las podemos encontrar para el bamboleo de un contenedor en [137], para el seguimiento de la trayectoria de un vehículo aéreo no tripulado basado en helicóptero en [138], para un modelo de un codo músculo-esquelético en [139], para vehículos totalmente eléctricos en [140], para un transmisor por láser en [141], para el control de enfoque y seguimiento en un sistema con disco óptico en [142], para sistemas subestructurados dinámicamente en [143], para un canal de distribución de agua en [144], para sistemas de almacenamiento de energía para ferrocarriles eléctricos en [145], para el control de deslizamiento de las ruedas en [146], para un sistema de levitación magnética de una bola en [147], para un cuadricoptero en [148].

Como el lector ha podido comprobar, el control FF-FB ha sido un campo de publicaciones y de aplicaciones de gran amplitud, y que en la actualidad continúa generando conocimiento.

— Sección 3.3 —

Control basado en modelo de referencia

A veces no se quiere seguir exactamente la señal de entrada, sino una versión más relajada de ésta. Para ello se añade un modelo de referencia, de modo que la señal de salida de la planta debe asemejarse a la señal de salida de dicho modelo. La idea fue debida a Whitaker [149].

3.3.1. Estructura por modelo de referencia

En el control de una planta por modelo de referencia (figura 3.4), la salida del controlador puede depender de la salida de la planta, de la salida del modelo de referencia que se desee seguir, del error entre la salida de la planta y el modelo de referencia y de valores anteriores de la propia señal de control.

En el error cometido a la salida, que es la diferencia entre la salida del modelo de referencia y la salida de la planta, influye el error cometido en la obtención del modelo de la planta, las variaciones de la salida de la planta debidas a las perturbaciones a las que esté sujeta y el error en la obtención del modelo que representará al controlador. El modelo de referencia, que debe

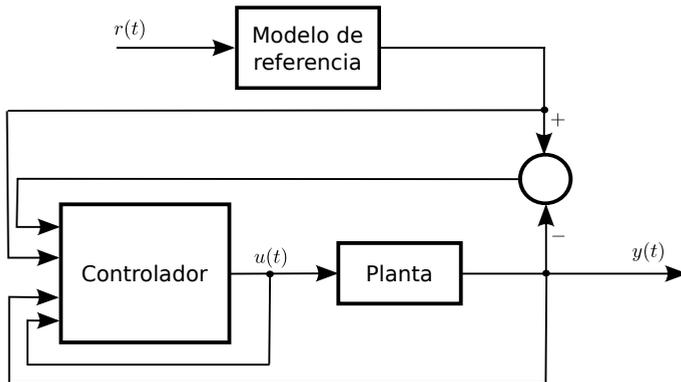


Figura 3.4: Control por modelo de referencia.

tener una dinámica exponencialmente estable, se elige de forma que tenga la misma respuesta dinámica que al diseñador le gustaría que tuviera el sistema controlado. Para ello es necesario tener el suficiente conocimiento de la planta, de forma que se pueda definir el comportamiento deseado por medio de dicho modelo.

3.3.2. Control inverso adaptativo con modelo de referencia

En el control inverso adaptativo, un algoritmo minimiza el error producido por la diferencia entre la consigna y la salida de la planta, ajustando los parámetros del controlador (figura 3.5).

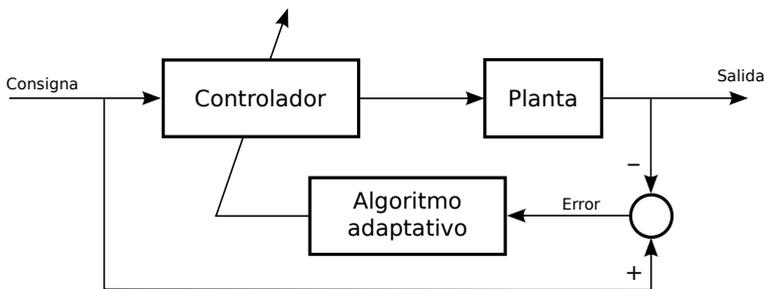


Figura 3.5: Control inverso adaptativo.

El control inverso adaptativo con modelo de referencia, básicamente tiene la misma estructura que la de la figura 3.5, con la diferencia de que el error, que procesa el algoritmo adaptativo, es la diferencia entre la salida de un modelo de referencia y la salida de la planta (figura 3.6).

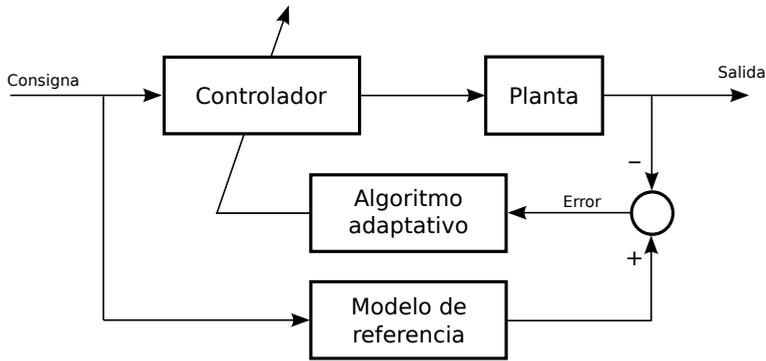


Figura 3.6: Control inverso adaptativo con modelo de referencia.

Se han desarrollado varios esquemas para la cancelación de las perturbaciones que se producen en el sistema, mostrándose en la figura 3.7 la estructura básica.

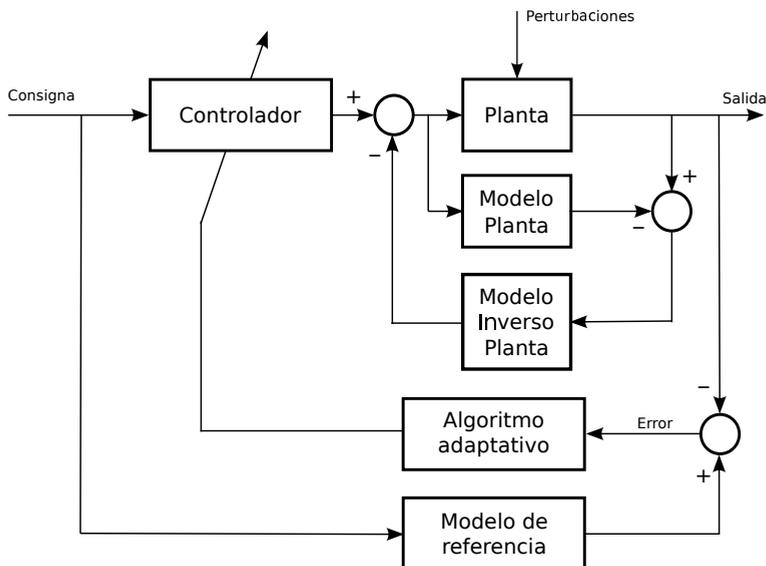


Figura 3.7: Control para la cancelación del ruido.

La entrada de la planta conduce a ésta y a su modelo, siendo la diferencia entre la salida de la planta y la del modelo la que sirve como entrada al modelo inverso de la planta, de forma que se pueda eliminar el efecto de la perturbación.

Control de un motor de corriente continua

Una vez que en los capítulos anteriores se asentaron las bases sobre la arquitectura y metodología de diseño del sistema, exponiendo los fundamentos y estado del arte, con este capítulo se empieza la serie dedicada a aplicaciones. Aquí, concretamente, se enfoca al control de la velocidad angular de un motor de corriente continua e imán permanente que, aunque sencillo, es usado en multitud de aplicaciones. Se diseñará tanto el controlador inverso como el controlador de realimentación, de forma que queden compensadas las incertidumbres de modelado. Para realizar el diseño del controlador inverso, primero se realiza la identificación de la planta, obteniéndose un modelo borroso Takagi-Sugeno y, posteriormente, se establece el modelo de referencia con el cual se fijan las especificaciones en régimen permanente y transitorio del sistema final deseado. Una vez diseñado el controlador inverso, se procede a analizar el comportamiento del sistema y observar su estabilidad y robustez frente a las incertidumbres en el modelado de la planta. También se examinará con qué parámetros se pueden disminuir las influencias adversas que producen estas incertidumbres. Para anular el error en régimen permanente y hacer al sistema más robusto frente a las incertidumbres de modelado, se incorpora y se diseña un controlador borroso. Concretamente se desarrollará un PI borroso y se discutirán los parámetros más adecuados para este controlador.

En el próximo capítulo se extenderá el diseño del controlador PI borroso de forma que además de compensar los errores de modelado, el mismo controlador sea capaz de mitigar los efectos debidos a las perturbaciones de carga que se producen en el sistema.

El capítulo está organizado como sigue. En la sección 4.1, se describe una

planta lineal invariante en el tiempo de primer orden, consistente en un motor de corriente continua e imán permanente con una carga acoplada al eje. Aplicando un algoritmo de agrupamiento, se procede a la identificación de la planta en la sección 4.2. La forma de calcular el modelo de referencia se presenta en 4.3. En la sección 4.4 se procede al diseño del controlador inverso y el comportamiento del sistema es ilustrado en 4.5, junto con un análisis de la estabilidad y robustez ante los errores de modelado de la planta. El diseño del controlador borroso se realiza en la sección 4.6. Por último, en la sección 4.7, se describen las aportaciones de este capítulo.

Sección 4.1

El motor DC PM

La respuesta dinámica de muchos sistemas se puede representar mediante la ecuación diferencial de primer orden (4.1), donde $u(t)$ es la entrada al sistema, $y(t)$ la salida y $\{a_0, a_1, b\}$ son constantes.

$$a_1 \cdot \frac{dy(t)}{dt} + a_0 \cdot y(t) = b \cdot u(t) \quad (4.1)$$

Realizando la transformada de Laplace de (4.1), y despejando la relación entre la salida $Y(s)$ y la entrada $U(s)$, se obtiene la función de transferencia de un sistema lineal invariante en el tiempo LTI de primer orden (4.2), siendo K la ganancia y τ la constante de tiempo.

$$\frac{Y(s)}{U(s)} = \frac{b}{a_1 \cdot s + a_0} = \frac{b/a_1}{(a_1/a_0) \cdot s + 1} = \frac{K}{\tau \cdot s + 1} \quad (4.2)$$

Uno de los actuadores más comunes en la industria es el motor. Entre la amplia variedad de motores, el motor de corriente continua o corriente directa —*Direct Current* (DC)— y de imán permanente —*Permanent Magnet* (PM)— es simple de modelar mediante ecuaciones analíticas o usando algún método de identificación.

La función de transferencia (4.2) puede representar un sistema compuesto por un motor de DC PM con una carga acoplada al eje del motor, ya que este tipo de plantas se puede modelar como un sistema de un simple polo [150,151]. La entrada $u(t)$ representaría la tensión que se aplica a la armadura del motor y la salida $y(t)$ la velocidad angular del eje del motor. Por ejemplo, dado un sistema con una ganancia $K = 10$ y una constante de tiempo $\tau = 0.4481$ s, la función de transferencia será la presentada en (4.3). Además, se establece que

el motor tendrá una tensión máxima de entrada de $\pm 12 V$, siendo ésta una característica importante que se debe conocer.

$$G_p(s) = \frac{Y(z)}{U(z)} = \frac{10}{0.4481 \cdot s + 1} \quad (4.3)$$

Discretizando la función de transferencia (4.3), con un periodo de muestreo $T = 0.1 s$, se obtiene la planta equivalente discreta (4.4). Si se realiza la transformada inversa Z de $Y(z)$, se obtiene la ecuación en diferencias (4.5) que describe a la salida de la planta $y(k)$.

$$G_p(z) = \frac{Y(z)}{U(z)} = \frac{2}{z - 0.8} \quad (4.4)$$

$$y(k) = 0.8 \cdot y(k - 1) + 2 \cdot u(k - 1) \quad (4.5)$$

Sección 4.2

Identificación de la planta

Para el diseño del controlador inverso, el primer paso es obtener un modelo Takagi-Sugeno de la planta y para ello se aplicará un algoritmo de agrupamiento borroso. Aunque el objetivo principal de la identificación borrosa es la aplicación a plantas no lineales, para mostrar un ejemplo de los pasos a seguir, se identificará la planta LTI de primer orden descrita en (4.3).

La identificación y posterior validación de la planta se hace *offline* y consta de dos pasos:

1. Obtención de los vectores de entrada y salida en *Simulink*[®].
2. Identificación y validación: se usará el paquete de herramientas FMID. Mediante la sintonización correcta de los parámetros del algoritmo, se intenta conseguir un modelo con el índice de prestaciones VAF (sección 2.3.4) lo más elevado posible.

Para esta planta es suficiente con aplicar señales en forma de escalón (figura 4.1), siendo el período de muestreo elegido de $T = 0.1 s$. La mitad de las muestras se emplearán en el proceso de identificación y la otra mitad en el proceso de validación, creándose los vectores de entrada y de salida que los utilizará el algoritmo de agrupamiento borroso para obtener el modelo.

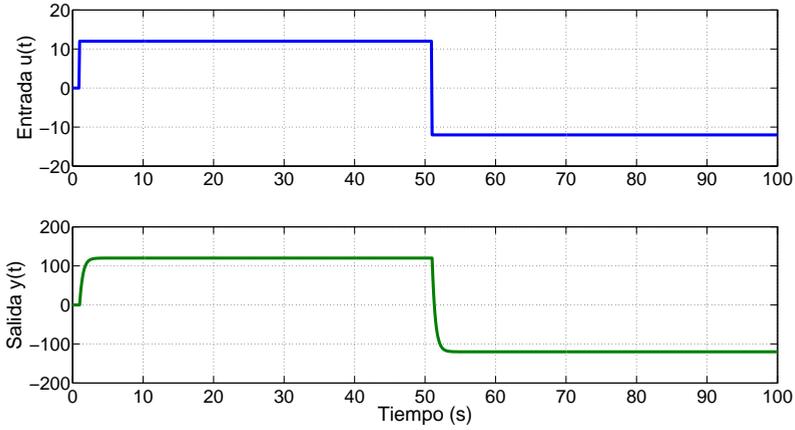


Figura 4.1: Señales para la identificación.

Dado que la planta es lineal es necesario únicamente un *clúster* para realizar la identificación. Los parámetros elegidos en el algoritmo de identificación son dados en la tabla 4.1¹.

TABLA 4.1: PARÁMETROS DEL ALGORITMO DE AGRUPAMIENTO

Número de clústers	<i>.c</i> = 1
Coefficiente de borrosificación	<i>.m</i> = 2
Tolerancia de terminación	<i>.tol</i> = 0.01
Semilla para la inicialización	<i>.seed</i> = 1
Tipo de los antecedentes	<i>.ante</i> = 2
Estimación del consecuente	<i>.cons</i> = 1
Número de retardos de la salida	<i>.Ny</i> = 1
Número de retardos de la entrada	<i>.Nu</i> = 1
Número de retardos de transporte	<i>.Nd</i> = 1

En el proceso de validación (figura 4.2) se obtiene un $VAF = 100\%$ y el modelo borroso obtenido presenta una única regla dada en (4.6).

$$\left. \begin{array}{l} \text{If } y(k-1) \text{ is } A_{11} \text{ and } u(k-1) \text{ is } A_{12} \text{ then} \\ y(k) = 8.00 \cdot 10^{-1}y(k-1) + 2.00 \cdot 10^0u(k-1) + 1.24 \cdot 10^{-12} \end{array} \right\} \quad (4.6)$$

¹Una descripción detallada de estos parámetros se encuentra en la referencia [19].

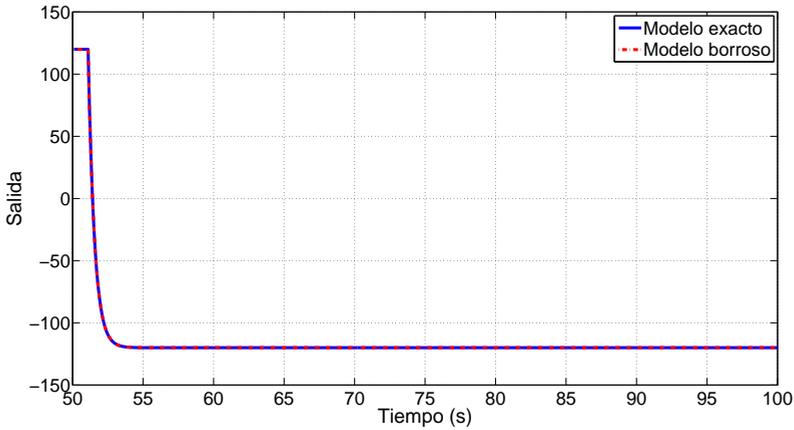


Figura 4.2: Resultado de la validación.

El consecuente de la regla (4.6) obtenida coincide con la ecuación en diferencias de la planta (4.5), puesto que el término $1.24 \cdot 10^{-12}$ se puede despreciar. Por lo tanto, y tal como refleja el valor del VAF, la identificación ha sido exacta.

Sección 4.3

Modelo de referencia

En el modelo de referencia deben quedar especificadas las características que se quieren obtener del sistema final controlado, tanto en el régimen transitorio como en el régimen permanente. Para establecer este modelo de referencia, se debe tener un buen conocimiento de la dinámica de la planta y de ciertas características operativas, como pueden ser los límites máximos, tanto de entrada como de salida, o la velocidad de respuesta. Esta información se puede conseguir a partir de las características técnicas de que se dispongan y de las pruebas empíricas realizadas en el proceso de identificación.

Se puede establecer como señal de salida del modelo de referencia, $y_r(t)$, la de un sistema de primer orden, con respuesta monótona, y error en régimen permanente igual a cero ante entrada escalón. Con ello lo que se consigue es que si se imponen cambios en la consigna en forma de escalón, la salida alcance el valor final de forma suave, sin sobreimpulsos, y sin error en régimen permanente. Con estas especificaciones, el modelo de referencia deberá ser de tipo 1, con un polo en $z = 1$ en lazo abierto [152]. A partir del esquema de la

figura 4.3 se obtiene el modelo (4.7). Realizando la transformada inversa Z de $Y_r(z)$ de (4.7), para entrada en escalón de amplitud A , se obtiene la respuesta del modelo de referencia $y_r(k)$ (4.8), siendo k el número de muestra.

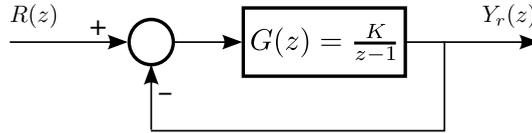


Figura 4.3: Sistema para calcular el modelo de referencia.

$$G_r(z) = \frac{Y_r(z)}{R(z)} = \frac{K}{z - (1 - K)} \quad (4.7)$$

$$y_r(k) = A - A \cdot (1 - K)^k \quad (4.8)$$

Para el cálculo de K , se debe fijar el tiempo de establecimiento del transitorio requerido, siendo para ello necesario, tal como se ha comentado anteriormente, tener conocimiento de la dinámica de la planta. Una vez calculada K , la ecuación en diferencias que describe al modelo de referencia se obtiene realizando la transformada inversa Z de $Y_r(z)$.

Si, por ejemplo, se quisiera error en régimen permanente cero ante entrada en rampa, habría que fijar un modelo de referencia de segundo orden y tipo 2, quedando la función de transferencia $G(z)$ en lazo abierto, del esquema de la figura 4.3, como (4.9).

$$G(z) = \frac{Y_r(z)}{R(z)} = K \cdot \frac{(z - c_1)}{(z - 1)^2} \quad (4.9)$$

Mediante el método del lugar geométrico de las raíces, se podría calcular la posición del cero c_1 aplicando la condición de fase y de K por la condición de módulo [152, 153].

Para esta planta se establecerá como señal de salida del modelo de referencia, $y_r(t)$, la de un sistema de primer orden, con respuesta monótona, y error en régimen permanente igual a cero ante entrada escalón. Con ello lo que se consigue es que si se imponen cambios en la consigna en forma de escalón, la velocidad angular del motor alcance el valor de la consigna de forma suave, sin sobreimpulsos, y sin error final. Con estas especificaciones, el modelo de referencia deberá ser de tipo 1, con un polo en $z = 1$ en lazo abierto [152]. A partir del esquema de bloques de la figura 4.4 se obtiene el modelo (4.10).

$$G_r(z) = \frac{Y_r(z)}{R(z)} = \frac{K}{z - (1 - K)} \quad (4.10)$$

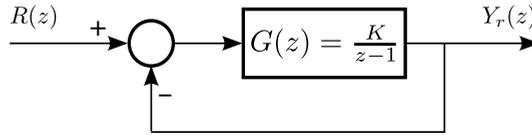


Figura 4.4: Sistema en lazo cerrado.

Realizando la transformada inversa Z de $Y_r(z)$ de (4.10), para entrada en escalón de amplitud A , se obtiene la respuesta del modelo de referencia $y_r(k)$ (4.11), siendo k el número de muestra.

$$y_r(k) = A - A \cdot (1 - K)^k \quad (4.11)$$

Ahora se debe fijar el tiempo de establecimiento requerido, del sistema final controlado, para el cálculo del valor de K . Si en las especificaciones del sistema controlado se indica que se quiere realizar un control de la velocidad angular en el rango de -30 rad/s a $+30 \text{ rad/s}$, y como el motor admite una tensión máxima de entrada de $\pm 12 \text{ V}$, se parte del caso más desfavorable y es que el motor esté funcionando a la velocidad angular de -30 rad/s y se quiera hacer un cambio de dicha velocidad angular a $+30 \text{ rad/s}$. Para ello, la señal de control máxima podría ser de $+12 \text{ V}$. En la figura 4.5 se observa como evolucionaría la velocidad angular y el tiempo que tardaría en cambiar de -30 rad/s a $+30 \text{ rad/s}$ estaría entre 0.2 s y 0.3 s .

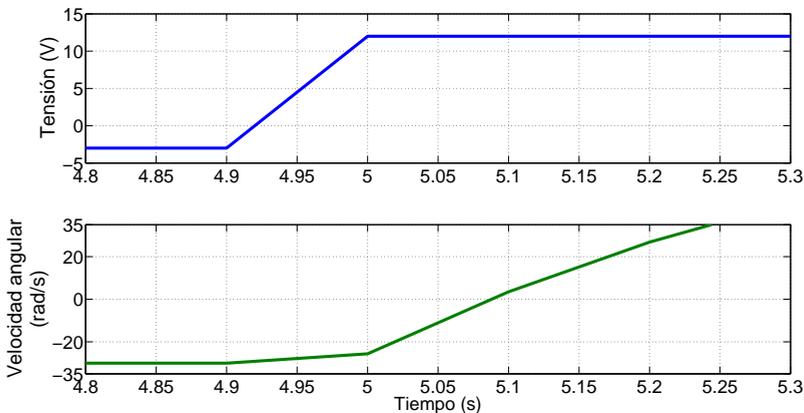


Figura 4.5: Respuesta del motor.

Por lo tanto, el tiempo de establecimiento del modelo de referencia se fijará entre 0.2 s y 0.3 s , y como el periodo de muestreo elegido es $T = 0.1 \text{ s}$, equi-

valdría a conseguir el régimen permanente entre el segundo y tercer periodo de muestreo, o sea, entre $k = 2$ y $k = 3$. Para conseguir esto, el intervalo de valores de K sería el dado en (4.12), pudiéndose elegir una $K = 0.8$ y el modelo de referencia ser el dado en (4.13). Realizando la transformada inversa Z de $Y_r(z)$ se obtiene la ecuación en diferencias (4.14) que describe al modelo de referencia.

$$\left. \begin{aligned} 0.98 \cdot A &= A - A \cdot (1 - K)^2 \Rightarrow K = 0.8586 \\ 0.98 \cdot A &= A - A \cdot (1 - K)^3 \Rightarrow K = 0.7286 \end{aligned} \right\} \quad (4.12)$$

$$G_r(z) = \frac{Y_r(z)}{R(z)} = \frac{0.8}{z - 0.2} \quad (4.13)$$

$$y_r(k) = 0.2 \cdot y_r(k - 1) + 0.8 \cdot r(k - 1) \quad (4.14)$$

Sección 4.4

Diseño del controlador inverso

Una vez que se ha determinado el modelo Takagi-Sugeno de la planta y establecido el modelo de referencia requerido, se procede al diseño del *Controlador Inverso*. En la figura 4.6 se muestra el esquema en diagramas de bloques del sistema de partida propuesto. Se establece un modelo de referencia cuya salida $y_r(t)$ es la que se desea que siga la salida $y(t)$ de la planta a controlar. En definitiva, lo que se pretende conseguir es que $y(t)$ sea igual a $y_r(t)$. Si $y_r(t)$ fuera igual a $r(t)$ el controlador debería ser un inverso perfecto de la planta, lo cual, y según lo discutido en la sección 3.1.2, es imposible de obtener en la práctica. Así pues, al incorporar el filtro, que constituye el modelo de referencia, lo que se está haciendo es crear un inverso aproximado de la planta (sección 3.1.3).

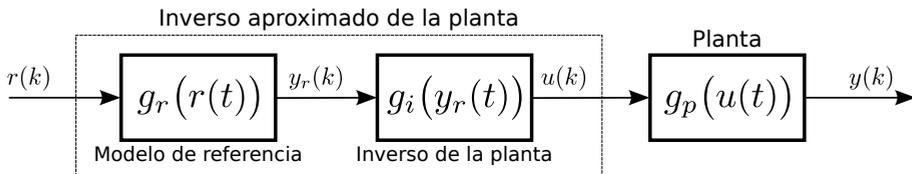


Figura 4.6: Diagrama de bloques del sistema de partida.

Si se quiere que la planta responda según el modelo de referencia (4.13), el esquema de bloques de la figura 4.6 tendría la estructura de la figura 4.7.

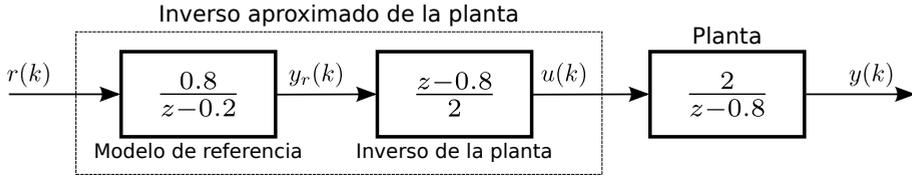


Figura 4.7: Sistema para el control en lazo abierto.

El inverso exacto de la planta no se puede implementar físicamente, pero sí el conjunto modelo de referencia en cascada con el inverso de la planta, que constituiría el controlador, consiguiéndose un inverso aproximado de la planta.

Para obtener el controlador, se parte de las ecuaciones en diferencias que describen a la planta (4.5) y al modelo de referencia (4.14). Igualando la salida de la planta a la salida del modelo de referencia, $y(k) = y_r(k)$, se deduce la expresión de la señal de control $u(k)$ necesaria (4.15), siendo el esquema de bloques equivalente el mostrado en figura 4.8.

$$u(k) = -0.3 \cdot y(k) + 0.4 \cdot r(k) \tag{4.15}$$

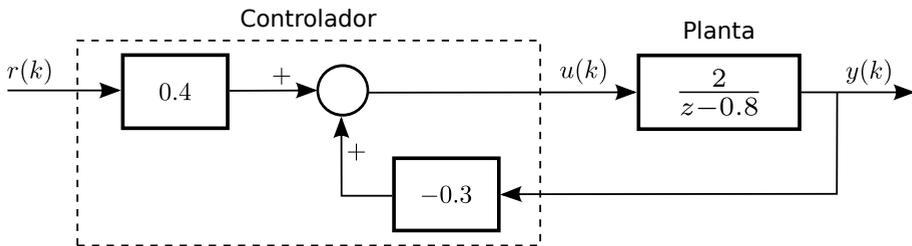


Figura 4.8: Sistema para el control en lazo cerrado.

La función de transferencia del sistema de la figura 4.8, dada en (4.16), da como resultado el modelo de referencia que se había fijado.

$$\frac{Y(z)}{R(z)} = 0.4 \cdot \frac{\frac{2}{z-0.8}}{1 + 0.3 \cdot \frac{2}{z-0.8}} = \frac{0.8}{z-0.2} \tag{4.16}$$

La señal de control $u(k)$ (4.15) se puede generar a partir de una red neuronal con función de activación lineal (4.17).

$$u(k) = \omega_{11} \cdot y(k) + \omega_{12} \cdot r(k) \tag{4.17}$$

Para este caso, la red neuronal consta de una única neurona con dos entradas y una salida. La entrada $r(k)$ tiene un peso de 0.4, la entrada $y(k)$ un peso de -0.3 y la salida $u(k)$ tiene función de activación lineal (figura 4.9).

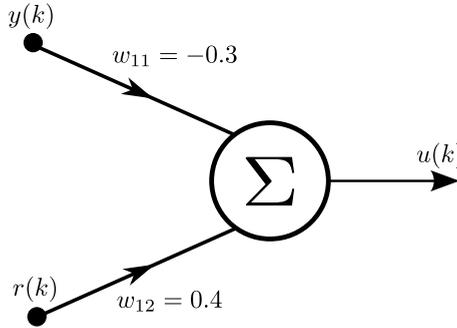


Figura 4.9: Neurona que implementa a $u(k)$.

Sección 4.5

Comportamiento del sistema controlado

En la figura 4.10 se muestra el comportamiento del sistema controlado, frente al deseado del modelo de referencia, al introducir consignas tipo escalón y rampa. Se observa claramente que la señal del sistema controlado coincide con la salida deseada del modelo de referencia, tanto en el transitorio como en régimen permanente, siendo el error prácticamente nulo (figura 4.11). También es importante observar que la señal de control no va a exigir un gran esfuerzo a la planta, una cuestión importante en un sistema controlado.

A continuación se analiza el comportamiento del sistema cuando hay errores de modelado de la planta. Si se supone un cambio en la ganancia y en la constante de tiempo de la planta del $\pm 25\%$, se obtienen cuatro funciones de transferencia (4.18).

$$\left. \begin{aligned} G_{p1}(s) &= \frac{Y(z)}{U(z)} = \frac{12.5}{0.3361 \cdot s + 1} \\ G_{p2}(s) &= \frac{Y(z)}{U(z)} = \frac{12.5}{0.5601 \cdot s + 1} \\ G_{p3}(s) &= \frac{Y(z)}{U(z)} = \frac{7.5}{0.3361 \cdot s + 1} \\ G_{p4}(s) &= \frac{Y(z)}{U(z)} = \frac{7.5}{0.5601 \cdot s + 1} \end{aligned} \right\} \quad (4.18)$$

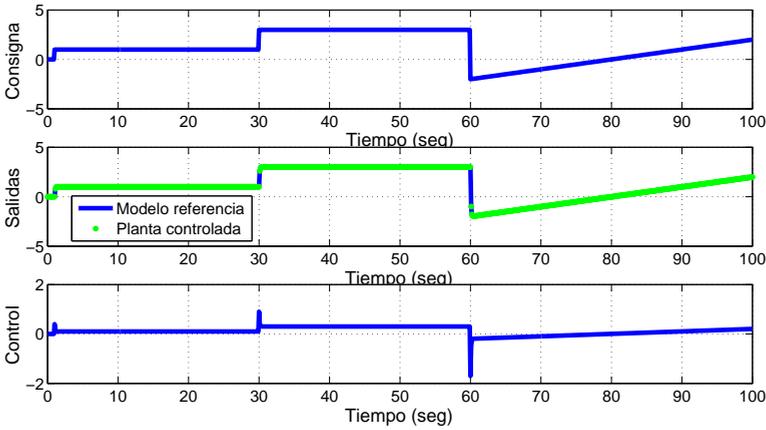


Figura 4.10: Respuesta del sistema controlado.

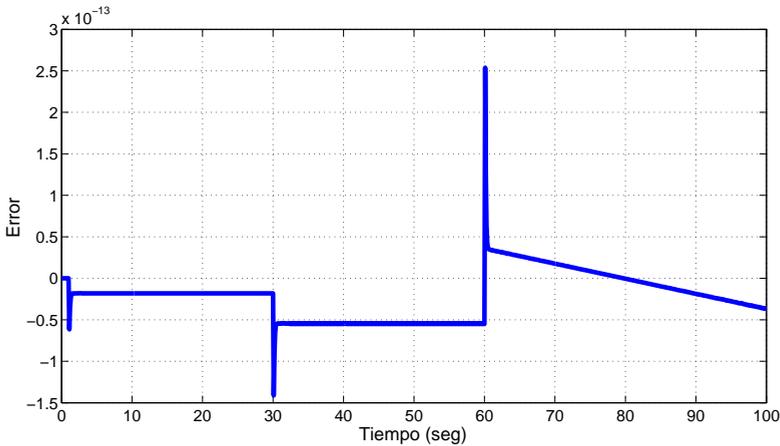


Figura 4.11: Error que se produce en el seguimiento.

En la figura 4.12 se observa que G_{p1} y G_{p4} son las que más se ven afectadas en el transitorio, es decir cuando hay un aumento de uno de los parámetros, en la ganancia o en la constante de tiempo, y una disminución en el otro, siendo el transitorio de las funciones G_{p2} y G_{p3} aceptable. El error en régimen permanente es de $+0.5264 \text{ rad/s}$ para G_{p1} y G_{p2} , -0.7693 rad/s para G_{p3} y -0.7740 rad/s para G_{p4} . Por lo tanto, aunque el sistema es estable, no se puede considerar robusto.

Indudablemente, el periodo de muestreo elegido para el diseño ha sido un poco elevado para la dinámica de la planta, por lo que ahora se disminuye

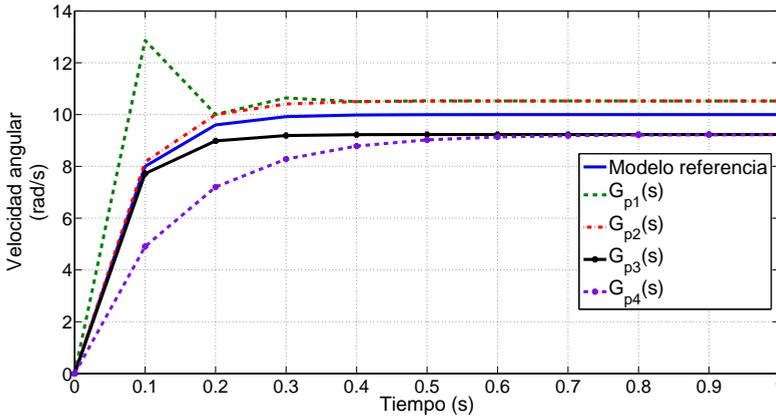


Figura 4.12: Respuesta con errores de modelado y 10 rad/s .

a $T = 0.01 \text{ s}$, obteniéndose las respuestas que se muestran en la figura 4.13. Como se puede apreciar, el transitorio ha mejorado para las cuatro funciones de transferencia, sobre todo en G_{p1} y G_{p4} que con $T = 0.1 \text{ s}$ eran las peores. Por otro lado, en régimen permanente también han disminuido los errores, siendo estos de $+0.3660 \text{ rad/s}$ para G_{p1} y G_{p2} , -0.5558 rad/s para G_{p3} y -0.5585 rad/s para G_{p4} . Al disminuir el periodo de muestreo, el sistema ha ganado en robustez.

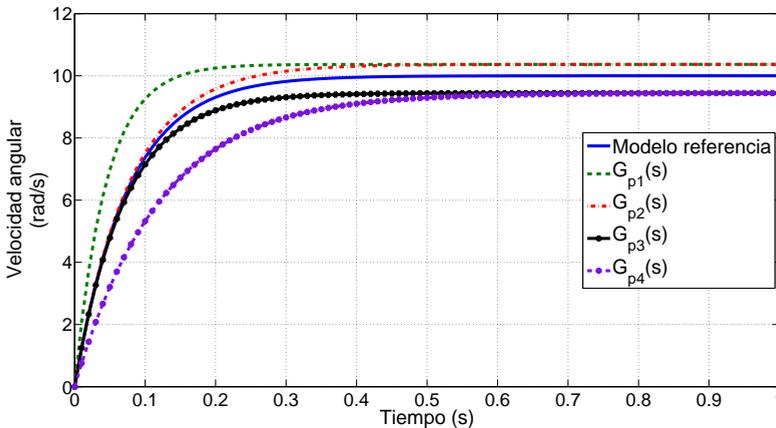


Figura 4.13: Respuesta con errores de modelado con $T = 0.01 \text{ s}$.

Si se disminuye aún más el periodo de muestreo, la mejora no es sustancial,

viéndose el transitorio ligeramente mejorado y los errores en régimen permanente escasamente disminuidos. Por ejemplo, para $T = 0.001 \text{ s}$, los errores son de $+0.3540 \text{ rad/s}$ para G_{p1} y G_{p2} , -0.5391 rad/s para G_{p3} y -0.5414 rad/s para G_{p4} .

Por lo tanto, se puede optar por disminuir el periodo de muestreo para disminuir los errores en el transitorio y en el régimen permanente. Los contras de disminuir el periodo de muestreo es que, además de necesitar un hardware de mayor rapidez de procesamiento, la señal de control va a tener una magnitud mayor. Por ejemplo, en la figura 4.14 se observa la señal de control necesaria para $T = 0.1 \text{ s}$ y $T = 0.01 \text{ s}$.

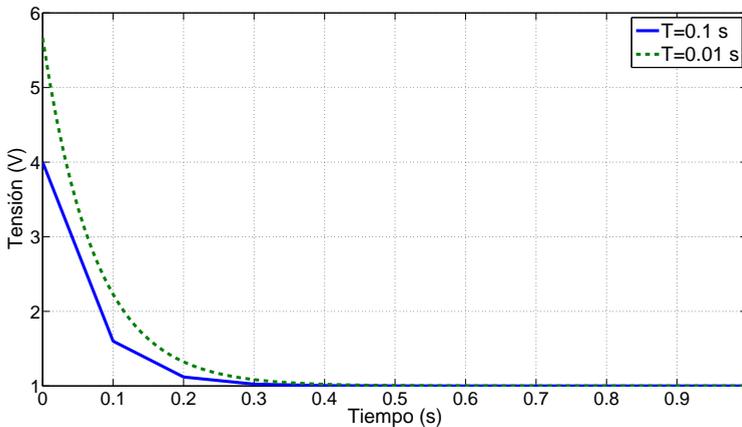


Figura 4.14: Señal de control con 10 rad/s para $T = 0.1 \text{ s}$ y $T = 0.01 \text{ s}$.

Investigando otras alternativas para hacer al sistema más robusto ante errores de modelado de la planta, se ha encontrado que disminuir el tiempo de establecimiento del sistema de referencia, y por lo tanto del sistema controlado, conlleva una mejora en la robustez del sistema, aunque tenga que manejar señales de control de mayores amplitudes que en algunos casos pueden saturar la entrada de la planta. Debido a esta saturación también se disminuye el margen dinámico donde el control sigue las especificaciones. Por ejemplo, si se supone que se trabaja con $T = 0.01 \text{ s}$ y disminuimos el tiempo de establecimiento del modelo de referencia a 0.05 s , los errores en régimen permanente disminuyen. En la figura 4.15 se observan las respuestas para la planta nominal $G_p(s)$ (4.3) y para las cuatro plantas con errores de modelado (4.18). Analizando la figura 4.15, se deduce que en el transitorio el peor comportamiento es para $G_{p1}(s)$ y $G_{p4}(s)$, mientras que los errores en régimen permanente han disminuido y son de $+0.082 \text{ rad/s}$ para G_{p1} y G_{p2} y de -0.1337 rad/s para

G_{p3} y G_{p4} . El transitorio se degrada debido a que la señal de control llega a saturarse durante determinado tiempo, tal como muestra la figura 4.16.

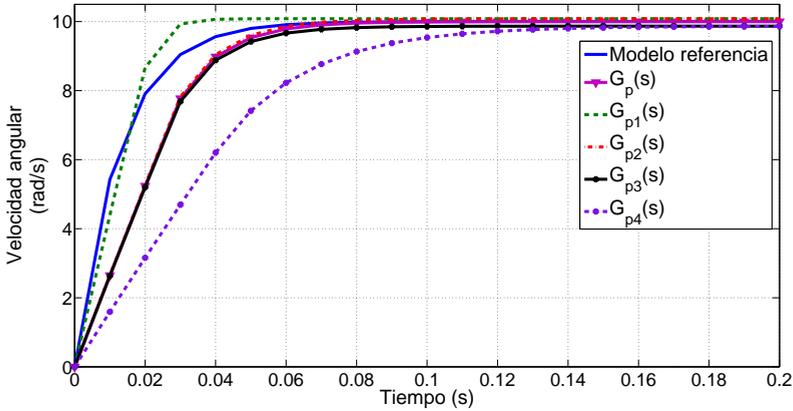


Figura 4.15: Comportamiento para $T = 0.01$ s y $t_e = 0.05$ s.

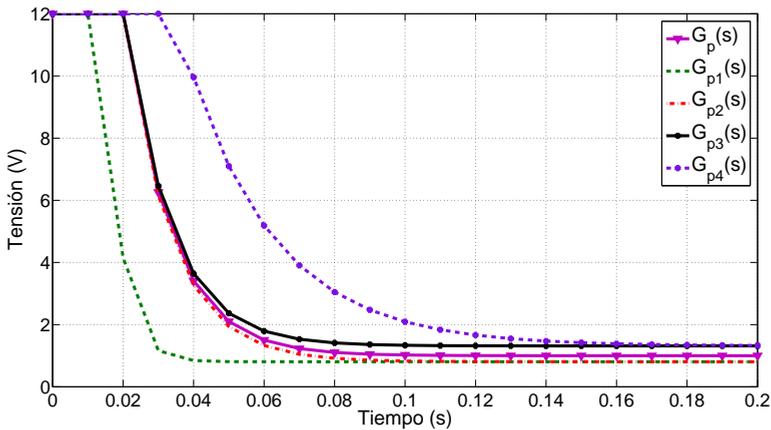


Figura 4.16: Señal de control para $T = 0.01$ s y $t_e = 0.05$ s.

Sección 4.6

Diseño del controlador borroso

Para la eliminación total del error en régimen permanente, debido a los errores de modelado, se propone la incorporación de un controlador proporcional

integral —*Proportional Integral* (PI)— borroso (sección 2.2.5) en la cadena de realimentación (*Controlador Borroso* de la figura 1.1). La función de este controlador borroso será el supervisar si hay error entre la salida del modelo de referencia y la salida de la planta. En el caso de que haya error, aportará una señal correctiva que se sumará a la del *Controlador Inverso*.

Para el PI borroso se elige la estructura de la figura 4.17. Las entradas a la lógica borrosa serán el error e y la derivada del error de , mientras que la salida será la señal uc . La lógica borrosa estará definida por:

- Los conjuntos borrosos de las entradas y salidas.
- Las reglas que relacionan a las variables de entrada y salida con los conjuntos borrosos.
- Los métodos usados para la operación *and*, *or* y *then* de cada regla.
- El método usado para la agregación de todas las reglas.
- El método usado para la desborrosificación final.

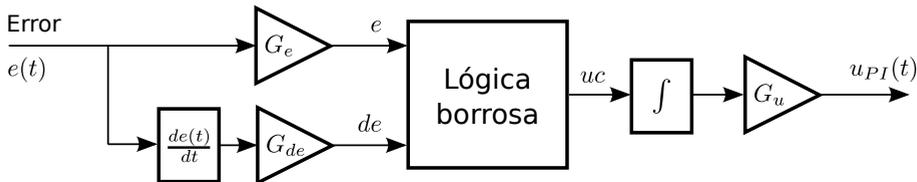


Figura 4.17: Estructura del PI borroso.

Se normalizan los universos de discurso de entrada y salida en el intervalo $[-1, 1]$, y se elige el mínimo número de conjuntos borrosos para las entradas y la salida: *Neg* (negativo), *Cero* (cero) y *Pos* (positivo). Para el conjunto borroso *Cero* se elige un tipo *Singleton*, mientras que para los conjuntos borrosos *Neg* y *Pos* se eligen primero funciones de pertenencia triangulares (figura 4.18) y posteriormente tipo *gaussianas* (figura 4.19). Con estas dos elecciones se realizará una comparativa de resultados obtenidos.

En cuanto a las reglas, se forman según actuaría un operador que quisiese realizar el control observando la evolución de la salida. Dado que el error es la diferencia entre la salida del modelo de referencia y la salida de la planta, para consignas positivas, cuando la salida de la planta tiene un valor inferior a la salida deseada del modelo de referencia el error es positivo y es negativo en el caso opuesto. Para consignas negativas ocurriría lo contrario. Por lo tanto,

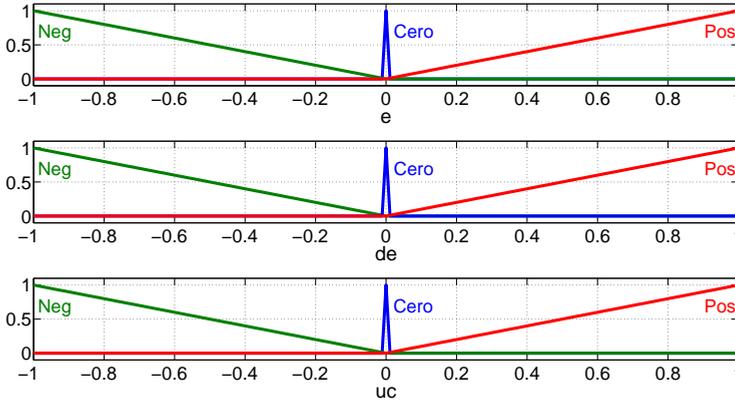


Figura 4.18: PI borroso con conjuntos triangulares.

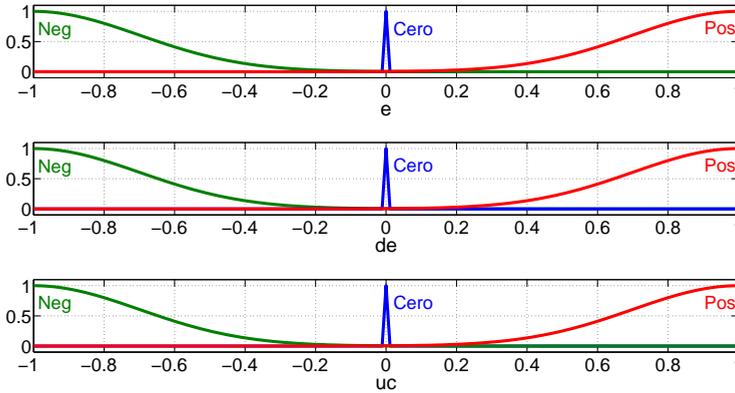


Figura 4.19: PI borroso con conjuntos gaussianos.

se establecen nueve reglas, dadas en (4.19), que definen diferentes estados para el error y para la derivada del error según la respuesta de un sistema subamortiguado. La figura 4.20 muestra la activación de las reglas en el caso de que la referencia fuese positiva.

Para la operación *and* se usa el producto, para la implicación el mínimo, para la agregación la suma y para la desborrosificación el COG. Estas elecciones son las que mejores resultados han ofrecido después de experimentar con diferentes opciones. También, después de las simulaciones realizadas, la elección de las funciones de pertenencia de la figura 4.19 son las que han ofrecido mejores

resultados finales.

$$\left. \begin{aligned}
 R_1 : & \text{If } e \text{ is Pos and } de \text{ is Neg then } uc \text{ is Pos} \\
 R_2 : & \text{If } e \text{ is Cero and } de \text{ is Neg then } uc \text{ is Neg} \\
 R_3 : & \text{If } e \text{ is Neg and } de \text{ is Neg then } uc \text{ is Neg} \\
 R_4 : & \text{If } e \text{ is Neg and } de \text{ is Cero then } uc \text{ is Neg} \\
 R_5 : & \text{If } e \text{ is Neg and } de \text{ is Pos then } uc \text{ is Neg} \\
 R_6 : & \text{If } e \text{ is Cero and } de \text{ is Pos then } uc \text{ is Pos} \\
 R_7 : & \text{If } e \text{ is Pos and } de \text{ is Pos then } uc \text{ is Pos} \\
 R_8 : & \text{If } e \text{ is Pos and } de \text{ is Cero then } uc \text{ is Pos} \\
 R_9 : & \text{If } e \text{ is Cero and } de \text{ is Cero then } uc \text{ is Cero}
 \end{aligned} \right\} \quad (4.19)$$

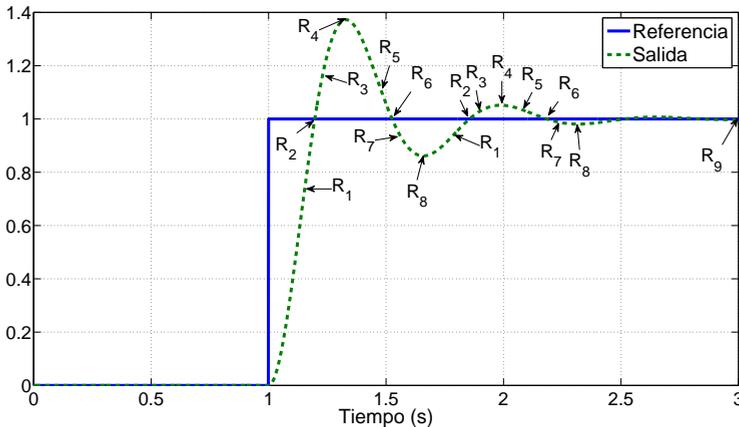


Figura 4.20: Zona en la que se activa cada una de las reglas.

En cuanto a las ganancias, G_e , G_{de} y G_u , se han inicializado al valor 1 y se han ajustado, por el método de prueba y error, en función de la respuesta obtenida. Los valores finales obtenidos han sido $G_e = 2$, $G_{de} = 0.5$ y $G_u = 10$.

En la figura 4.21 se observa la respuesta final para el modelo nominal (4.3) y los cuatro modelos (4.18) con errores de modelado. Estos resultados se han obtenido para un tiempo de muestreo $T = 0.01$ s y un tiempo de establecimiento de modelo de referencia de 0.05 s. En definitiva, la figura 4.21 ofrece los resultados cuando se incorpora el PI borroso a los resultados obtenidos en la figura 4.15. Como se puede apreciar, el PI borroso ha anulado el error en régimen permanente y ha mejorado ligeramente el transitorio.

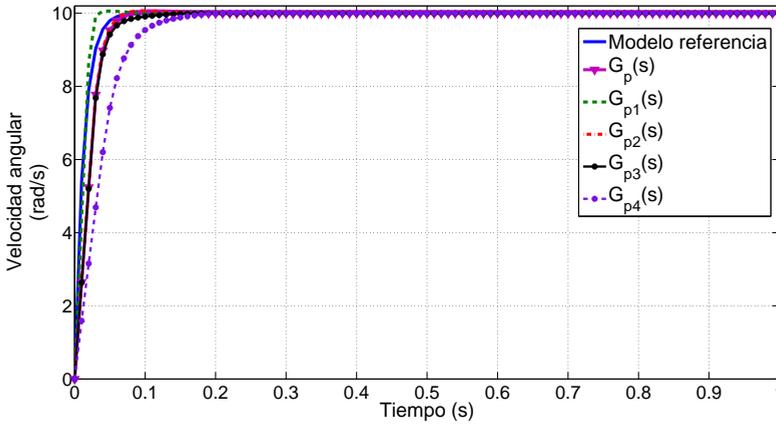


Figura 4.21: Comportamiento para $T = 0.01$ s y $t_e = 0.05$ s.

Sección 4.7

Aportaciones

En este capítulo se ha investigado, sobre todo, de qué parámetros, con la estructura presentada, depende la robustez del sistema, tanto en lo relativo al controlador inverso como al borroso. Se han establecido dos pautas a seguir si se quiere que el sistema conste únicamente de un controlador inverso, resultando una estructura sencilla:

1. Se puede aumentar la robustez disminuyendo el periodo de muestreo del sistema. Esto, como es lógico, supone el tener que emplear un hardware con mayor rapidez de procesamiento. Por otra parte, disminuir el periodo de muestreo por debajo de cierto valor no supone grandes mejoras, incrementándose la magnitud de la señal de control, pudiendo saturar la entrada de la planta durante un intervalo largo de tiempo y a la vez disminuir el margen dinámico de trabajo.
2. El valor del tiempo de establecimiento fijado en el modelo de referencia influye de manera notable en la robustez. Para el ejemplo mostrado en este capítulo, la disminución del tiempo de establecimiento en el modelo de referencia supuso una disminución importante en el error en régimen permanente.

Si la aplicación no permite los errores en régimen permanente, ante errores de modelado, se puede incluir un PI borroso en la cadena de realimentación

que complemente al controlador inverso. La segunda parte de este capítulo se ha centrado en investigar como diseñar este PI borroso para hacer más robusto al sistema ante incertidumbres en el modelado. Las conclusiones que se han podido obtener sobre su diseño, después de realizar múltiples simulaciones con distintas combinaciones, se resumen en los siguientes puntos:

1. El menor número de conjuntos borrosos necesarios para las entradas y salida se establecen en tres: *negativo*, *cero* y *positivo*.
2. Los mejores resultados se obtienen para funciones de pertenencia tipo *gaussiana* para los conjuntos borrosos *negativo* y *positivo*, mientras que para el conjunto borroso *cero* del tipo *Singleton*.
3. Se establecen en nueve reglas las necesarias para cubrir de forma apropiada la posible evolución de la salida.
4. Los métodos usados en las distintas operaciones, llevadas a cabo por la lógica borrosa, también tienen una importante influencia en el resultado final obtenido. Los mejores operadores han sido: el producto para la operación *and*, el mínimo para la implicación *then*, la suma para la agregación y el COG para la desborrosificación.
5. Las ganancias del PI borroso se pueden sintonizar por el método de prueba y error sin invertir un excesivo tiempo.

La planta objeto de control en este capítulo es la misma que la que se usará en el siguiente capítulo para la compensación de las perturbaciones de carga, formando parte de las publicaciones presentadas en:

Betancor, C. S., Sosa-González, C. J., Montiel, J. A., Vega, A., “PI Fuzzy System Gains Tuning Using Genetic Algorithms”, EUROGEN 2011, Conference Proceedings, pp 238-243, Capua, Italy, 2011.

Betancor, C. S., Sosa-González, C. J., Montiel, J. A., Vega, A., “Gains Tuning of a PI-Fuzzy controller by genetic algorithms”, Engineering Computations, Emerald, Vol. 31, Issue 6, pp 1074-1097, 2014.

Compensación de perturbaciones

En el capítulo anterior se expuso la metodología propuesta para el desarrollo del sistema de control. Mediante un ejemplo, consistente en el control de la velocidad angular de un motor DC PM, se diseñó el controlador inverso. Además, se incorporó un controlador PI borroso para hacer al sistema más robusto ante errores de modelado de la planta.

En este capítulo se extiende el diseño del PI borroso, que pertenece a la estructura de realimentación y complementa al controlador inverso, para que, aparte de compensar las incertidumbres de modelado, pueda corregir y minimizar los efectos debidos a las perturbaciones de carga. Para ello se usará la misma planta que en el capítulo anterior, consistente en el control de velocidad angular de un motor DC PM. Aparte de establecer todos los parámetros del PI borroso, la investigación se centrará en sintonizar las ganancias de este controlador por medio de un algoritmo genético, estableciendo la influencia de las diferentes variables del algoritmo. Para conseguir los resultados propuestos, se ha diseñado una nueva función objetivo, que el algoritmo genético deberá minimizar, para que la respuesta obtenida ante la perturbación de carga sea lo más parecida posible a la que ofrecería un modelo de rechazo de perturbación, el cual se fija en función de la respuesta deseada. Para corroborar los resultados que se obtuvieron con el ejemplo del control de velocidad de un motor DC PM, también se aplica a la compensación de las perturbaciones de carga en un sistema motor-generador y se compara el resultado con otra técnica propuesta en la literatura. En el capítulo 9 se aplica la metodología, desarrollada en este capítulo, para la compensación de las perturbaciones de carga en un convertidor DC/DC conmutado.

El siguiente capítulo se centrará en el diseño del controlador inverso, para sistemas lineales de segundo y tercer orden, y se discutirán algunos aspectos a tener en cuenta para obtener un inverso aproximado.

El capítulo está organizado como sigue. En la sección 5.1 se expone el problema a resolver y la solución adoptada. La sección 5.2 se centra en la descripción de la función objetivo creada para el algoritmo genético. La elección y el modo de cálculo del modelo de rechazo de perturbación se explica en la sección 5.3. Con el ejemplo de diseño mostrado en 5.4, consistente en la compensación del par de carga en el control de velocidad de un motor, se realiza un análisis de los parámetros que influyen en el algoritmo genético para conseguir una sintonización eficaz y cumplir con los requerimientos impuestos. En la siguiente sección 5.5 se aplica la metodología analizada para el rechazo del par de carga en un sistema motor-generator y se comparan los resultados obtenidos con otra técnica propuesta en la literatura. Finalmente, en la sección 5.6 se exponen las aportaciones realizadas, incluyendo las publicaciones llevadas a cabo sobre la temática expuesta en este y en el capítulo anterior.

Sección 5.1

Perturbación de carga

Uno de los principales objetivos de los sistemas de control usados en la industria es el rechazo de las perturbaciones [154], debido a que el comportamiento dinámico de las plantas depende de estas. Una de las fuentes más comunes de perturbaciones son debidas a cambios en las cargas. Una tarea fundamental del sistema de control será minimizar los efectos de las perturbaciones de carga tan rápido como sea posible. Otro importante objetivo es la forma en que las perturbaciones de carga son rechazadas [155, 156]. Muchos fabricantes se esfuerzan por reducir los sobreimpulsos después de una perturbación. Algunos ejemplos ilustrativos son:

- En los convertidores de potencia evitar los picos de tensión que pueden ser perjudiciales para las cargas.
- Es importante evitar los desbordamientos en estanques con sustancias peligrosas.
- En diversos sistemas de control de posicionamiento no es aconsejable la existencia de sobreimpulsos, como por ejemplo en el control de un ascensor.

- En algunos sistemas térmicos no es recomendable rebasar la medida deseada, como en el caso del control de la temperatura en las unidades de cuidados intensivos neonatales.

Entre las diferentes alternativas existentes para resolver este problema, se ha optado por el diseño de un controlador PI borroso, el cual actuará según la estructura típica de un sistema de control realimentado (figura 5.1).

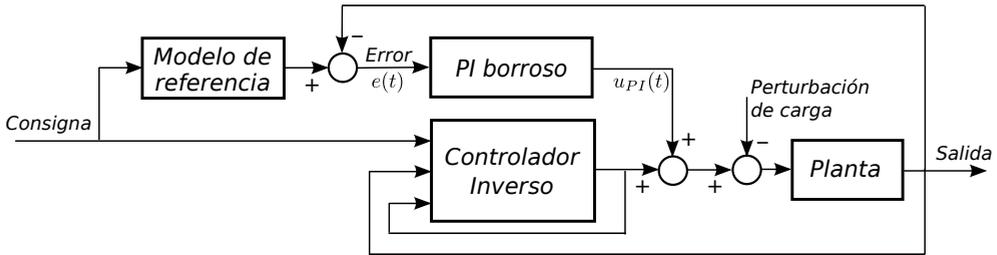


Figura 5.1: PI borroso en la realimentación.

Existen múltiples variables que hay que determinar en un PI borroso (figura 5.2) jugando un papel importante las ganancias G_e , G_{de} y G_u [157]. La sintonización de estas ganancias usando métodos de prueba y error o métodos analíticos requieren un tiempo y esfuerzo computacional elevado. Aquí se propone una solución basada en un algoritmo genético —*Genetic Algorithm* (GA)— para obtener los valores de estas ganancias. Las comparativas frente a otros métodos, demuestran la calidad de las soluciones obtenidas con un tiempo y un esfuerzo computacional moderado.

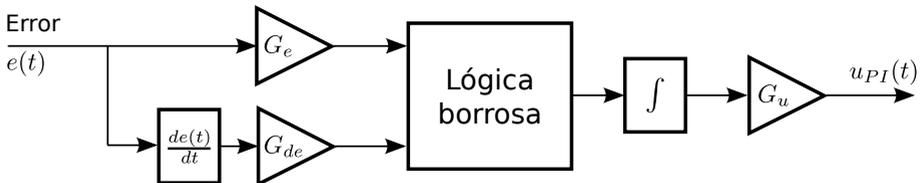


Figura 5.2: Estructura del PI borroso.

Como se indicó con anterioridad, la sintonización de las ganancias del PI borroso se puede llevar a cabo de diferentes formas:

- **Método de prueba y error.** El diseñador, normalmente, no puede garantizar que el sistema diseñado tenga unas buenas prestaciones.

- **Analíticamente.** Este método es engorroso y se deberán realizar simplificaciones en el sistema borroso, tal como el uso de funciones de pertenencia de forma triangular o trapezoidal [158]. El tiempo que se debe emplear usando este método puede que no justifique los resultados obtenidos.
- **Usando un GA.** La sintonización de los parámetros de un controlador borroso basada en un GA [159] es una herramienta potente. El GA muestra su gran adaptabilidad para encontrar los mejores parámetros de las ganancias.

Por lo comentado anteriormente, se empleará un GA para la obtención de los valores óptimos de las ganancias G_e , G_{de} y G_u , minimizando el tiempo empleado en su sintonización, la cual se realiza *offline*, evitando el costo computacional cuando el sistema está operando normalmente.

Sección 5.2

La función objetivo

Cuando se aplica una perturbación al sistema, la salida de este debe verse afectada lo mínimo y durante el menor tiempo posible. La figura 5.3 muestra el posible comportamiento de un sistema controlado ante una perturbación en forma de escalón, debiendo el controlador cancelar el efecto indeseado en la salida.

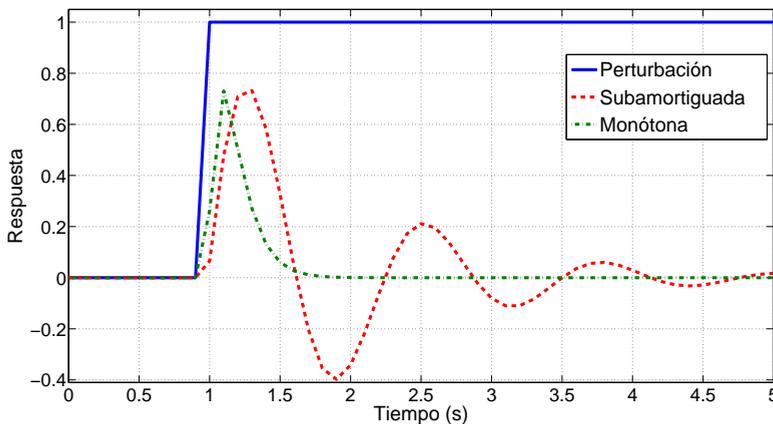


Figura 5.3: Posibles respuestas ante una perturbación en escalón.

La función objetivo del GA determinará si la respuesta del sistema completo ante una perturbación específica es aceptable o no. Dependiendo de la función objetivo, la respuesta puede ser subamortiguada o con forma monótona, como ilustra la figura 5.3. El interés está centrado en obtener un rechazo a la perturbación en forma monótona y evitar sobreimpulsos que, como se ha dicho anteriormente, son muy perjudiciales en ciertos sistemas. Para conseguir este propósito, se ha definido una nueva función objetivo para el GA.

En un sistema de control, la función objetivo o de evaluación (sección 2.4.3) normalmente se define como la integral del cuadrado del error —*Integral Square Error* (ISE)— [160], la integral del tiempo multiplicado por el cuadrado del error —*Integral of Time multiplied by the Squared Error* (ITSE)—, la integral del valor absoluto del error —*Integral of Absolute Error* (IAE)— [161] o la integral del tiempo multiplicado por el valor absoluto del error —*Integral of Time multiplied by Absolute Error* (ITAE).

De los experimentos realizados, las funciones objetivos tradicionales, tales como ISE, ITSE, IAE o ITAE, no ofrecen la respuesta deseada. Por ello, se ha definido una nueva función objetivo, cuya expresión se presenta en (5.1), y que se ha denominado integral del cuadrado del error o de la raíz cuadrada del valor absoluto del error —*Integral Square Error Or Integral Square Root of Absolute Error* (ISEOISRAE).

$$f = \left[\int e_1^2(t) dt \right] + \left[\int \sqrt{|e_2(t)|} dt \right] \quad (5.1)$$

El error es la diferencia entre la salida deseada ante la perturbación, $u_d(t)$, y la salida real de la planta, $u_r(t)$. En la ecuación (5.1), el error etiquetado como $e_1(t)$ aparece cuando la salida de la planta tiene el mismo signo que la salida deseada y el error $e_2(t)$ en caso contrario (figura 5.4). El propósito de definir $e_1(t)$ y $e_2(t)$ de esta manera es intentar evitar que se produzcan variaciones en los signos entre la salida real y la deseada.

Un análisis de la salida esperada es el siguiente:

1. El signo de la salida de la planta es igual al de la salida deseada. En este caso, $e_1(t)$ es la diferencia entre la salida deseada y la salida real, mientras que $e_2(t)$ es cero. Para este caso, la ecuación (5.1) es la función objetivo ISE.
2. El signo de la salida de la planta es diferente al signo de la salida deseada. Este caso se deberá al comportamiento con sobreimpulso de la planta. Aquí, $e_1(t)$ es cero, mientras que $e_2(t)$ será la diferencia entre la salida deseada y la salida real. Así la ecuación (5.1) es la integración de la raíz cuadrada del valor absoluto del error.

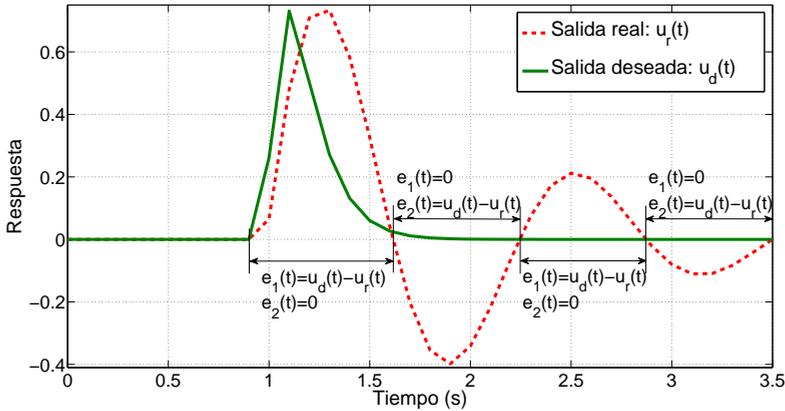


Figura 5.4: Salida deseada y salida real.

Si el valor absoluto de $e_1(t)$ es igual al de $e_2(t)$ e inferior a uno, $e_1^2(t)$ tendrá un valor más pequeño que $\sqrt{|e_2(t)|}$ y por lo tanto la prioridad del GA será minimizar el segundo sumando de la función objetivo (5.1), reduciendo o anulando el sobreimpulso de la planta.

Por otro lado, si el orden de la raíz se hace mayor de dos, se pone demasiado énfasis sobre los errores $e_2(t)$ y el comportamiento del sistema será peor para los errores $e_1(t)$, con lo cual se degrada la respuesta y el comportamiento del sistema no va a ser el deseado.

La función objetivo propuesta ISEOISRAE da mejores resultados que las funciones ISE o IAE, ya que en estas dos últimas no se tienen en cuenta si hay sobreimpulsos.

Para la función de aptitud se ha usado la de asignación de aptitud proporcional que fue explicada en la sección 2.4.3.

Sección 5.3

Modelo de rechazo de la perturbación

Desde el punto de vista de las perturbaciones, para el diseño del controlador borroso se debe establecer en primer lugar un modelo de rechazo de la perturbación, o lo que es lo mismo, definir como se quiere que responda la salida de la planta ante una perturbación.

En la figura 5.3 se mostraron los tipos de respuestas ante perturbación en escalón. Este trabajo se centrará en el rechazo de perturbación en forma monótona y para ello se deberán establecer la amplitud del pico de la respuesta

monótona y el tiempo necesario para rechazar la perturbación en el modelo de rechazo de perturbación. Para ello es necesario ciertos parámetros de la dinámica de la planta, como la máxima señal de entrada y su constante de tiempo dominante.

El modelo de rechazo de perturbación, $G_{DR}(s)$, se puede elegir según la forma expresada en (5.2) ó (5.3).

$$G_{DR}(s) = \frac{Y_{DR}(s)}{D(s)} = \frac{a \cdot s}{(s + b)} \quad (5.2)$$

$$G_{DR}(s) = \frac{Y_{DR}(s)}{D(s)} = \frac{a \cdot s}{(s + b)^2} \quad (5.3)$$

Los coeficientes a y b controlan la forma que tendrá la respuesta ante entrada escalón. Si la entrada es un escalón de amplitud A , $D(s) = A/s$, la respuesta en el tiempo del modelo (5.2) será la expresada por la ecuación (5.4) y la del modelo (5.3) la indicada en (5.5).

$$y_{PR}(t) = A \cdot a \cdot e^{-b \cdot t} \quad (5.4)$$

$$y_{PR}(t) = A \cdot a \cdot t \cdot e^{-b \cdot t} \quad (5.5)$$

La diferencia en la respuesta de ambos modelos se ve claramente en la figura 5.5. En este caso se ha elegido $a = 1.8$ para (5.2) y $a = 50$ para (5.3), de forma que el valor de pico en ambas respuestas tengan el mismo valor. Los polos se han situado en $s = -10$, o sea, $b = 10$.

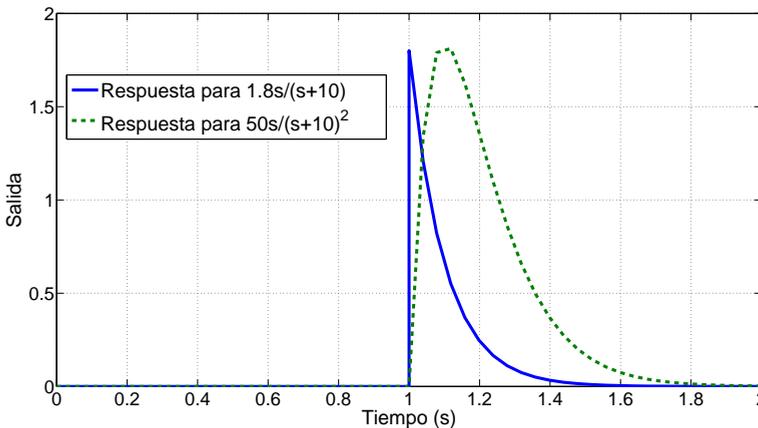


Figura 5.5: Respuesta de los modelos (5.2) y (5.3).

Como se refleja en la figura 5.5, y aunque la respuesta del modelo (5.4) es más rápida en cancelar el efecto de la perturbación, la respuesta del modelo (5.5) es más suave en el inicio de la perturbación. Además, la rapidez en cancelar el efecto de la perturbación se controlaría con el parámetro b , es decir, con la posición de los polos. Por otro lado, el valor del pico se controla con el parámetro a , lo cual se puede deducir fácilmente de las ecuaciones (5.4) y (5.5). Por lo comentado, se elegirá el modelo de rechazo de perturbación como el expresado en (5.3). Para comenzar con el diseño del PI borroso, se debe establecer en primer lugar un modelo de rechazo de la perturbación, o lo que es lo mismo, definir como se quiere que responda la salida de la planta ante una perturbación de carga. El ejemplo que se muestra en esta sección es ilustrativo sobre la metodología a seguir en el diseño. Con este ejemplo lo que se pretende es demostrar que es posible conseguir una respuesta del sistema ante perturbación parecida a la establecida por un modelo de rechazo de perturbación.

En la figura 5.3 se mostraron los tipos de respuesta ante perturbación en escalón. El modelo de rechazo de perturbación deberá tener la respuesta monótona. La amplitud del pico de la respuesta monótona y el tiempo necesario para rechazar la perturbación se establecerá en dicho modelo de rechazo de perturbación. Para ello es necesario conocer parámetros de la dinámica de la planta. En el caso que se propone en esta sección, sería la amplitud máxima de la tensión de entrada al motor y la constante de tiempo del motor.

Sección 5.4

Compensación de la perturbación del par de carga

5.4.1. Introducción

Este caso de aplicación industrial ilustrará la eficiencia del método propuesto para reducir los efectos de las perturbaciones del par de carga en la velocidad angular de un motor DC PM.

El motor objeto de estudio será el mismo que el estudiado en el capítulo anterior y cuya función de transferencia (5.6) se reproduce otra vez aquí para mayor claridad, siendo $\omega(s)$ la velocidad angular en radianes por segundo (rad/s) y $v_A(s)$ la tensión aplicada a la armadura del motor en voltios (V).

$$G_P(s) = \frac{Y(s)}{U(s)} = \frac{\omega(s)}{v_A(s)} = \frac{10}{0.4481 \cdot s + 1} \quad (5.6)$$

Un aumento o disminución de la carga afectaría a la ω , o visto de otra forma,

y desde un punto de vista de modelado implicaría una reducción o aumento de la v_A (figura 5.6).

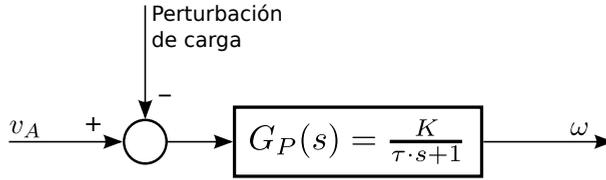


Figura 5.6: Forma de modelar la perturbación de carga.

Todo lo relacionado con el diseño del controlador inverso se expuso en el capítulo anterior, incluido el estudio del comportamiento del sistema para errores de modelado de la planta y como compensarlos, pero no se estudió dicho comportamiento frente a perturbaciones de carga. En la figura 5.7 se ve la evolución de la salida del sistema cuando se aplica una perturbación de $-0.5 V$ de amplitud en el instante $t = 10 s$, siendo el periodo de muestreo $T = 0.1 s$. Tal como se puede observar, aunque la señal de control aumentó su valor, fue incapaz de cancelar la perturbación y la salida del sistema no pudo ser controlada.

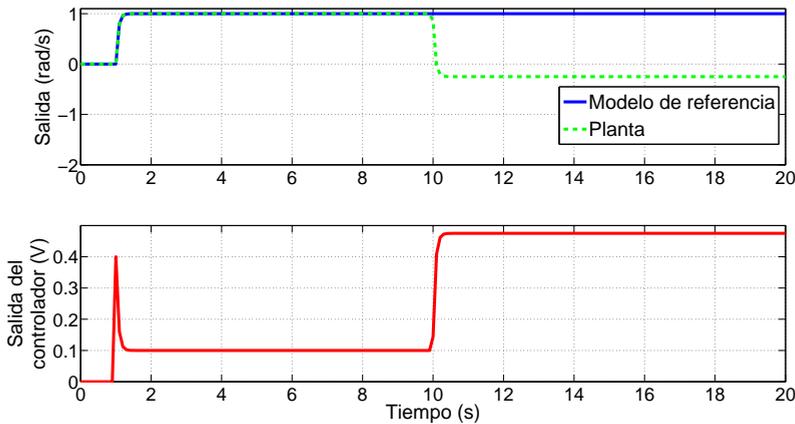


Figura 5.7: Comportamiento ante una perturbación de carga.

Si se disminuye el periodo de muestreo, al igual que sucedía con los errores de modelado, el efecto de la perturbación es menor, pero la señal de control aumenta. A continuación se propone un método para rechazar la perturbación sin tener que disminuir el periodo de muestreo. Este método consistirá en incorporar y diseñar un PI borroso.

5.4.2. Diseño del PI borroso

Para esta aplicación, se usará la estructura presentada en la figura 5.2. En primer lugar hay que fijar los conjuntos borrosos de las entradas y las salidas y la base de reglas asociada.

Para la entrada error (e) se definieron cinco conjuntos borrosos: *NegG* (negativo grande), *Neg* (negativo), *Cero* (cero), *Pos* (positivo) y *PosG* (positivo grande). Para la entrada derivada del error (de) y para la salida (uc) se fijaron tres conjuntos borrosos: *Neg* (negativo), *Cero* (cero) y *Pos* (positivo). Las tres variables borrosas se normalizaron en el rango $[-1, 1]$.

La forma del conjunto borroso (triangular, *gaussiana*, etc.) va a determinar la calidad de los resultados obtenidos. Después de diversas simulaciones, se encontró una solución satisfactoria con una superficie de control no lineal como se muestra en la figura 5.8. Esta superficie de control se consiguió al usar la función *singleton* para el conjunto borroso *Cero* (Cero) y la función *gaussiana* para el resto de los conjuntos borrosos (figura 5.9).

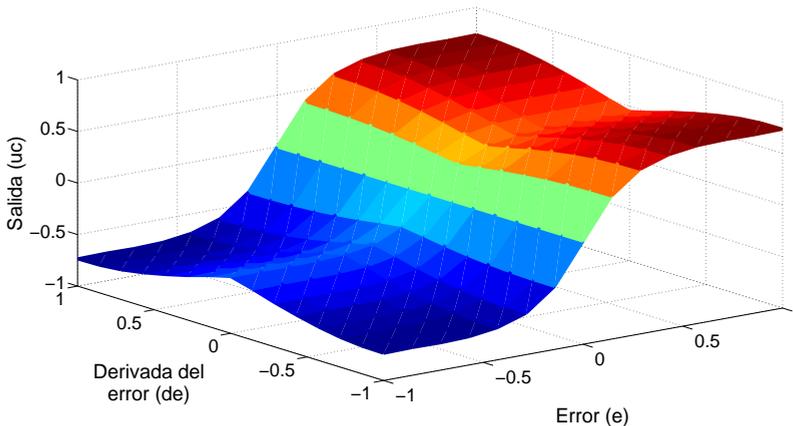


Figura 5.8: Superficie de control del PI borroso.

Analizando la superficie de control se deduce que hay una mayor sensibilidad para el error entorno al valor 0 y esta sensibilidad disminuye conforme el error aumenta, de forma que a partir de un error aproximado de $e = |0.5|$, la señal de salida es constante. Con este comportamiento lo que se pretende conseguir es que desde que la señal de salida difiera de la de referencia, el PI borroso intente cancelar esta desviación, pero no de una forma brusca, para que no se produzcan oscilaciones. También se ve que hay un pequeño pliegue de la superficie de control entorno al valor 0 de la derivada del error. Conforme la

derivada del error aumenta, se incrementa ligeramente el valor de la señal de salida. Esto lo que produce es un efecto de suavizado para ayudar a que no se produzcan oscilaciones. Posteriormente, con las ganancias G_e , G_{de} y G_u se podrá controlar de forma más exacta el efecto de la superficie de control no lineal.

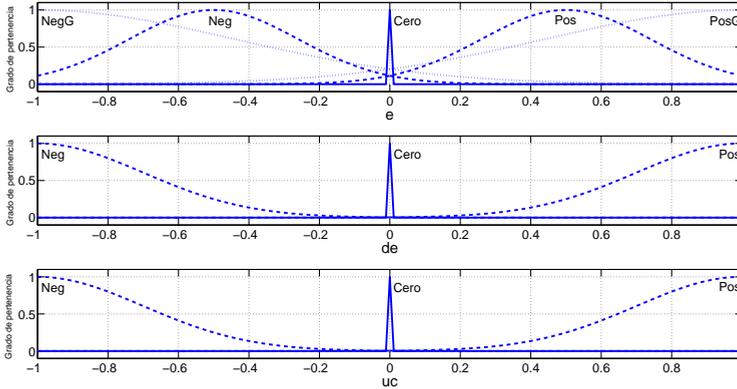


Figura 5.9: Conjuntos borrosos de las entradas y salida.

La base de reglas usada es la de Mamdani [162] y las reglas se basan en que el controlador debería actuar sobre la planta de manera similar a la de la experiencia de un operador, de forma que la salida del sistema siga a la salida del modelo de rechazo a la perturbación [163]. Para esta aplicación la base de reglas contiene once reglas (5.7), las cuales cubren diversos estados de las entradas error y derivada del error.

En la figura 5.10 se aclara donde se activaría cada una de las reglas. Por ejemplo, la regla R_5 se define para cuando la salida de la planta excede a la salida del modelo de referencia, es decir, cuando el error es grande y negativo ($NegG$), y este error se incrementa, es decir, la derivada del error es negativa (Neg). En este caso, el bloque PI borroso activaría el conjunto borroso negativo (Neg) en la salida para reducir el nivel de salida de la planta y acercarse al valor de referencia.

Para el operador *and* se utiliza el método del producto, para la implicación *then* el mínimo, para la agregación la suma y el método COG [164] para el proceso de desborrosificación. Estas elecciones se realizan después de realizar múltiples simulaciones y observando las respuestas obtenidas.

$$\left. \begin{aligned}
 R_1 &: \text{If } e \text{ is PosG and } de \text{ is Neg then } uc \text{ is Pos} \\
 R_2 &: \text{If } e \text{ is Pos and } de \text{ is Neg then } uc \text{ is Pos} \\
 R_3 &: \text{If } e \text{ is Cero and } de \text{ is Neg then } uc \text{ is Cero} \\
 R_4 &: \text{If } e \text{ is Neg and } de \text{ is Neg then } uc \text{ is Neg} \\
 R_5 &: \text{If } e \text{ is NegG and } de \text{ is Neg then } uc \text{ is Neg} \\
 R_6 &: \text{If } e \text{ is NegG and } de \text{ is Pos then } uc \text{ is Neg} \\
 R_7 &: \text{If } e \text{ is Neg and } de \text{ is Pos then } uc \text{ is Neg} \\
 R_8 &: \text{If } e \text{ is Cero and } de \text{ is Pos then } uc \text{ is Cero} \\
 R_9 &: \text{If } e \text{ is Pos and } de \text{ is Pos then } uc \text{ is Pos} \\
 R_{10} &: \text{If } e \text{ is PosG and } de \text{ is Pos then } uc \text{ is Pos} \\
 R_{11} &: \text{If } e \text{ is Cero and } de \text{ is Cero then } uc \text{ is Cero}
 \end{aligned} \right\} (5.7)$$

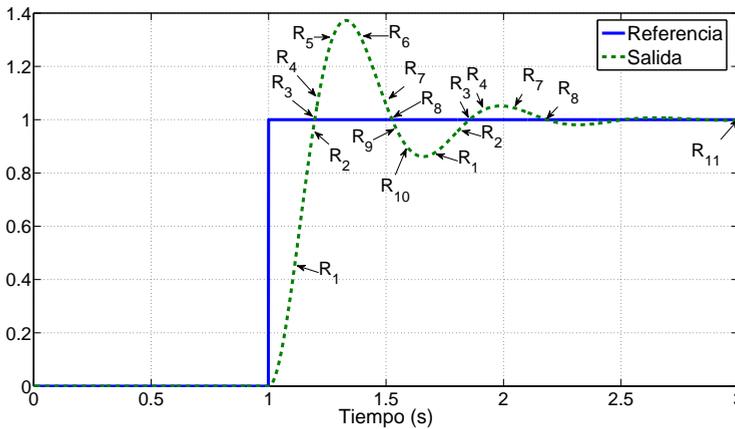


Figura 5.10: Región en la que se activa cada una de las reglas.

5.4.3. Modelo de rechazo elegido

Como se comentó en la sección 5.3, es necesario un conocimiento de la dinámica de la planta para poder establecer el modelo de rechazo de perturbación. Ya se ha dicho que el aumentar el par de carga es equivalente a disminuir la tensión de armadura que se aplica al motor. En este caso se hacen los cálculos para una variación del par de carga equivalente a una variación de $-0.5 V$ en la entrada al motor. El periodo de muestreo elegido es un poco elevado para este motor, concretamente $0.1 s$, sin embargo con ello se mostrará la eficacia

del método propuesto ante condiciones adversas. Al producirse una desviación de $-0.5 V$ en la entrada del motor, la salida en el primer periodo de muestreo decae -1.12 rad/s aproximadamente. Esto supone que el modelo de referencia debe tener un pico mínimo de -1.12 . Elijiendo el modelo (5.3) y el tiempo de rechazo aproximadamente igual a la constante de tiempo de la planta y para el valor de pico mencionado de 1.12 , los valores de a y b serán aproximadamente de 61 y 10 respectivamente, calculados a partir de (5.5).

5.4.4. Parámetros del algoritmo genético

A continuación se realizará un análisis de la influencia de los diferentes parámetros del GA para obtener una solución óptima de los valores de las ganancias, G_e , G_{de} y G_u , del PI borroso. Las simulaciones se han realizado usando el paquete de herramientas para algoritmos genéticos —*Genetic Algorithm Toolbox* (GAT)—, desarrollado en la Universidad de Sheffield [165], siendo una colección de rutinas, escritas en su mayoría en ficheros *.m* para MATLAB[®], que implementa las funciones más importantes para un GA.

El GA tendrá dos objetivos principales:

- Obtener el mínimo valor de la función objetivo. En otras palabras, obtener los valores correctos de las ganancias del controlador para minimizar los efectos de las perturbaciones.
- Forzar un mínimo esfuerzo computacional. Es decir, con un conjunto reducido de individuos y/o un bajo número de evaluaciones llegar a los valores óptimos de las ganancias.

Cada individuo está compuesto de tres variables que representan a las tres ganancias del PI borroso. La representación del cromosoma va a ser un número real debido a la naturaleza del problema.

En primer lugar se estudiará la tasa de reinsertión. La figura 5.11 muestra los mejores valores de la función objetivo (5.1) frente a las evaluaciones, para reinsertiones iguales al 100 % y al 60 %, para 10, 30 y 50 individuos.

De acuerdo a la figura 5.11, con 10 individuos, 100 % de reinsertión y 1684 evaluaciones, se consigue un mínimo de la función objetivo con un valor de 0.1152. Para 30 individuos, 60 % de reinsertión y 2892 evaluaciones, se alcanza un mínimo de la función objetivo de 0.1123.

El efecto de elegir un reemplazamiento generacional, el cual genera una población completamente nueva de individuos en cada paso de la evolución, implica una reinsertión igual al 100 %. Esto da como resultado que hijos menos aptos que sus padres puedan formar parte de la nueva población y de esta

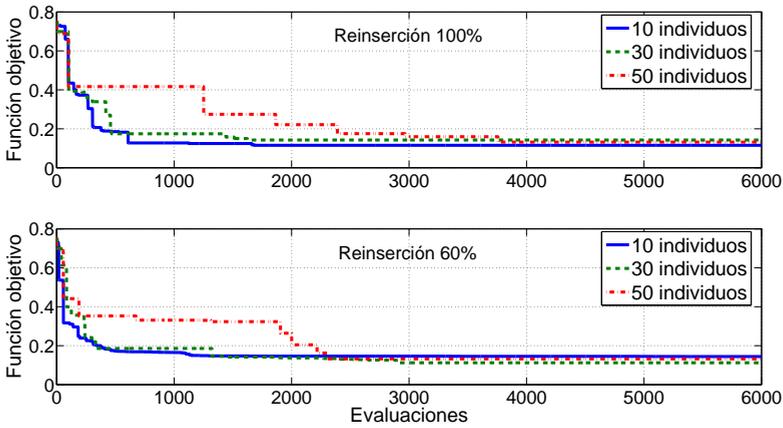


Figura 5.11: Función objetivo al variar la reinserción.

forma la función objetivo puede verse incrementada o decrementada. Se eligió este método porque fue el que dió mejores resultados para bajo número de generaciones y un tamaño de población pequeño. La figura 5.11 muestra que después de 2000 evaluaciones la función objetivo no se reduce más. De esta forma, el criterio de parada es el número de 2000 evaluaciones. Este efecto se puede ver claramente en la figura 5.12, en la que se ha representado los valores mínimos y promedios de la función objetivo de la figura 5.11.

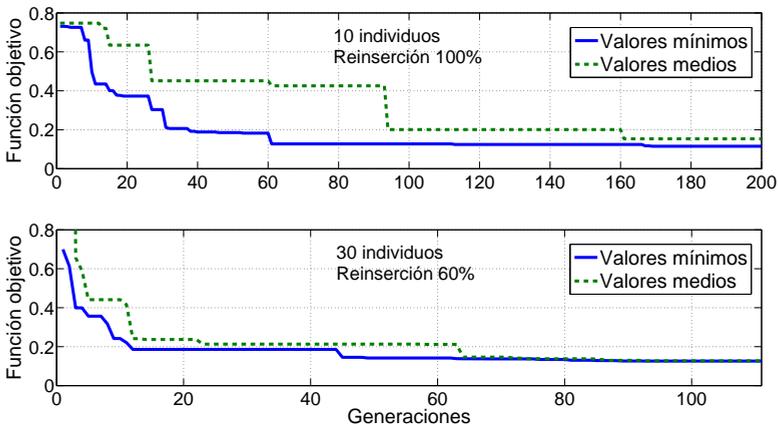


Figura 5.12: Valores mínimos y medios de la función objetivo.

Por otra parte, se han llevado a cabo simulaciones, para 10 y 30 individuos,

con reinserciones inferiores al 100 %, es decir, cuando los padres con mejor aptitud pasan a la próxima generación. Como muestra la tabla 5.1, el valor mínimo de la función objetivo se alcanza para 30 individuos y una tasa de reinserción igual al 60 %, pero son necesarias más evaluaciones con el costo computacional adicional.

TABLA 5.1: MÍNIMOS DE ISEOISRAE PARA 6000 EVALUACIONES

Reinserción (%)	10 individuos	30 individuos	50 individuos
10	0.1619	0.1209	0.1278
20	0.1606	0.1196	0.1257
30	0.1612	0.1180	0.1353
40	0.1663	0.1199	0.1294
50	0.1427	0.1178	0.1319
60	0.1444	0.1123	0.1324
70	0.1363	0.1366	0.1457
80	0.1222	0.1398	0.1478
90	0.1190	0.1287	0.1231
100	0.1152	0.1422	0.1316

En resumen, el interés se centra en alcanzar el valor mínimo de la función objetivo con el mínimo número de evaluaciones y el mínimo número de individuos. Se ha determinado que un tamaño de población adecuado para este problema, y para obtener soluciones aceptables, es una micropoblación de 10 individuos. Para 30 y 50 individuos el valor de la función objetivo es reducido ligeramente, mientras que el costo computacional aumenta drásticamente.

Una característica fundamental en un sistema de control es su estabilidad y obtener un conjunto de ganancias que garanticen un punto de trabajo estable no es una tarea trivial. Para asegurar la estabilidad de las soluciones se inicializan a todos los individuos cerca de un punto estable bien conocido, que consiste en poner todas las ganancias igual a uno ($G_e = 1$, $G_{de} = 1$ y $G_u = 1$). Este punto se usa como primer individuo en la población inicial y los restantes individuos son ligeras modificaciones de este bien conocido punto estable. Los restantes individuos de la población son puestos usando un número pseudo-aleatorio siguiendo una distribución normal con una media de 1 y una desviación estándar de 0.5.

En los experimentos realizados, y después de diversas simulaciones con múltiples variaciones, usando dos métodos de selección (tabla 5.2) y nueve técnicas

de cruce (tabla 5.3)¹, se ha observado que para el proceso de selección el más adecuado es el *sus* y que para la función de cruce la más adecuada varía en función del número de individuos y de la tasa de reinserción, pero ambas no tienen una gran inferencia en el resultado obtenido.

TABLA 5.2: TÉCNICAS DE SELECCIÓN

Selección por ruleta	<i>rws</i>
Muestreo universal estocástico	<i>sus</i>

TABLA 5.3: TÉCNICAS DE CRUCE

Recombinación discreta	<i>recdis</i>
Recombinación intermedia	<i>recint</i>
Recombinación lineal	<i>reclin</i>
Cruce de doble punto	<i>xovdp</i>
Cruce de doble punto con sustituto reducido	<i>xovdprs</i>
Cruce aleatorio	<i>xovsh</i>
Cruce aleatorio con sustituto reducido	<i>xovshrs</i>
Cruce de punto simple	<i>xovsp</i>
Cruce de punto simple con sustituto reducido	<i>xovsprs</i>

La figura 5.13 muestra la convergencia del GA cuando se cambia la metodología de cruce, para los mejores casos y reinserción igual al 100%. Como se muestra, se consigue el mejor resultado para la recombinación lineal, *reclin*, con 10 individuos y el peor resultado, no mostrado en la figura, para la recombinación discreta, *recdis*. El segundo mejor resultado se obtiene para el cruce aleatorio, *xovsh*. El resto de métodos proporcionan, aproximadamente, el mismo resultado. Para 30 individuos, el mejor resultado es para el cruce de doble punto, *xovdp*, y el segundo mejor resultado se obtiene para el cruce de punto simple con sustituto reducido, *xovsprs*. La figura 5.14 muestra los valores mínimos y medios con el método *reclin* para 10 individuos y el método *xovdp* para 30 individuos. En la tabla 5.4 aparecen los valores mínimos obtenidos para las diferentes metodologías de cruce.

La figura 5.15 ilustra los valores de la función objetivo con diferentes probabilidades de cruce para el método *reclin* con 10 individuos y para el método

¹Una descripción detallada de los diferentes métodos y técnicas se encuentra en la referencia [165].

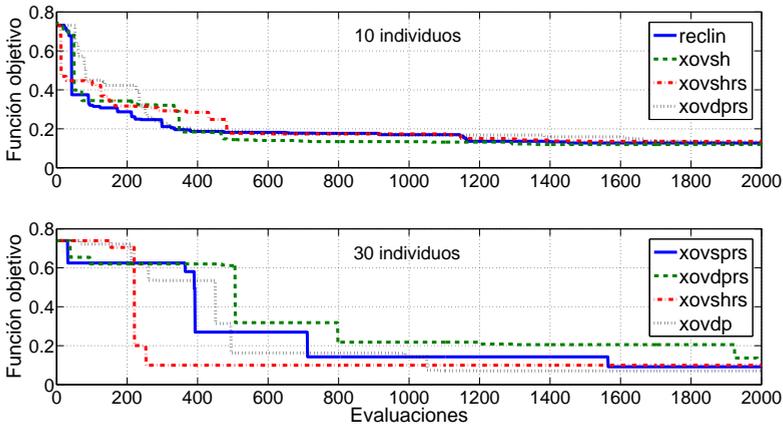


Figura 5.13: Función objetivo al variar la metodología de cruce.

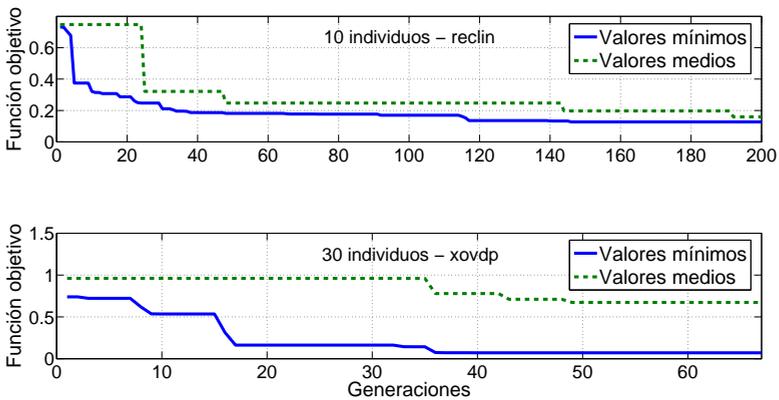


Figura 5.14: Valores mínimos y medios para *reclin* y *xovdp*.

xovdp con 30 individuos. Para las probabilidades de cruce entre 0.6 y 0.8, los valores de la función objetivo tienden al valor mínimo. La figura 5.16 ilustra las gráficas para los valores mínimos y medios con probabilidad igual al 70 % para el método *reclin* con 10 individuos y probabilidad igual al 80 % para el método *xovdp* con 30 individuos. En la tabla 5.5 aparecen los valores mínimos obtenidos para diferentes probabilidades de cruce.

Por otro lado, el esquema de mutación afecta directamente a la solución óptima. Los mejores resultados se obtienen cuando la mutación varía entre 0.4 (40 %) y 0.6 (60 %) para el algoritmo de Breeder [166]. La figura 5.17

TABLA 5.4: MÍNIMOS DE ISEOISRAE VARIANDO EL MÉTODO DE CRUCE

Metodología de cruce	10 individuos (100 % de reinserción)	30 individuos (100 % de reinserción)
<i>recdis</i>	0.1519	0.1640
<i>recint</i>	0.1308	0.4573
<i>reclín</i>	0.1167	0.2185
<i>xovdp</i>	0.1502	0.0706
<i>xovdprs</i>	0.1206	0.1164
<i>xovsh</i>	0.1192	0.0998
<i>xovshrs</i>	0.1249	0.0997
<i>xovsp</i>	0.1325	0.1275
<i>xovsprs</i>	0.1419	0.0827

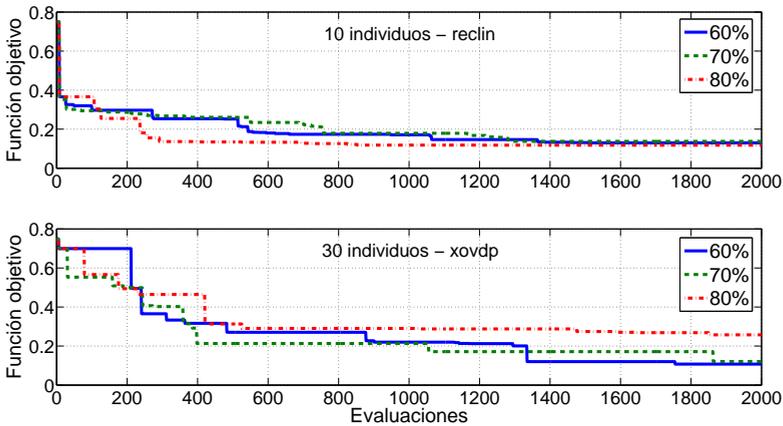


Figura 5.15: Función objetivo con diferentes probabilidades de cruce.

muestra los valores mínimos para el 40 %, 50 % y 60 % de mutación con 10 y 30 individuos. En la figura 5.18 se representan los valores mínimos y valores medios para el 40 % con 10 individuos y el 50 % con 30 individuos. En la tabla 5.6 se resumen los valores mínimos obtenidos para diferentes probabilidades de mutación.

5.4.5. Resultados obtenidos

En la figura 5.19 se puede ver la respuesta del sistema controlado cuando se produce una perturbación de amplitud -0.5 en el instante 10 segundos. Como

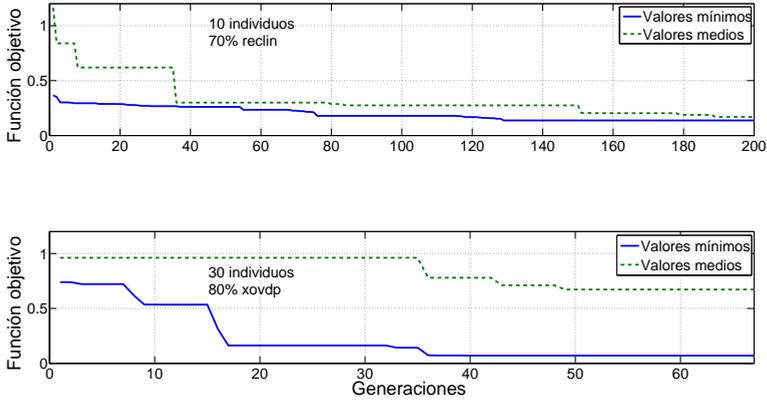


Figura 5.16: Valores mínimos y medios al variar la probabilidad de cruce.

TABLA 5.5: MÍNIMOS DE ISEOISRAE VARIANDO LA PROBABILIDAD DE CRUCE

Probabilidad de cruce (%)	10 individuos (<i>reclin</i>)	30 individuos (<i>xovdp</i>)
10	0.1250	0.1221
20	0.1211	0.1250
30	0.1241	0.1232
40	0.1315	0.1139
50	0.1254	0.1003
60	0.1204	0.0953
70	0.1169	0.0960
80	0.1178	0.0710
90	0.1230	0.1040
100	0.1326	0.1294

se muestra, la función objetivo ISE e IAE generan sobreimpulsos para rechazar la perturbación, mientras que con la función objetivo ISEOISRAE propuesta (5.1) no se generan sobreimpulsos.

Como se mencionó en la sección 5.2, la función objetivo propuesta se enfoca en rechazar los sobreimpulsos. Sin embargo, las funciones ISE e IAE ajustan más el comportamiento deseado a la salida del modelo de rechazo en el comienzo de la perturbación.

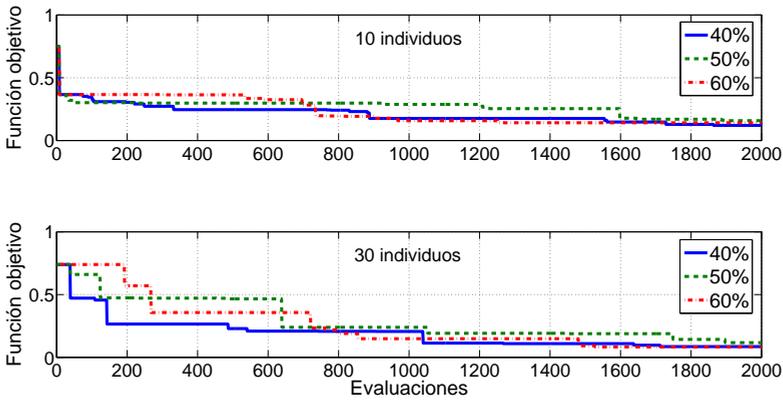


Figura 5.17: Mínimos para diferentes probabilidades de mutación.

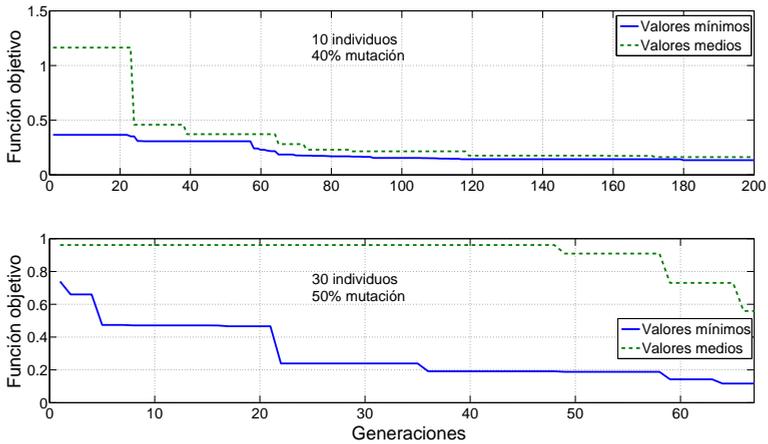


Figura 5.18: Valores mínimos y medios al variar la mutación.

La figura 5.20 muestra el comportamiento del sistema cuando se aplican diferentes perturbaciones en diferentes instantes. La salida del sistema sigue el comportamiento deseado, sin presentar sobreimpulsos cuando aparecen las perturbaciones de carga. Las ganancias del PI borroso se obtuvieron mediante entrenamiento con GA.

TABLA 5.6: MÍNIMOS DE ISEOISRAE VARIANDO LA PROBABILIDAD DE MUTACIÓN

Probabilidad de mutación (%)	10 individuos (<i>reclin</i>)	30 individuos (<i>xovdp</i>)
10	0.1719	0.1430
20	0.1499	0.1257
30	0.1417	0.0905
40	0.1178	0.0855
50	0.1405	0.0782
60	0.1410	0.0832
70	0.1336	0.0959
80	0.1622	0.1152
90	0.1867	0.1439
100	0.1709	0.1408

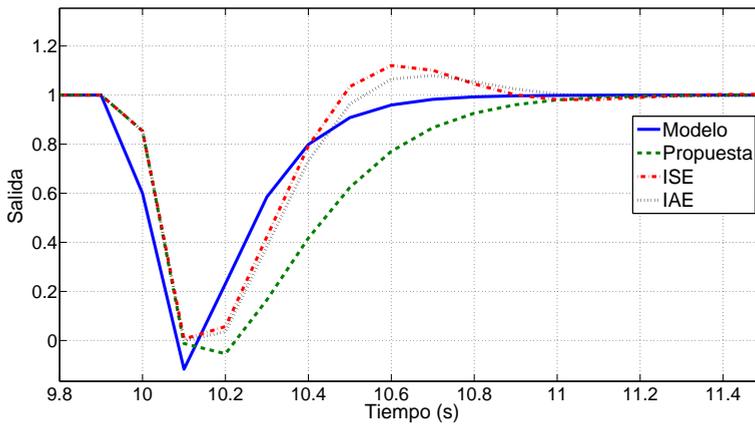


Figura 5.19: Rechazo de la perturbación para diversas funciones objetivos.

Sección 5.5

Sistema motor-generador

En [167] se realiza el control de la posición angular de un sistema motor-generador cuya función de transferencia está dada en (5.8), siendo θ la posición angular, $V_a(s)$ la tensión de armadura aplicada, K la constante de ganancia, T_m la constante de tiempo del motor y T_g la constante de tiempo del generador.

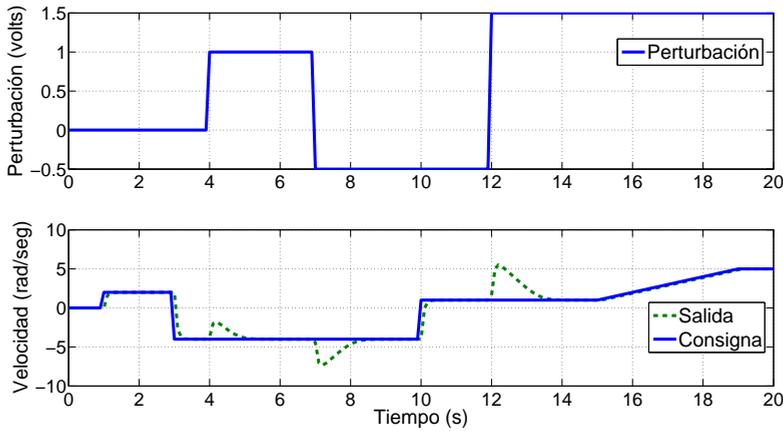


Figura 5.20: Respuesta final del sistema.

$$G_P(s) = \frac{\theta(s)}{V_a(s)} = \frac{K}{s \cdot (T_m \cdot s + 1) \cdot (T_g \cdot s + 1)} \quad (5.8)$$

Los cambios en la carga se tratan como una perturbación externa $M(s)$, de forma que la posición angular tiene la forma (5.9) cuando $K = 1$, $T_m = 1$ y $T_g = 0.25$.

$$\theta(s) = \frac{1}{s \cdot (s + 1) \cdot (0.25 \cdot s + 1)} \cdot (V_a(s) + M(s)) \quad (5.9)$$

Este apartado se centrará en el rechazo de la perturbación de carga siguiendo el procedimiento presentado anteriormente².

En [167] se diseña un control robusto por el método de diagrama de coeficientes —*Coefficient Diagram Method* (CDM)— [168, 169] y en la figura 5.21 se muestra la respuesta obtenida cuando se produce una perturbación en forma de escalón de valores $M = 1$, $M = 2$ y $M = 3$ en el instante $t = 10$ s, cuando la salida estaba en la posición de $\theta = 1$ rad.

La respuesta obtenida con la propuesta presentada, una vez diseñado el PI borroso, se muestra en la figura 5.22. Como se observa, el rechazo presenta respuesta sin sobreimpulsos. En la tabla 5.7 se presentan las medidas del tiempo de establecimiento (t_e) y del desplazamiento angular (M_p) para el método CDM y con el PI borroso. La propuesta con el PI borroso presenta un

²Para el diseño del controlador inverso se sigue el mismo procedimiento que el que se presentará en el apartado 6.2, ya que la función de transferencia (5.8) presenta la misma forma que la función (6.9).

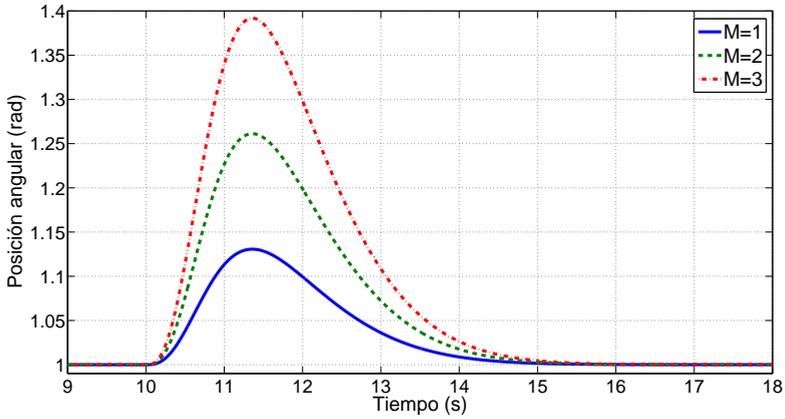


Figura 5.21: Respuesta ante la perturbación por el método CDM.

menor desplazamiento angular para los tres casos y únicamente para $M = 3$ el método CDM de [167] ofrece mejor tiempo de establecimiento.

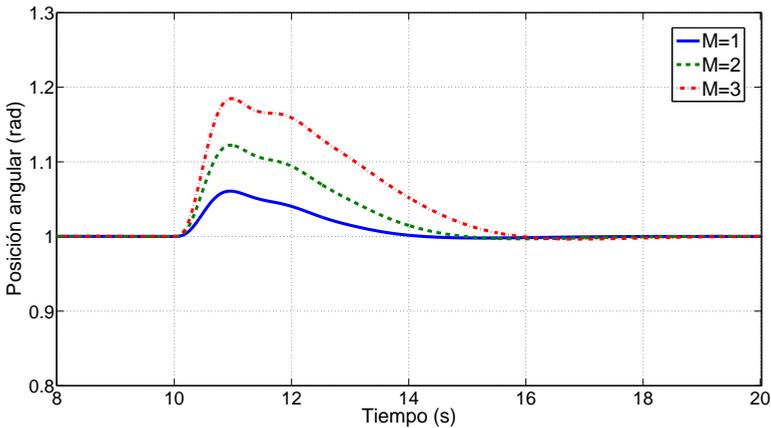


Figura 5.22: Respuesta ante la perturbación con el PI borroso.

Sección 5.6

Aportaciones

En este capítulo se ha demostrado que un PI borroso bien sintonizado es capaz de compensar perturbaciones de carga en forma de escalón, a la vez que

TABLA 5.7: TIEMPO DE ESTABLECIMIENTO Y MÁXIMO SOBREPULSO

	PI borroso	Método CDM
t_e (s)	2.76 (M=1)	3.30 (M=1)
	3.80 (M=2)	3.80 (M=2)
	4.80 (M=3)	4.10 (M=3)
M_p (rad)	0.06 (M=1)	0.13 (M=1)
	0.12 (M=2)	0.26 (M=2)
	0.18 (M=3)	0.39 (M=3)

minimiza los efectos producidos por los errores de modelado.

Las aportaciones realizadas respecto al PI borroso han sido:

1. Se han definido los parámetros de la lógica borrosa del PI borroso como una extensión del capítulo anterior.
2. El número de conjuntos borrosos para la entrada *error* se ha establecido en cinco (*Negativo Grande, Negativo, Cero, Positivo y Positivo Grande*) y en tres para la entrada *derivada del error* y para la salida (*Negativo, Cero y Positivo*).
3. El tipo *singleton* para el conjunto borroso *Cero* y el tipo *gaussiano* para el resto de conjuntos, son los que mejores resultados han ofrecido.
4. Se fijan en once las reglas para cubrir de forma adecuada las diferentes situaciones que se pueden dar. Esto supone una ampliación en dos reglas con respecto al PI borroso para compensación de errores de modelado.
5. Los métodos usados en las distintas operaciones, llevadas a cabo por la lógica borrosa, han sido: el producto para la operación *and*, el mínimo para la implicación *then*, la suma para la agregación y el COG para la desborrosificación. De esta forma, hay una coincidencia con los métodos encontrados en el PI borroso del capítulo anterior.

Respecto al algoritmo genético encargado de sintonizar de forma adecuada las ganancias del PI borroso:

1. Se ha creado una nueva función objetivo ISEOISRAE. Esta aportación ha sido fundamental para conseguir los mejores resultados.

2. Para ciertas aplicaciones se necesita un rechazo evitando sobreimpulsos de la variable controlada, por lo que se fija el modelo de rechazo de perturbación para que responda de esta forma.
3. Tras numerosas simulaciones, se concluye que una micropoblación de diez individuos con una reinscripción del 100 % es suficiente para conseguir resultados excelentes. Con esto se minimiza el coste computacional con respecto a tamaños de poblaciones mayores.
4. También se han establecido las metodologías más idóneas para las distintas fases del algoritmo genético. Probando con diferentes técnicas y diferentes tasas, se han establecido los parámetros más adecuados del algoritmo genético para la función encomendada.

Las aportaciones presentadas en este capítulo han sido publicadas en:

Betancor, C. S., Sosa-González, C. J., Montiel, J. A., Vega, A., "PI Fuzzy System Gains Tuning Using Genetic Algorithms", EUROGEN 2011, Conference Proceedings, pp 238-243, Capua, Italy, 2011.

Betancor, C. S., Sosa-González, C. J., Montiel, J. A., Vega, A., "Gains Tuning of a PI-Fuzzy controller by genetic algorithms", Engineering Computations, Emerald, Vol. 31, Issue 6, pp 1074-1097, 2014.

Control de sistemas lineales

En el capítulo anterior se amplió el diseño del controlador borroso para que, además de compensar las incertidumbres en el modelado de la planta, fuera capaz de mitigar los efectos producidos por las perturbaciones de carga que pueden aparecer en el sistema.

Este capítulo se centrará en el diseño del controlador inverso para plantas lineales de segundo y tercer orden que poseen efecto integrativo. De esta forma, este capítulo es una extensión del capítulo 2, donde se explicó el proceso a seguir para diseñar el controlador inverso para una planta de primer orden. Como se mostrará, no será necesario incluir el PI borroso en el caso de que se produzcan errores de modelado. Se comienza con una planta LTI de segundo orden, realizando la identificación para obtener el modelo matemático y generar la red neuronal que implementa al controlador inverso. Posteriormente se analiza el comportamiento del sistema controlado y se comparan las respuestas obtenidas con otras técnicas de control. Se investiga como mejorar la señal de control y finalmente se analiza la estabilidad y robustez del control diseñado. Se sigue con una planta LTI de tercer orden, que tiene un equivalente discreto de fase no mínima, por lo que el controlador inverso sería inestable. Esto conlleva a discutir alternativas en la identificación que hagan que con el modelo de la planta obtenido se pueda generar un controlador inverso aproximado estable. Se estudiarán los parámetros que influyen para crear dicho inverso aproximado, de forma que el control final ofrezca mejores prestaciones que otras técnicas planteadas en la literatura. Además, se analizará la estabilidad y robustez del sistema resultante.

En el siguiente capítulo se abordan los sistemas que poseen no linealidades

continuas. Concretamente se estudiarán los casos de no linealidades suaves y de no linealidades duras.

El capítulo está organizado como sigue. En la sección 6.1 se diseña el sistema de control para una planta LTI de segundo orden. Para ello, en primer lugar, se procede a la descripción e identificación de la planta y a partir del modelo obtenido se diseña la red neuronal que actuará como controlador inverso. Se continúa comparando el comportamiento del sistema controlado con otras técnicas planteadas en la literatura. De los resultados obtenidos se analiza la señal de control y se proponen soluciones para su mejora. Seguidamente, se estudia la estabilidad y robustez del sistema ante errores de modelado de la planta. En la sección 6.2, una planta LTI de tercer orden es el objeto de control. Como en el caso anterior, se realiza la descripción e identificación para generar un modelo Takagi-Sugeno de la planta y posteriormente se diseña y se analiza el comportamiento del sistema controlado, comparándose con otras alternativas de diseño propuestas en la literatura. La estabilidad y la robustez ante errores de modelado de la planta también son analizadas. Para finalizar, en la sección 6.3, se describen las aportaciones de este capítulo.

Sección 6.1

Planta LTI de segundo orden

6.1.1. Descripción e identificación de la planta

Sea la planta LTI de segundo orden definida en (6.1).

$$G_P(s) = \frac{Y(s)}{U(s)} = \frac{4}{s \cdot (s + 0.5)} \quad (6.1)$$

Para realizar la identificación de la planta bastará con aplicar señales en forma de escalón (figura 6.1), siendo suficiente un clúster dado el carácter lineal de la planta. Los parámetros fijados en el algoritmo de agrupamiento borroso se reflejan en la tabla 6.1.

En el proceso de validación se obtiene un $VAF = 100\%$ (figura 6.2) y el modelo borroso obtenido presenta un única regla cuyo consecuente se presenta en (6.2).

$$y(k) = \left. \begin{aligned} &+ 1.95y(k-1) - 0.951y(k-2) + 0.0197u(k-1) + \\ &+ 0.0193u(k-2) - 3.19 \cdot 10^{-12} \end{aligned} \right\} \quad (6.2)$$

TABLA 6.1: PARÁMETROS PARA LA IDENTIFICACIÓN DE LA PLANTA DE SEGUNDO ORDEN

Número de clústers	$.c = 1$
Coefficiente de borrosificación	$.m = 2$
Tolerancia de terminación	$.tol = 0.001$
Semilla para la inicialización	$.seed = 1$
Tipo de los antecedentes	$.ante = 2$
Estimación del consecuente	$.cons = 1$
Número de retardos de la salida	$.Ny = 2$
Número de retardos de la entrada	$.Nu = 2$
Número de retardos de transporte	$.Nd = 1$

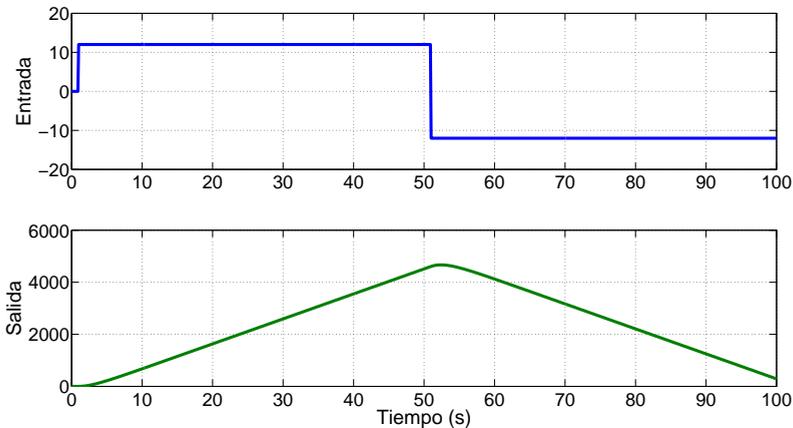


Figura 6.1: Señales para la identificación.

Si se parte de la planta continua original (6.1) y se discretiza, con periodo de muestreo $T = 0.1$ s, se obtiene el modelo equivalente discreto (6.3). La salida de la planta (6.4) se calcula realizando la transformada inversa Z de $Y(z)$ en (6.3).

$$G_P(z) = \frac{Y(z)}{U(z)} = \frac{0.0197 \cdot z + 0.0193}{z^2 - 1.9512 \cdot z + 0.9512} \quad (6.3)$$

$$\left. \begin{aligned} y(k) = & + 1.9512y(k-1) - 0.9512y(k-2) + \\ & + 0.0197u(k-1) + 0.0193u(k-2) \end{aligned} \right\} \quad (6.4)$$

El consecuente (6.2) del modelo obtenido mediante identificación, despreciando el término independiente, es igual a la la salida de la planta (6.4), con

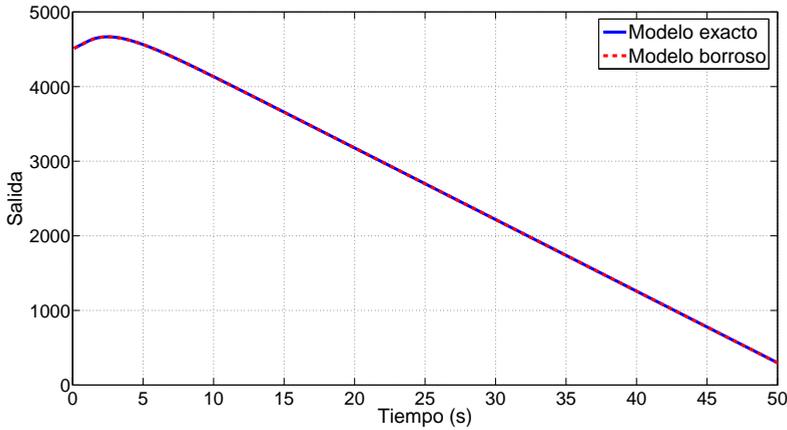


Figura 6.2: Resultado de la validación.

lo que se ha comprobado que la identificación ha sido exacta.

6.1.2. Generación de la red neuronal

Fijando un tiempo de establecimiento $t_e = 1$ s, con periodo de muestreo $T = 0.1$ s, se obtiene la ecuación en diferencias, dada en (6.5), que describe al modelo de referencia.

$$y_r(k) = 0.6762y_r(k-1) + 0.3238r(k-1) \quad (6.5)$$

Igualando (6.4) a (6.5) se obtiene la señal de control necesaria $u(k)$ (6.6). En este caso se requiere una red neuronal recurrente.

$$u(k) = + \left. \begin{aligned} & \frac{(0.6762 - 1.95)}{0.0197}y(k) + \frac{0.951}{0.0197}y(k-1) - \\ & - \frac{0.0193}{0.0197}u(k-1) + \frac{0.3238}{0.0197}r(k) \end{aligned} \right\} \quad (6.6)$$

6.1.3. Comportamiento del sistema controlado

En [170] se realiza el diseño de dos sistemas de control neuroborrosos — *NEural Fuzzy CONtrol* (NEFCON)— y comparan las respuestas obtenidas con un método analítico de diseño en el que se obtiene un control tipo *lead-lag* (adelanto-atraso de fase) [153]. En la figura 6.3 se muestra la comparativa de las tres salidas que se obtienen, con los tres controladores, para una consigna en escalón de amplitud 1 aplicada en el instante $t = 0$ s.

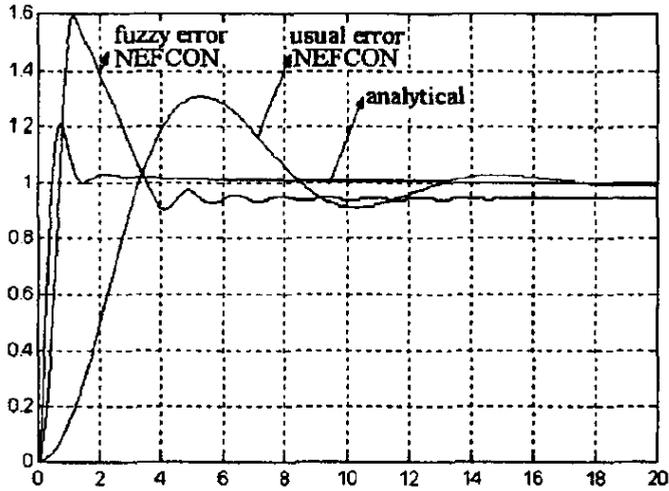


Figura 6.3: Respuestas con dos controles neuroborrosos y un *lead-lag* (Fuente: [170]).

A continuación se realizan simulaciones para estudiar el comportamiento del sistema propuesto. Aplicando una consigna en forma de escalón unitario, en la figura 6.4 se observa que la respuesta del sistema controlado coincide exactamente con la salida deseada del modelo de referencia, tanto en el transitorio como en régimen permanente, por lo que en principio se han obtenido los resultados deseados. Comparando estos resultados con los obtenidos en [170], se observa claramente la superioridad del controlador diseñado. Se puede fijar el tiempo de establecimiento del modelo de referencia que se quiera a costa de aumentar la amplitud de la señal de control.

Aunque en [170] no se muestran las señales de control obtenidas con las estrategias NEFCON, en la figura 6.5 se muestra la señal de control que genera el controlador *lead-lag* en comparación con la del controlador inverso de la propuesta presentada. Está claro que la señal de control obtenida con el *lead-lag* es mucho mejor, puesto que apenas tiene oscilaciones. Sin embargo, la del controlador inverso no es muy buena para aplicar a una planta, ya que cada 0.1 s cambia de un valor positivo a uno negativo suponiendo un gran esfuerzo y además tarda tiempo en tender al valor cero. Aumentando el tiempo de establecimiento impuesto, se disminuiría la amplitud de la señal de control, pero no variarían las oscilaciones que se presentan.

Analizando el esquema base implementado (figura 6.6), se deduce que las oscilaciones de la señal de control $u(k)$ las produce el polo del bloque *Inverso de la Planta* y situado en $z = -0.9835$.

Para aliviar esta señal de control, se plantea el que la señal de salida no

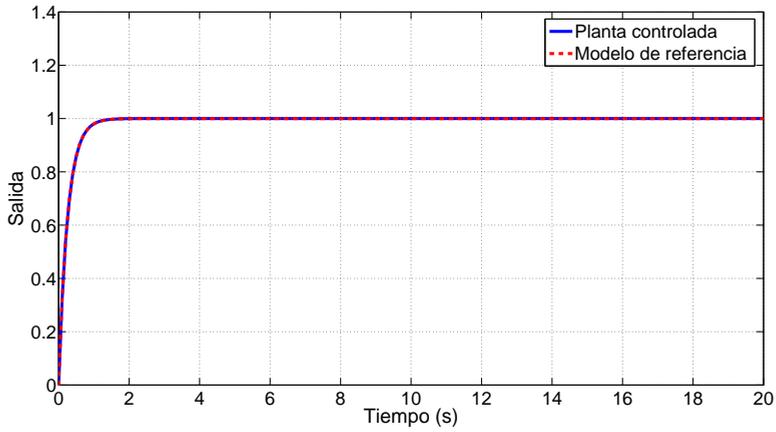


Figura 6.4: Señal de salida para el sistema controlado.

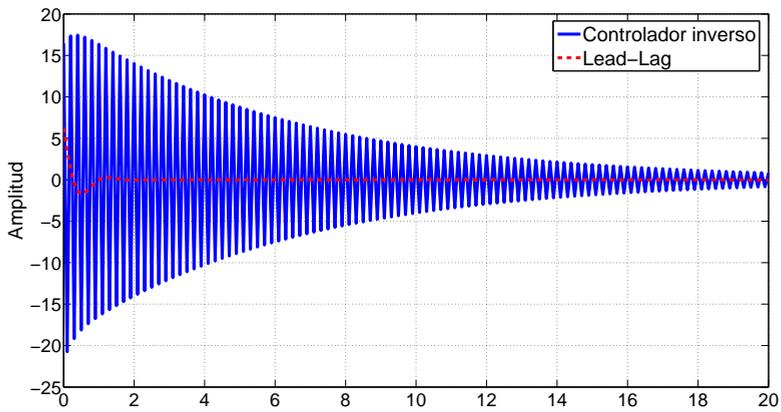


Figura 6.5: Señal de control que se aplica a la planta.

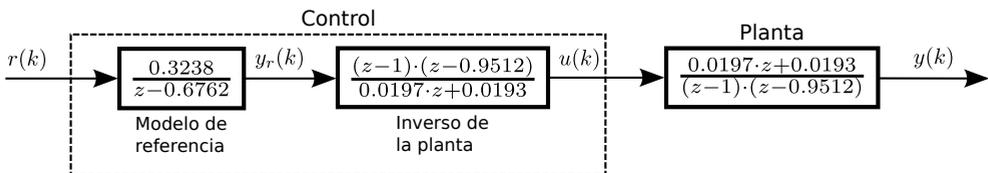


Figura 6.6: Estructura base del sistema de control propuesto.

siga exactamente a la salida del modelo de referencia sino a una aproximación de ésta, o sea, exigir menos exactitud en la respuesta del sistema controlado

intentando eliminar del modelo inverso de la planta el polo en $z = -0.9797$. Para ello será necesario eliminar el término $u(k - 2)$ del modelo borroso, o lo que es lo mismo, realizar una identificación aproximada con un orden del numerador igual a 1 ($.Nu = 1$). Con esta modificación, el modelo obtenido (6.7) presenta un $VAF = 99.9948 \%$, lo cual indica que es un buen modelo.

$$y(k) = + 1.95y(k - 1) - 0.954y(k - 2) + \left. \begin{array}{l} \\ + 0.0371u(k - 1) + 8.69 \cdot 10^{-12} \end{array} \right\} \quad (6.7)$$

Si se compara el modelo exacto (6.2) con el modelo aproximado (6.7), se ve que los coeficientes de $y(k - 1)$ e $y(k - 2)$ varían ligeramente y que el coeficiente de $u(k - 1)$ en (6.7) ha aumentado, lo que supondrá que la señal de control necesaria presentará menor amplitud.

Ahora se diseñan dos controladores para dos modelos de referencia, uno con tiempo de establecimiento $t_e = 1 s$ y el otro para $t_e = 2 s$. En la figura 6.7 se observan las respuestas obtenidas.

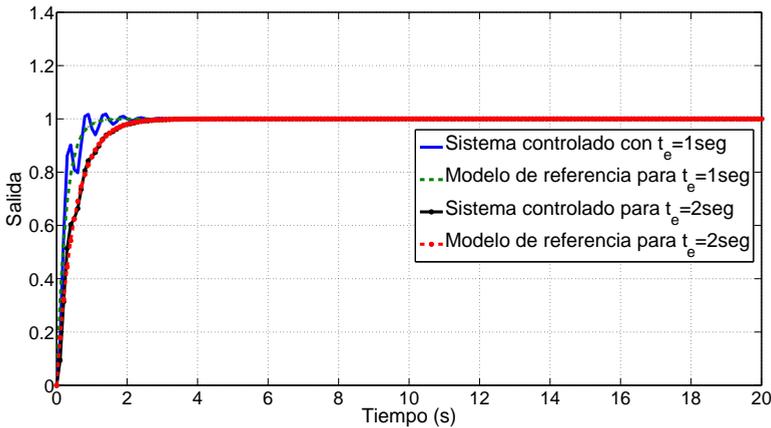


Figura 6.7: Salida del sistema controlado para t_e de 1 y 2 segundos.

Para $t_e = 1 s$ el sistema presenta un ligero error de seguimiento con respecto a la salida del modelo de referencia, pero en cambio, la señal de control (figura 6.8) exige menos esfuerzo a la planta y se estabiliza aproximadamente en el mismo tiempo que con el *lead-lag*, aunque presenta una mayor amplitud debida al tiempo de establecimiento requerido. Para $t_e = 2 s$ la salida del sistema sigue con bastante exactitud a la salida del modelo de referencia, presentando la señal de control casi la misma amplitud que con el *lead-lag*, pero con la ventaja de no presentar sobreimpulso la salida.

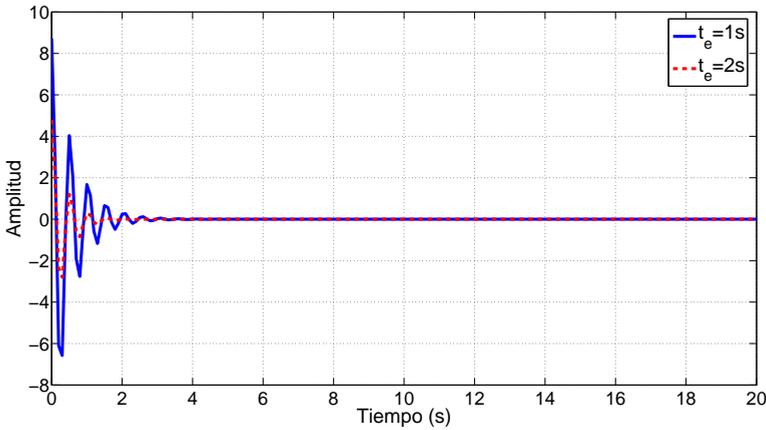


Figura 6.8: Señal de control que se aplica a la planta.

Se puede concluir que se obtienen mejores resultados que los diseños realizados en [170]. En el diseño propuesto el sobreimpulso es cero, mientras que con el *lead-lag* es del 20 %. Además, el tiempo de establecimiento se puede ajustar dependiendo de la amplitud máxima que admita la señal de control.

6.1.4. Estabilidad y robustez ante errores de modelado

A continuación se estudia la estabilidad y robustez del sistema frente a los errores de modelado de la planta. Si se supone una variación máxima en los parámetros de la planta del $\pm 25\%$ se obtienen las plantas presentadas en (6.8).

$$\left. \begin{aligned} G_{P1} &= \frac{3}{s^2 + 0.375 \cdot s} & G_{P2} &= \frac{3}{s^2 + 0.625 \cdot s} \\ G_{P3} &= \frac{5}{s^2 + 0.375 \cdot s} & G_{P4} &= \frac{5}{s^2 + 0.625 \cdot s} \end{aligned} \right\} \quad (6.8)$$

Eligiendo el modelo aproximado (6.7) y haciendo el cálculo con un tiempo de establecimiento $t_e = 2\text{ s}$, se obtienen las respuestas de la figura 6.9. Como se puede ver, los modelos más afectados son G_{P3} y G_{P4} . En ningún caso se produce sobreimpulso sobre el valor final y los tiempos de establecimiento son $t_{e1} = 1.93\text{ s}$, $t_{e2} = 2\text{ s}$, $t_{e3} = 2.054\text{ s}$ y $t_{e4} = 2.058\text{ s}$, para G_{P1} , G_{P2} , G_{P3} y G_{P4} respectivamente. Con esto queda demostrada la estabilidad y robustez del controlador diseñado cuando hay errores de modelado de la planta.

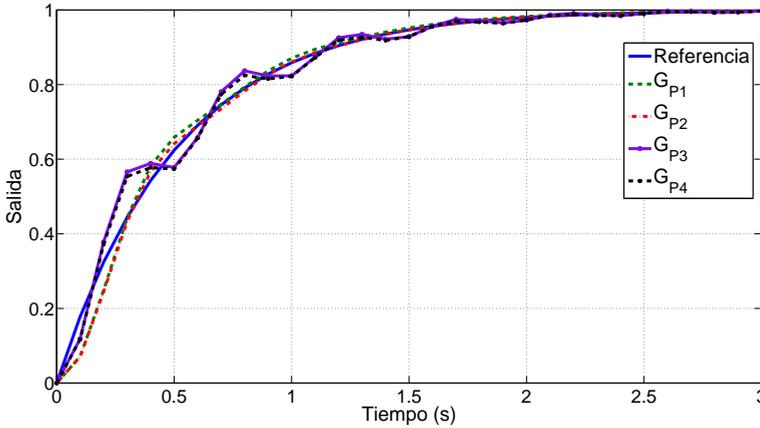


Figura 6.9: Salidas para diferentes modelos de planta.

Sección 6.2

Planta LTI de tercer orden

6.2.1. Descripción e identificación de la planta

Ahora se elige la planta de tercer orden descrita en (6.9).

$$G_P(s) = \frac{Y(s)}{U(s)} = \frac{1}{s \cdot (s + 1) \cdot (s + 5)} \tag{6.9}$$

La figura 6.10 muestra las señales de entrada y salida usadas en el proceso de identificación, con un periodo de muestreo $T = 0.2 s$. Los parámetros elegidos en el algoritmo de identificación se indican en la tabla 6.2.

En el proceso de validación se obtiene un $VAF = 100 \%$ y el modelo obtenido presenta una única regla con el consecuente dado en (6.10).

$$\left. \begin{aligned} y(k) = & + 2.19y(k - 1) - 1.49y(k - 2) + 0.302y(k - 3) + \\ & + 9.99 \cdot 10^{-4}u(k - 1) + 0.00304u(k - 2) + \\ & + 5.42 \cdot 10^{-4}u(k - 3) + 6.67 \cdot 10^{-16} \end{aligned} \right\} \tag{6.10}$$

Si se parte de la planta continua original (6.9) y se discretiza, con periodo de muestreo de $T = 0.2 s$, se obtiene el modelo equivalente discreto (6.11) de fase no mínima, y a partir de este modelo la ecuación en diferencias (6.12) que define a la planta.

TABLA 6.2: PARÁMETROS PARA LA IDENTIFICACIÓN DE LA PLANTA DE TERCER ORDEN

Número de clústers	$.c = 1$
Coefficiente de borrosificación	$.m = 2$
Tolerancia de terminación	$.tol = 0.001$
Semilla para la inicialización	$.seed = 1$
Tipo de los antecedentes	$.ante = 2$
Estimación del consecuente	$.cons = 1$
Número de retardos de la salida	$.Ny = 3$
Número de retardos de la entrada	$.Nu = 3$
Número de retardos de transporte	$.Nd = 1$

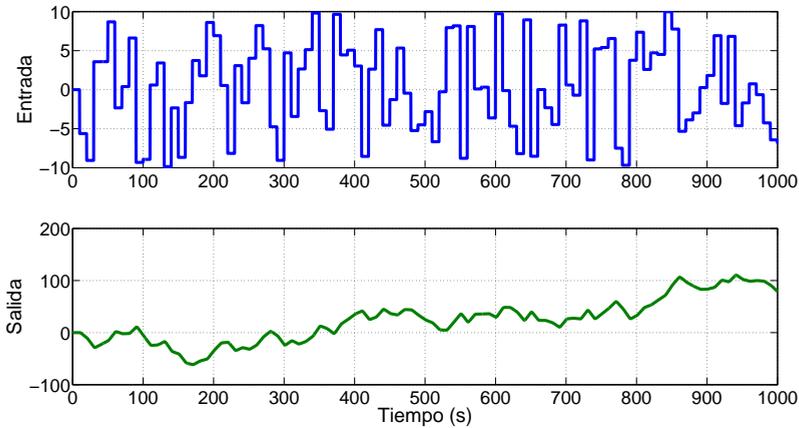


Figura 6.10: Señales para la identificación.

$$G_P(z) = \frac{Y(z)}{U(z)} = \frac{0.001004 \cdot z^2 + 0.003027 \cdot z + 0.000552}{z^3 - 2.186610 \cdot z^2 + 1.487804 \cdot z - 0.301194} \quad (6.11)$$

$$y(k) = \left. \begin{aligned} &+ 2.1866y(k-1) - 1.4878y(k-2) + 0.3012y(k-3) + \\ &+ 0.0010u(k-1) + 0.0030u(k-2) + 0.00055u(k-3) \end{aligned} \right\} \quad (6.12)$$

El consecuente (6.10) del modelo obtenido mediante identificación es igual a la ecuación en diferencias (6.12) que describe a la planta, por lo que la identificación ha sido correcta.

Analizando el esquema de la figura 6.11, se deduce que el sistema será inestable puesto que los polos del inverso de la planta estarán situados en

$z = -0.1949$ y $z = -2.8210$. Así pues, el bloque *Inverso de la Planta* no podrá ser exacto para que no aparezcan polos fuera del círculo unidad del plano Z .

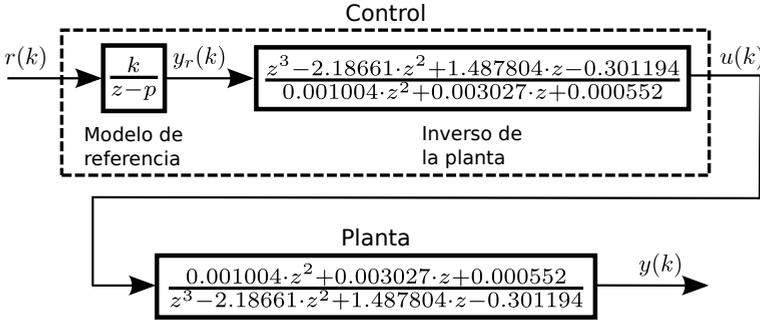


Figura 6.11: Sistema de control propuesto en cadena abierta.

Una solución a este problema es la misma que la planteada en el apartado 6.1.3, cuando se quiso mejorar la forma de la señal de control, es decir, no hacer una identificación exacta de la planta y así obtener un inverso aproximado. Fijando $.Ny = 3$ y $.Nu = 2$ se obtiene un $VAF = 99.9999\%$ y el modelo con el consecuente (6.13).

$$y(k) = \left. \begin{aligned} &+ 2.56y(k-1) - 2.15y(k-2) + 0.592y(k-3) + \\ &+ 0.000144u(k-1) + 0.000529u(k-2) + 1.25 \cdot 10^{-6} \end{aligned} \right\} \quad (6.13)$$

La ecuación (6.13) da lugar a la función de transferencia del modelo de la planta (6.14), donde los ceros están en $z = 0$ y $z = -3.6736$, y por lo tanto daría lugar a un sistema inestable ya que estos ceros serán los polos del inverso.

$$G_{P_m}(z) = \frac{Y(z)}{U(z)} = \frac{0.000144 \cdot z^2 + 0.000529 \cdot z}{z^3 - 2.56 \cdot z^2 + 2.15 \cdot z - 0.592} \quad (6.14)$$

Con $.Ny = 3$ y $.Nu = 1$ se obtiene un $VAF = 99.9967\%$ y el modelo con el consecuente (6.15), que dará lugar a la función de transferencia del modelo de la planta (6.16), con los ceros posicionados en $z = 0$, dando lugar a un sistema estable.

$$y(k) = \left. \begin{aligned} &+ 2.71y(k-1) - 2.43y(k-2) + 0.728y(k-3) + \\ &+ 0.000439u(k-1) + 5.61 \cdot 10^{-6} \end{aligned} \right\} \quad (6.15)$$

$$G_P(z) = \frac{Y(z)}{U(z)} = \frac{0.000439 \cdot z^2}{z^3 - 2.71 \cdot z^2 + 2.43 \cdot z - 0.728} \quad (6.16)$$

Para $.Ny = 2$ y $.Nu = 1$ se obtiene un $VAF = 99.9651\%$ y el modelo con el consecuente (6.17), el cual tiene un cero en $z = 0$ y dará lugar a un sistema estable.

$$\left. \begin{aligned} y(k) = &+ 1.94y(k-1) - 0.935y(k-2) + \\ &+ 0.00132u(k-1) - 6.12 \cdot 10^{-5} \end{aligned} \right\} \quad (6.17)$$

Con $.Ny = 2$ y $.Nu = 2$ se obtiene un $VAF = 99.9874\%$ pero generaría un sistema inestable, puesto que el cero de la planta identificada se situaría en $z = -9.0604$.

En general, con $.Nu = 1$ los ceros que aparezcan en el sistema identificado, que serán los polos del inverso, estarán en $z = 0$, dando lugar a un sistema estable.

6.2.2. Comportamiento del sistema controlado

En [170] se realiza una comparativa de las respuestas obtenidas con un PID, sintonizado mediante las reglas de Ziegler-Nichols, y dos controladores neuroborrosos NEFCON. Las respuestas de estos tres controladores son mostradas en la figura 6.12.

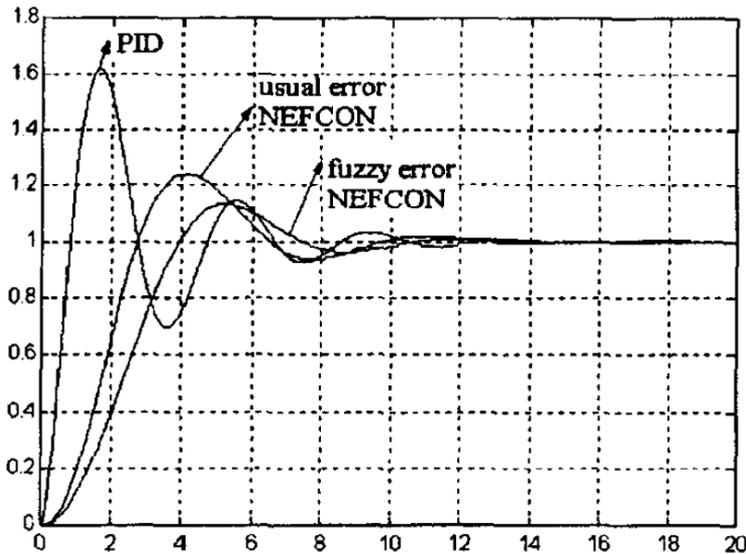


Figura 6.12: Respuestas con dos controles neuroborrosos y un PID (Fuente: [170]).

Procediendo del mismo modo que en el apartado 6.1.2 se calcula la expresión de la señal de control. Fijando el modelo de referencia con un tiempo de establecimiento de 7 s y con el consecuente (6.15) se obtiene una red neuronal no

recurrente. El comportamiento del sistema controlado ha resultado ser inestable, debido a que se ha partido de un modelo borroso aproximado, y la salida de éste no es igual a la de la planta real. Para solucionar este problema, una alternativa es disminuir el periodo de muestreo en el control del sistema. Si se usa un periodo de muestreo en el control $T = 0.1 \text{ s}$, la mitad del usado en la identificación, el sistema sí se hace estable y responde de manera satisfactoria, comportándose de forma muy superior a los diseños realizados en [170], donde el mejor tiempo de establecimiento para *fuzzy error* NEFCON está sobre los 10 s y con un sobreimpulso superior al 10% . Para el diseño propuesto se ha conseguido un tiempo de establecimiento de 3.4 s y sobreimpulso cero.

La figura 6.13 muestra la respuesta para $.Ny=3$, $.Nu=1$ y para $.Ny=2$ y $.Nu=1$. Aunque en la figura no se aprecia claramente, se produce un ligero error en régimen permanente para $.Ny = 2$ y $.Nu = 1$, por lo que la elección final sería para $.Ny = 3$ y $.Nu = 1$, donde se consigue un seguimiento de la referencia sin error en régimen permanente.

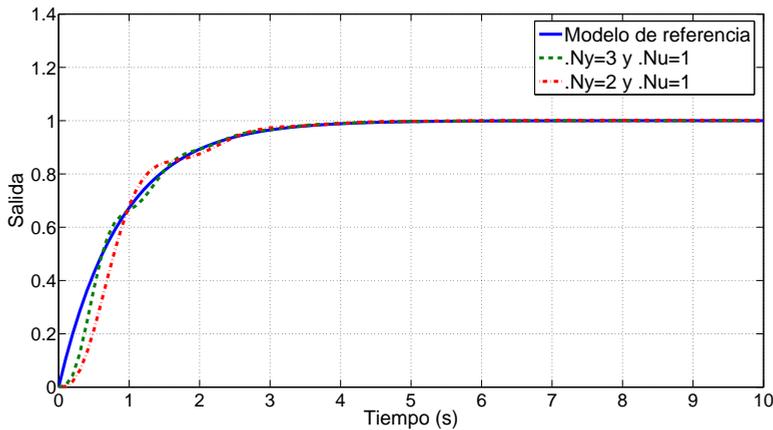


Figura 6.13: Respuestas del sistema controlado.

Ahora se analiza la estabilidad y robustez del sistema frente a variaciones en el modelado de la planta. La planta tiene polos en $s = 0$, $s = -1$ y $s = -5$. Se suponen dos casos G_{P1} y G_{P2} dados en (6.18). En G_{P1} los polos se desplazan a $s = 0$, $s = -5$ y $s = -7$. En G_{P2} los polos se sitúan en $s = 0$, $s = -0.7$ y $s = -4$.

$$\left. \begin{aligned} G_{P1}(s) &= \frac{1}{s^3 + 10 \cdot s^2 + 21 \cdot s} \\ G_{P2}(s) &= \frac{1}{s^3 + 4.7 \cdot s^2 + 2.8 \cdot s} \end{aligned} \right\} \quad (6.18)$$

En la figura 6.14 se observa la salida del modelo de referencia y de las dos plantas. Como se puede observar, cuando los polos se acercan al eje $j\omega$, el comportamiento del sistema tiende a presentar más oscilaciones. Se puede concluir que el diseño basado en el método propuesto es estable y presenta robustez ante errores de modelado.

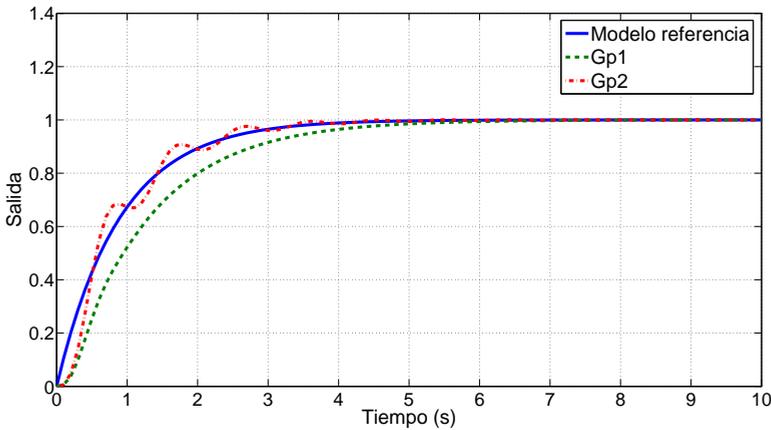


Figura 6.14: Respuesta del sistema ante errores de modelado.

Sección 6.3

Aportaciones

En este capítulo se ha diseñado el controlador inverso para dos plantas LTI. Para la planta de segundo orden, se han aportado las siguientes conclusiones:

1. Para ciertas aplicaciones, como es este caso, es mejor realizar una identificación aproximada de la planta.
2. Una red neuronal recurrente no es apropiada, ya que produce una señal de control que exige demasiado esfuerzo a la planta.
3. Para obtener una red neuronal no recurrente, es necesario igualar el orden del numerador a uno.

4. La mejora de la señal de control resultante se puede ajustar desde el proceso de identificación.
5. Las características del modelo de referencia elegido influyen en los resultados finales obtenidos.
6. Este sistema de control sencillo presenta una gran estabilidad y buena robustez.

Para la planta de tercer orden:

1. Debido a que una identificación exacta da lugar a una planta discreta de fase no mínima, el proceso de identificación es conveniente realizarlo con un orden del numerador igual a uno ($.Nu = 1$).
2. Aunque el índice VAF salga elevado, una identificación aproximada no es sinónimo de estabilidad en el sistema final obtenido.
3. Para estabilizar sistemas de control obtenidos mediante un modelo aproximado de la planta, se puede disminuir el periodo de muestreo del sistema final controlado con respecto al usado en la identificación.
4. Este sistema de control sencillo presenta una gran estabilidad.

La parte del controlador inverso para la planta dada por la función de transferencia (5.9) y presentada en el capítulo anterior, sección 5.5, fue diseñada según el método presentado en el apartado 6.2 de este capítulo. Dicha planta formó parte como ejemplo de la publicación:

Betancor, C. S., Sosa-González, C. J., Montiel, J. A., Vega, A., "Gains Tuning of a PI-Fuzzy controller by genetic algorithms", *Engineering Computations*, Emerald, Vol. 31, Issue 6, pp 1074-1097, 2014.

Control de no linealidades continuas

En el capítulo anterior se abordó el diseño del controlador inverso para sistemas lineales. Se discutió el caso de una planta LTI de segundo orden y otra de tercer orden. Esta última, cuando se discretizó, se convirtió en una planta de fase no mínima, dificultando, por tanto, su control.

Este capítulo se centrará en el control de sistemas con no linealidades continuas. Se comienza con un sistema con no linealidad continua y suave. Concretamente, el sistema vendrá definido por una ecuación diferencial no lineal de segundo orden. Un ejemplo de planta que presenta este comportamiento es el conjunto mecánico masa-resorte no lineal-amortiguador. Se realizarán comparativas con diversas técnicas que se han aplicado a este tipo de planta y se verá que el diseño propuesto ofrece mejores prestaciones.

Como segundo caso se aborda el control de un sistema que ofrece fuertes no linealidades, como es el caso del control del nivel de líquido en un tanque cónico. Las ventajas del control diseñado se demuestran mediante una serie de comparativas con diversas técnicas aplicadas en otras publicaciones. Además, se diseñará un controlador predictivo no lineal, con mucho mayor coste computacional. Entre las ventajas del método propuesto están que es un método simple, que la red neuronal se obtiene sin entrenamiento y el número de neuronas es reducido. Además, la consigna impuesta al proceso se alcanzará tan rápido como físicamente es posible y sin presentar sobreimpulso.

Para los dos tipos de plantas tratadas en este capítulo, será necesario diseñar un sistema que integre las salidas de las distintas neuronas, que surgen de los distintos modelos locales, resultantes del proceso de identificación.

El capítulo está organizado como sigue. En la sección 7.1, se realiza el control

de un sistema definido por una ecuación diferencial de segundo orden cuyas no linealidades son suaves. Para ello, se comienza con la descripción de un posible sistema mecánico que puede quedar definido por dicha ecuación diferencial. A continuación se exponen los resultados de diversas estrategias de control que se han propuesto en la literatura, cubriendo un amplio espectro de técnicas de control. Seguidamente, para comenzar con el diseño del control propuesto, se procede y se discute todo lo relacionado con la identificación de la planta para obtener un modelo Takagi-Sugeno. Con este modelo se explica y realiza el diseño del controlador inverso, junto con el sistema supervisor necesario para generar la señal de control. Una vez diseñado el controlador inverso, se muestran los resultados obtenidos. Finalmente, para anular los errores en régimen permanente y mejorar el transitorio, se diseña un PI borroso. El problema del control de nivel de líquido en un tanque cónico es el objeto de estudio en lo que resta de capítulo. En primer lugar, en la sección 7.2, se realiza una exposición sobre el estado del arte y posteriormente se describe la ecuación diferencial no lineal que modela al proceso. Se van a presentar dos tanques cónicos para diseñar el controlador y proceder a realizar comparativas con otras técnicas. En la sección 7.3 se estudia un tanque que posee una dinámica lenta en el vaciado. En primer lugar, se presentan los resultados obtenidos con controladores propuestos en la literatura. A continuación, se procede a la identificación para realizar el diseño del controlador inverso y proceder a mostrar los resultados obtenidos. Se continúa analizando la estabilidad y robustez del sistema diseñado ante incertidumbres de modelado y frente a perturbaciones. Para finalizar, y para aumentar el campo de comparativas, se extiende el diseño realizado en este trabajo a un controlador predictivo no lineal. Un segundo ejemplo, para un tanque cónico con una dinámica más rápida de descarga se presenta en la sección 7.4. En principio, se muestra el comportamiento con controladores PI diseñados por algunos autores. A continuación, se procede a identificar y a diseñar el controlador inverso. Los resultados obtenidos y el estudio de estabilidad y robustez ante errores de modelado y ante perturbaciones son presentados seguidamente. Finalmente, en la sección 7.5, se describen las aportaciones realizadas.

Sistema de segundo orden no lineal

7.1.1. Sistema masa-resorte-amortiguador

En la figura 7.1 se muestra un modelo mecánico constituido por una masa M , un resorte no lineal, con constantes k_1 y k_3 , y un amortiguador con fricción B .

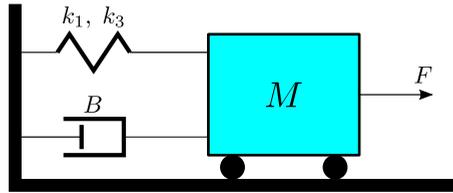


Figura 7.1: Sistema masa-resorte-amortiguador.

Los elementos de rigidez no lineales tienen muchas aplicaciones, como los muelles de las suspensiones de los vehículos, brazos robóticos, etc. Un resorte lineal, se rige por la ley de Hooke, donde el alargamiento x que experimenta el resorte es directamente proporcional a la fuerza aplicada F , según la constante elástica o recuperadora del resorte k : $F = k \cdot x$. Cuanto mayor es el valor de k , mayor fuerza se deberá aplicar para estirar el resorte, o sea, más rígido o duro es el resorte. Sin embargo, en un resorte no lineal la fuerza a aplicar al resorte es una función no lineal de la variable de desplazamiento x . Esta función se interpreta como una combinación de componentes de resortes lineales y no lineales, expresándose la relación fuerza-desplazamiento según la ecuación (7.1), donde el término $k_1 \cdot x$ corresponde a la parte lineal y el término $k_3 \cdot x^3$ a la parte no lineal. Si $k_3 > 0$ el resorte es duro y si $k_3 < 0$ es suave.

$$F = k_1 \cdot x + k_3 \cdot x^3 \quad (7.1)$$

De la figura 7.1, y aplicando la segunda ley de Newton ($F = m \cdot a$), se obtiene la ecuación diferencial (7.2) que modela al sistema.

$$\left. \begin{aligned} m \cdot a &= -B \cdot v - (k_1 \cdot x + k_3 \cdot x^3) + f \\ m \cdot \frac{d^2x(t)}{dt^2} + B \cdot \frac{dx(t)}{dt} + k_1 \cdot x + k_3 \cdot x^3 &= f \end{aligned} \right\} \quad (7.2)$$

Si, en (7.2), $m = 1 \text{ kg}$, $B = 1 \text{ N} \cdot \text{s}/\text{m}$, $k_1 = 1 \text{ N}/\text{m}$, $k_3 = 1 \text{ N}/\text{m}^3$ y se cambia la nomenclatura de y en lugar de x y de u por f , se obtiene la

ecuación diferencial (7.3), que se plantea en [4], como la planta no lineal para comprobar diferentes estrategias de control. El controlador deberá generar la señal de control $u(t)$ para que el valor de $y(t)$ sea el que se asigne como referencia o consigna. Se impondrá la restricción de que la máxima desviación permitida en el sistema va a ser de $y = \pm 2 m$.

$$\frac{d^2y(t)}{dt^2} + \frac{dy(t)}{dt} + y(t) + y^3(t) = u(t) \quad (7.3)$$

7.1.2. Resultados para diversas estrategias

En [4] se plantean y diseñan controladores mediante diferentes estrategias de control. En la figura 7.2 se muestra el resultado obtenido para un control inverso directo basado en redes neuronales. La red neuronal está constituida por siete neuronas en la capa oculta, con función de activación tangente hiperbólica para cada neurona, y una neurona de salida, con función de activación lineal. La red neuronal fue obtenida mediante entrenamiento general para actuar como el modelo inverso de la planta. Según los resultados, hay un sobreimpulso en los cambios de la señal de referencia, lo cual indica que el modelo inverso no es perfecto. Otro efecto que salta a la vista es que la señal de control oscila mucho y adquiere valores elevados.

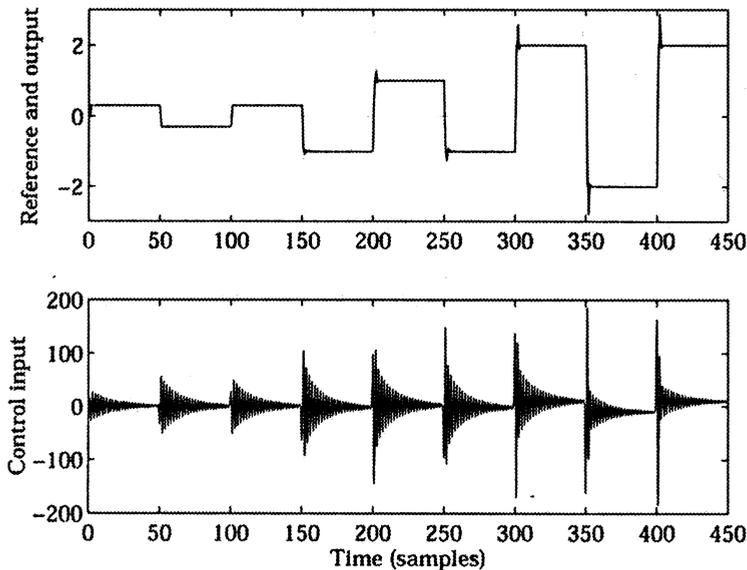


Figura 7.2: Control inverso directo (Fuente: [4]).

Para evitar los sobreimpulsos y limitar la amplitud de la señal de control, se puede poner un filtro paso-bajo en la señal de referencia. En la figura 7.3 se observa que han desaparecido los sobreimpulsos en la salida ante cambios de la señal de referencia y que la señal de control ha disminuido su amplitud, sin embargo, continúan las oscilaciones de la misma.

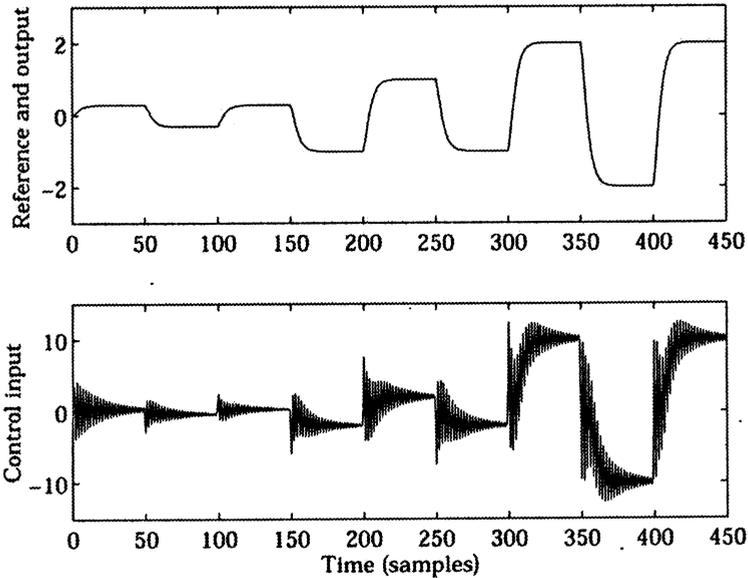


Figura 7.3: Control inverso directo con filtro paso-bajo (Fuente: [4]).

Usando entrenamiento especializado se obtiene el resultado de la figura 7.4. Para ello, se partió de la red creada usando entrenamiento general y el proceso de entrenamiento continuó directamente con la trayectoria de la señal de referencia. La trayectoria se repitió ocho veces mientras se actualizaba el modelo inverso. Se ve que la referencia se alcanza de forma perfecta, pero la señal de control tiene gran amplitud y oscila mucho.

Usando linealización por realimentación [171] los polos en lazo cerrado se sitúan en $z = -0.7$. Para ello se utilizan dos redes neuronales con cinco neuronas, con función tangente hiperbólica cada una. En la figura 7.5 se observa que la salida sigue a la trayectoria de referencia y que la señal de control es muy activa.

Realizando el control con el PID dado en (7.4), se obtiene el resultado de la figura 7.6. Como el PID se diseñó para el modelo linealizado en estado de reposo, las prestaciones son mejores para pequeños cambios de la referencia.

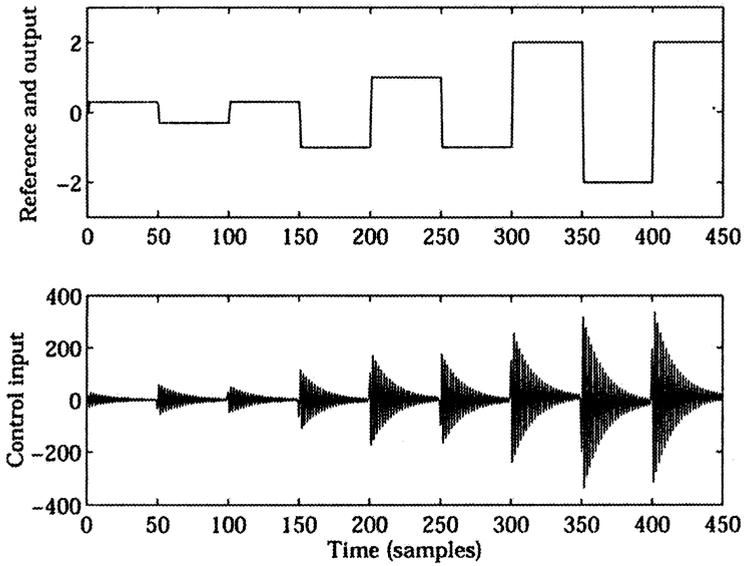


Figura 7.4: Control inverso mediante entrenamiento especializado (Fuente: [4]).

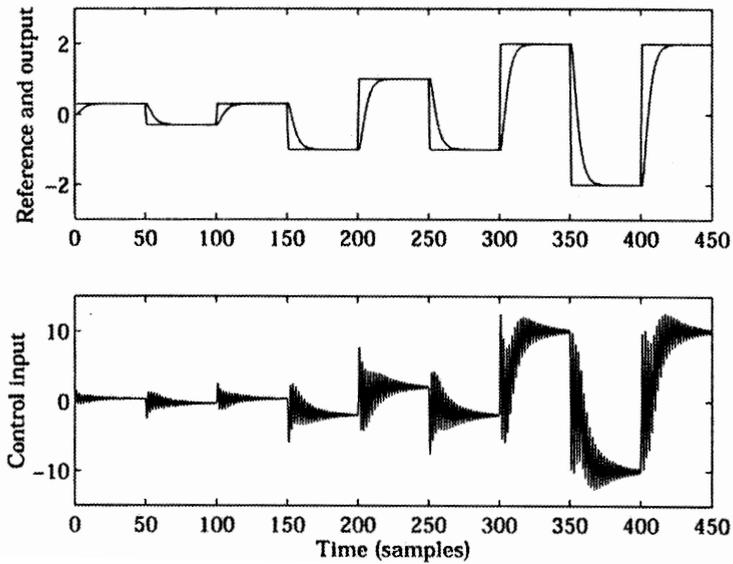


Figura 7.5: Control por linealización mediante realimentación (Fuente: [4]).

$$G_{PID}(s) = 8 \cdot \frac{5s + 1}{5s} \cdot \frac{0.8s + 1}{0.08s + 1} \quad (7.4)$$

Si al PID se añade una señal de compensación proveniente del modelo inverso creado con el entrenamiento especializado, se obtiene la respuesta mostrada en la figura 7.7. Se ve que el seguimiento de la referencia es mejor, aunque cuando la amplitud de la referencia aumenta el modelo inverso no es suficientemente preciso.

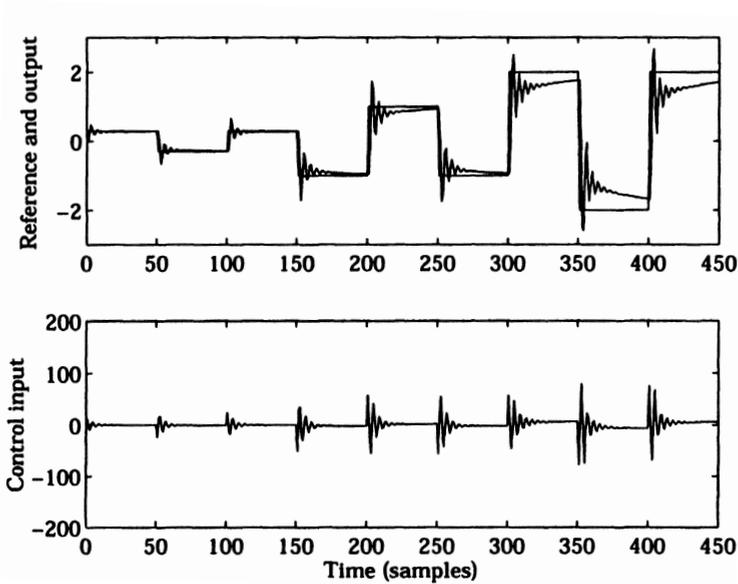


Figura 7.6: Control con un PID (Fuente: [4]).

La figura 7.8 ilustra la respuesta de un control óptimo diseñado mediante el algoritmo de entrenamiento especializado que se aplicó al control inverso directo. Como se observa, el seguimiento de la referencia se ve deteriorada con respecto al control inverso, ya que se producen sobreimpulsos y hay error en régimen permanente. Sin embargo, la señal de control es más suave y no alcanza los valores elevados del control inverso directo. Los sobreimpulsos y el error en régimen permanente se pueden reducir si se aminora el factor de penalización, aunque la señal de control será más activa.

Mediante el posicionamiento de polos, y la completa cancelación de ceros, se obtiene el resultado mostrado en la figura 7.9. Aquí la señal de control no es de amplitud elevada, pero sí es bastante activa.

Si se considera el posicionamiento de polos, sin la cancelación de ceros, se obtiene la respuesta de la figura 7.10, donde las oscilaciones de la señal de control de la figura 7.9 se han eliminado.

La respuesta para un control predictivo generalizado no lineal —*Nonlinear*

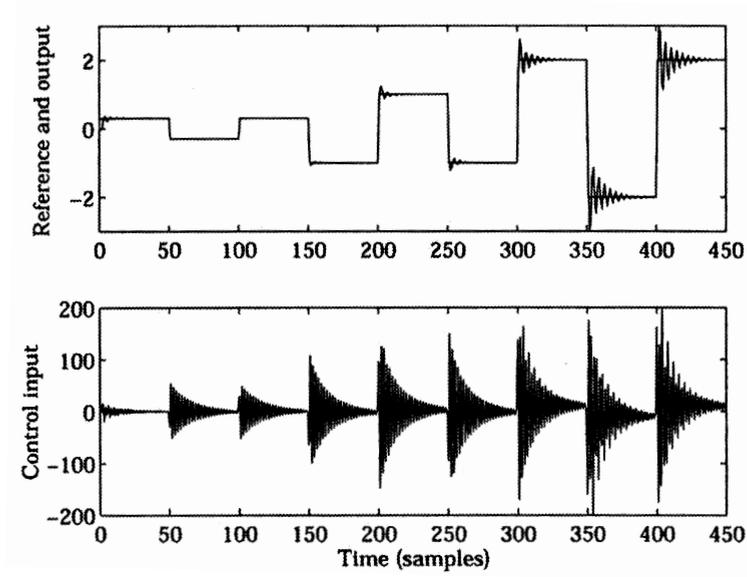


Figura 7.7: Control PID con retroalimentación de la referencia (Fuente: [4]).

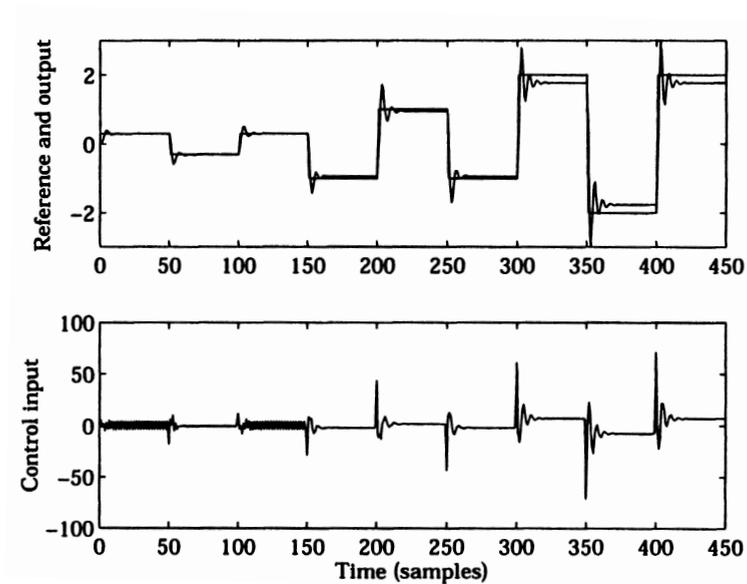


Figura 7.8: Control óptimo (Fuente: [4]).

Generalized Predictive Control (NGPC)— se presenta en la figura 7.11. Según

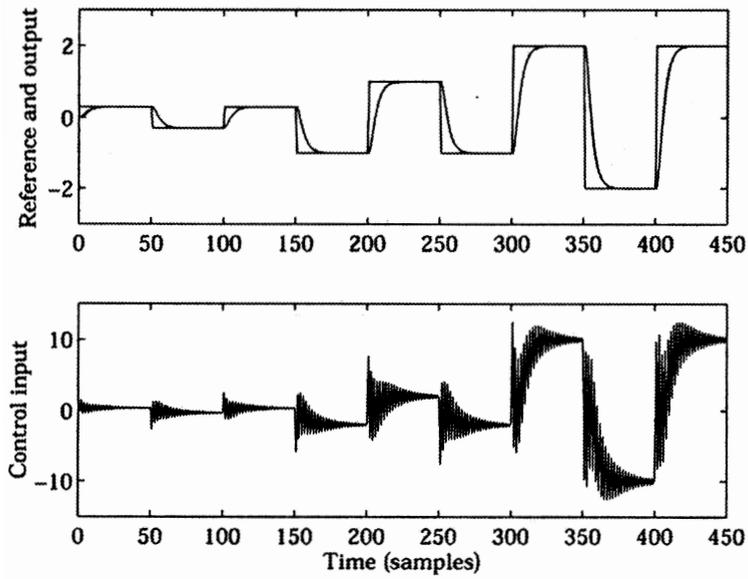


Figura 7.9: Control con polo que cancela el cero (Fuente: [4]).

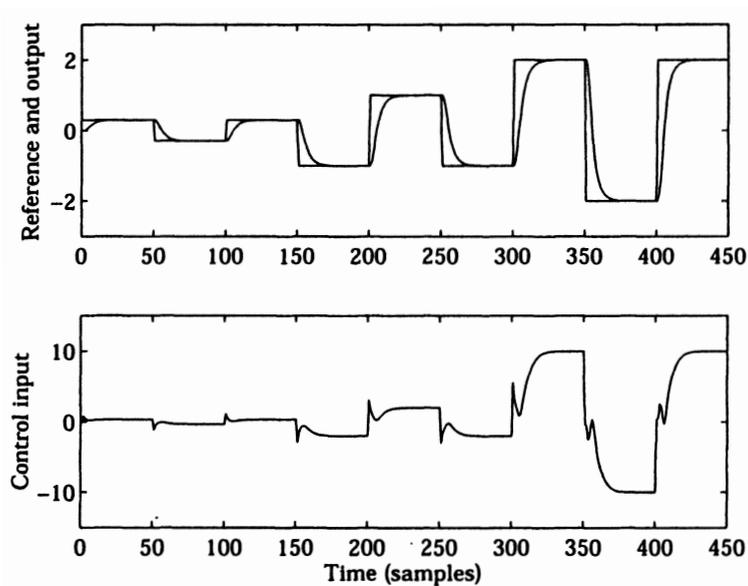


Figura 7.10: Control con polo sin cancelación del cero (Fuente: [4]).

la figura, se consigue una respuesta bastante rápida a los cambios de referencia

mientras que la señal de control se mantiene con magnitudes pequeñas. También se puede apreciar que el controlador se anticipa a los cambios en escalón de la referencia, debido a que el control está basado en las referencias futuras. Además, la respuesta muestra un comportamiento de fase no mínima, aunque el inverso del sistema es estable. Esto no es infrecuente cuando un sistema se controla mediante un controlador predictivo.

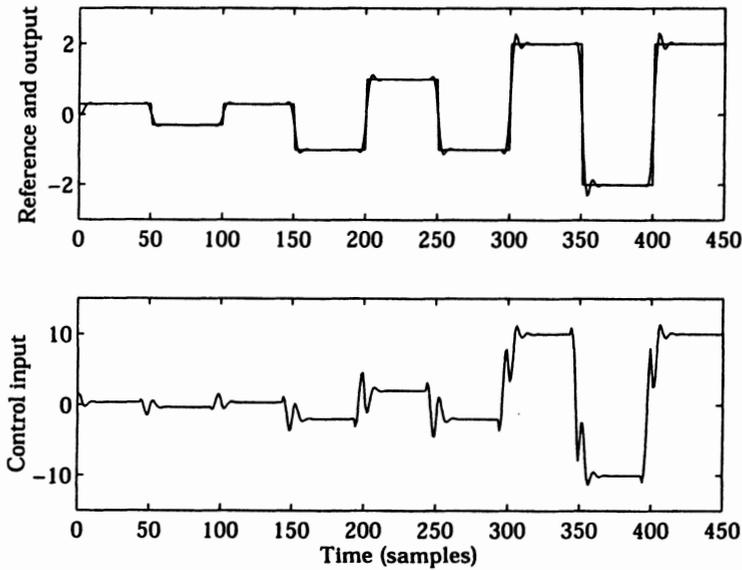


Figura 7.11: Control predictivo no lineal (Fuente: [4]).

Debido a los requerimientos hardware de NGPC, se puede optar por usar un control predictivo aproximado. La figura 7.12 muestra la respuesta y se puede observar que es parecida a la ofrecida por NGPC, siendo esto debido a que la no linealidad de la planta es suave.

7.1.3. Identificación de la planta

Para realizar la identificación de la planta 7.3, y obtener un modelo borroso Takagi-Sugeno, se ha experimentado con distintos tipos de señales de entrada, siendo una señal distribuida uniformemente para u en el rango $[-10, 10]$ la que mejores resultados ha dado (figura 7.13).

Las señales del tipo secuencia binaria pseudo aleatoria —*Pseudo Random Binary Sequence* (PBRS)— han dado peores resultados en la validación. El periodo de muestreo elegido ha sido de $T = 0.1$ s y, como siempre, la mitad

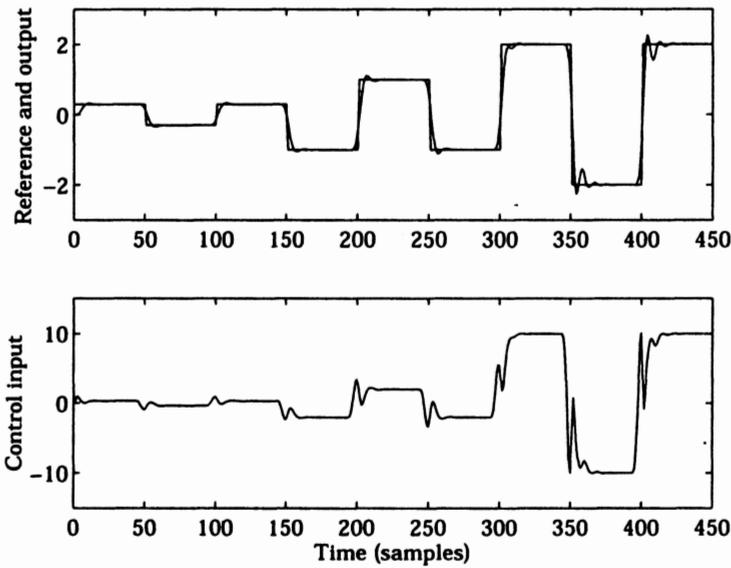


Figura 7.12: Control predictivo generalizado aproximado (Fuente: [4]).

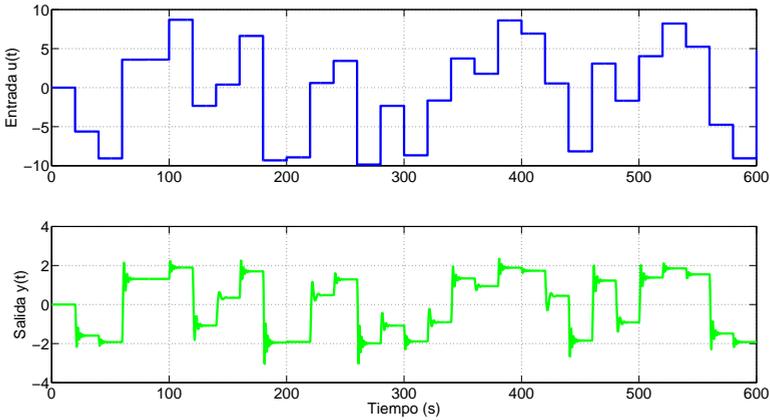


Figura 7.13: Señales para la identificación.

de las muestras se emplean en el proceso de identificación y la otra mitad en el proceso de validación.

El rango de $[-10, 10]$ para u es por el hecho de que el máximo desplazamiento permitido en el sistema, según las especificaciones impuestas, es de $\pm 2 m$, siendo ese rango de u el que produce aproximadamente esta variación máxima.

Para obtener un buen modelo, supuesto que las señales de entrada para la identificación han sido correctamente elegidas, es necesario dedicar tiempo a la sintonización correcta de los parámetros que definen las constantes que utiliza el algoritmo de agrupamiento. Para uno y dos clústers el VAF obtenido es bajo, por lo que los resultados que se presentan a continuación son de tres clústers en adelante. Asimismo, aumentar el orden del denominador ($.Ny$) por arriba de 2 no implica obtener un modelo más exacto y aumentar el orden del numerador ($.Nu$) por arriba de 1 tampoco supone mejora. Se elige una tolerancia de terminación $.tol = 0.001$ y la variación de la semilla de inicialización ($.seed$), una vez que se han optimizado los demás parámetros, no ha implicado una mejora apreciable en el VAF final.

La figura 7.14 representa el VAF al variar el número de clústers, siendo los valores para el resto de parámetros del algoritmo de identificación los dados en la tabla 7.1.

TABLA 7.1: PARÁMETROS PARA LA IDENTIFICACIÓN DEL SISTEMA NO LINEAL

Coefficiente de borrosificación	$.m = 2$
Tolerancia de terminación	$.tol = 0.001$
Semilla para la inicialización	$.seed = 10$
Tipo de los antecedentes	$.ante = 2$
Estimación del consecuente	$.cons = 1$
Número de retardos de la salida	$.Ny = 2$
Número de retardos de la entrada	$.Nu = 1$
Número de retardos de transporte	$.Nd = 1$

En la figura 7.15 se observa el VAF al variar el coeficiente de borrosificación ($.m$) para distintos números de clústers.

Para esta planta se pueden sacar las siguientes conclusiones:

1. Un aumento inicial del número de clústers supone una mejora sustancial, pero un aumento excesivo no implica una gran mejora. Por ejemplo, al duplicar de 5 a 10 el número de clústers, el VAF varió del 99.8739 % al 99.9713 %, es decir una variación del 0.0974 %. El aumentar el número de clústers supone un aumento del tiempo de cómputo para obtener el modelo, además de que la implementación hardware posterior será más compleja, ya que el número de neuronas en el sistema de control va a ser igual al número de clústers obtenidos.
2. Variar el exponente de borrosificación por arriba o por debajo del valor

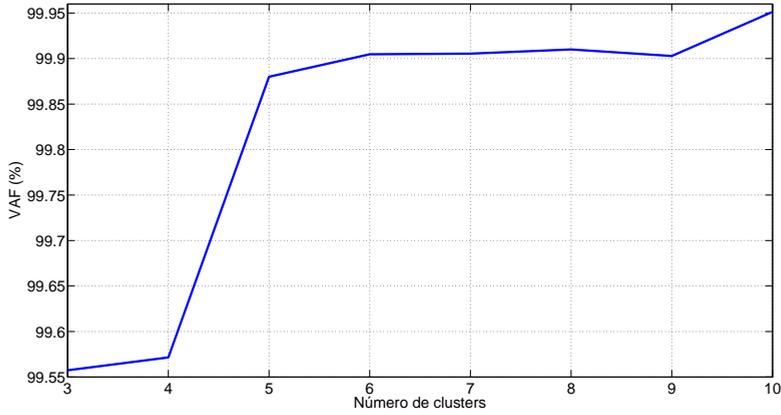


Figura 7.14: VAF obtenido al variar el número de clústers.

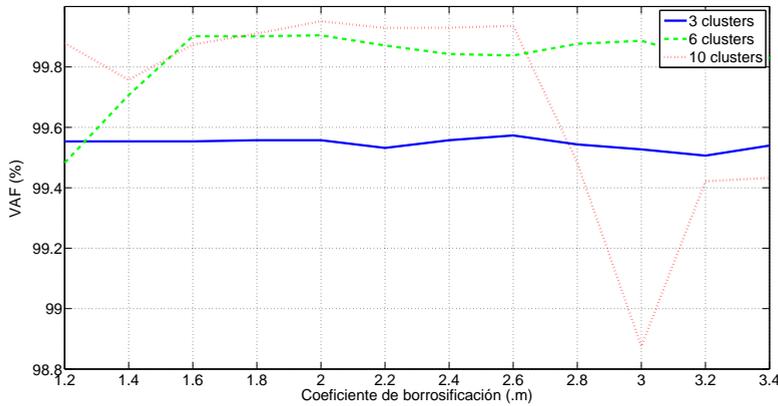


Figura 7.15: VAF obtenido al variar el coeficiente de borrosificación.

2 puede implicar una ligera mejora. Según la figura 7.15, el coeficiente de borrosificación afecta más conforme se aumenta el número de clústers. Para 3 clústers la influencia es mínima, mientras que para 6 y 10 sí afecta su elección.

- Al modificar el número de clústers, el mejor valor del coeficiente de borrosificación varía. Si se eligen 6 clústers para el modelo final, el mejor coeficiente de borrosificación tiene el valor de 2.02, el cual no coincide con el mejor para 5 clústers. En definitiva, una vez determinado el número de clústers, se debería modificar el coeficiente de borrosificación

para una sintonía fina.

4. El VAF no crece o decrece de forma uniforme según el coeficiente de borrosificación varía, y el disminuir la tolerancia de terminación hace que unas veces el VAF mejore, otras empeore y otras no varíe, sin seguir una regla.
5. Si se desea obtener un $VAF \geq 99.9\%$, se deberá escoger un mínimo de 6 clústers.

Ahora se procede a sintonizar los parámetros del algoritmo de agrupamiento para 3, 6 y 10 clústers. De esta forma, posteriormente se realizará una comparativa del sistema controlado para estos tres casos. Se verá si el aumentar significativamente el número de clústers, y por tanto de neuronas, conduce a una mejora apreciable en las prestaciones del sistema controlado obtenido.

Según los resultados de las diferentes simulaciones, los mejores valores de sintonía de parámetros para 3, 6 y 10 clústers se muestran en la tabla 7.2.

TABLA 7.2: VALORES FINALES PARA EL ALGORITMO DE AGRUPAMIENTO

.c	.m	.tol	.seed	.ante	.cons	Ny	.Nu	.Nd	VAF (%)
3	2.56	0.001	10	2	1	2	1	1	99.5759
6	1.90	0.001	10	2	1	2	1	1	99.9070
10	2.00	0.001	10	2	1	2	1	1	99.9513

Para el caso de 3 clústers, en la figura 7.16 se observa el resultado del proceso de validación, en el que se obtuvo un $VAF = 99.5759\%$, y las tres reglas que describen el modelo borroso están dadas en (7.5).

$$\left. \begin{aligned}
 R_1 : & \text{ If } y(k-1) \text{ is } A_{11} \text{ and } y(k-2) \text{ is } A_{12} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{13} \text{ then } y(k) = 1.78 \cdot 10^0 y(k-1) - \\
 & \quad - 8.93 \cdot 10^{-1} y(k-2) + 1.02 \cdot 10^{-2} u(k-1) - 1.28 \cdot 10^{-1} \\
 R_2 : & \text{ If } y(k-1) \text{ is } A_{21} \text{ and } y(k-2) \text{ is } A_{22} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{23} \text{ then } y(k) = 1.90 \cdot 10^0 y(k-1) - \\
 & \quad - 9.20 \cdot 10^{-1} y(k-2) + 8.58 \cdot 10^{-3} u(k-1) + 1.23 \cdot 10^{-3} \\
 R_3 : & \text{ If } y(k-1) \text{ is } A_{31} \text{ and } y(k-2) \text{ is } A_{32} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{33} \text{ then } y(k) = 1.83 \cdot 10^0 y(k-1) - \\
 & \quad - 8.98 \cdot 10^{-1} y(k-2) + 7.84 \cdot 10^{-3} u(k-1) + 5.92 \cdot 10^{-2}
 \end{aligned} \right\} \quad (7.5)$$

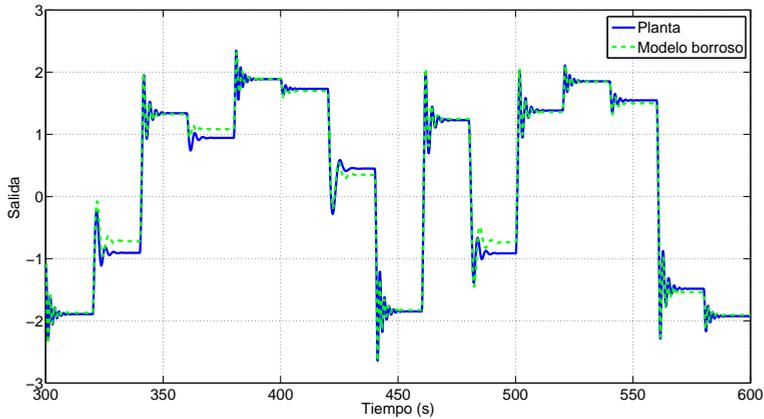


Figura 7.16: Resultado de la validación para 3 clústers.

7.1.4. Diseño del controlador inverso

El tiempo de establecimiento, t_e , del modelo de referencia se ha fijado mediante simulación. Partiendo de $t_e = 1 s$, se ha visto que la respuesta del sistema controlado era subamortiguada. Para evitar este comportamiento se ha ido incrementando este tiempo de establecimiento. Para $t_e = 3 s$ el sistema ya presenta respuesta monótona, similar a la del modelo de referencia. La ecuación en diferencias que describe al modelo de referencia es la dada en (7.6), con $t_e = 3 s$ y periodo de muestreo $T = 0.1 s$.

$$y_r(k) = 0.8777y_r(k-1) + 0.1223r(k-1) \quad (7.6)$$

En los diseños vistos hasta ahora se ha obtenido una única neurona ya que con un clúster era suficiente para sistemas LTI. En el caso que nos ocupa se obtendrá más de una neurona ya que para sistemas no lineales es necesario más de un clúster. A continuación se presenta el desarrollo para el modelo borroso (7.5) con 3 clústers, siendo el procedimiento el mismo para otro número de clústers.

Igualando el consecuente de las reglas del modelo borroso (7.5) con la salida deseada $y_r(k)$ del modelo de referencia establecido (7.6), y despejando el término $u(k-1)$, se obtiene una señal de control para cada regla (7.7).

$$\left. \begin{aligned}
 u_1(k) &= + \frac{(0.8777 - 1.78 \cdot 10^0)}{1.02 \cdot 10^{-2}} \cdot y(k) + \frac{8.93 \cdot 10^{-1}}{1.02 \cdot 10^{-2}} \cdot y(k-1) + \\
 &\quad + \frac{0.1223}{1.02 \cdot 10^{-2}} \cdot r(k) + \frac{1.28 \cdot 10^{-1}}{1.02 \cdot 10^{-2}} \\
 u_2(k) &= + \frac{(0.8777 - 1.90 \cdot 10^0)}{8.58 \cdot 10^{-3}} \cdot y(k) + \frac{9.20 \cdot 10^{-1}}{8.58 \cdot 10^{-3}} \cdot y(k-1) + \\
 &\quad + \frac{0.1223}{8.58 \cdot 10^{-3}} \cdot r(k) - \frac{1.23 \cdot 10^{-3}}{8.58 \cdot 10^{-3}} \\
 u_3(k) &= + \frac{(0.8777 - 1.83 \cdot 10^0)}{7.84 \cdot 10^{-3}} \cdot y(k) + \frac{8.98 \cdot 10^{-1}}{7.84 \cdot 10^{-3}} \cdot y(k-1) + \\
 &\quad + \frac{0.1223}{7.84 \cdot 10^{-3}} \cdot r(k) - \frac{5.92 \cdot 10^{-2}}{7.84 \cdot 10^{-3}}
 \end{aligned} \right\} \quad (7.7)$$

Desde el punto de vista de la implementación con redes neuronales, las leyes de control (7.7) tienen la forma dada en (7.8). De este modo, cada una de las leyes de control $u_i(k)$ se pueden implementar mediante una neurona, con 3 entradas $-y(k)$, $y(k-1)$ y $r(k)$, una polarización $-b_i$ y función de activación lineal, representando cada neurona a un subsistema que representa el control relacionado con un modelo local.

$$\left. \begin{aligned}
 u_1(k) &= w_{11} \cdot y(k) + w_{12} \cdot y(k-1) + w_{13} \cdot r(k) + b_1 \\
 u_2(k) &= w_{21} \cdot y(k) + w_{22} \cdot y(k-1) + w_{23} \cdot r(k) + b_2 \\
 u_3(k) &= w_{31} \cdot y(k) + w_{32} \cdot y(k-1) + w_{33} \cdot r(k) + b_3
 \end{aligned} \right\} \quad (7.8)$$

7.1.5. Diseño del sistema supervisor

Cada neurona obtenida en la sección anterior es un controlador y es necesario diseñar un bloque supervisor que seleccione la combinación más adecuada de entre los distintos reguladores para los distintos modelos locales. Debido a que el antecedente de cada regla es borroso, el bloque que integrará a todos los subsistemas será un sistema de inferencia borroso (figura 7.17). En el caso que nos ocupa, la red neuronal diseñada posee 3 neuronas, disponiéndose de 3 salidas a partir de las cuales habrá que generar una única señal de control hacia la planta.

Para que una regla esté activa, en el modelo borroso (7.5), se deben activar los tres conjuntos borrosos del antecedente, dado que la conectiva es la ope-

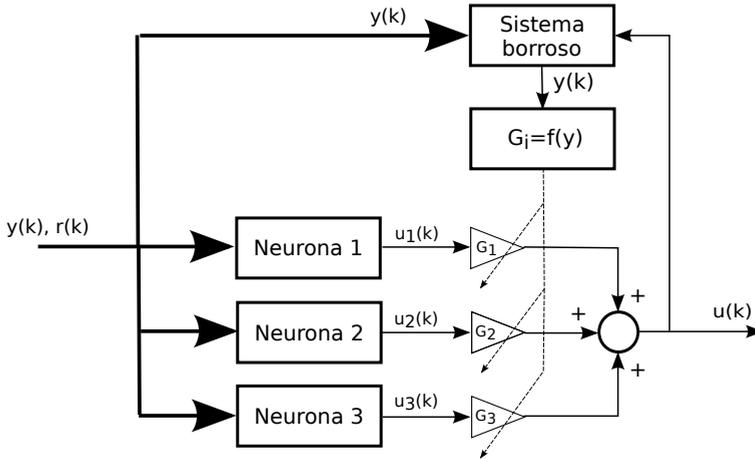


Figura 7.17: Integración de los subsistemas.

ración *and*. En la figura 7.18 se muestra el proceso de inferencia generado y se ve que, según los valores de las entradas presentados para $y(k-1) = -3$, $y(k-2) = -1.1$ y $u(k-1) = -4.6$, la regla activa es R_1 y la salida generará un valor $y = 1$, indicando que se deberá seleccionar la neurona que genera la señal de control $u_1(k)$ ($G_1 = 1$). De la regla R_2 sólo están activos dos conjuntos borrosos y no da lugar a señal de control alguna ($G_2 = 0$) y de la regla R_3 no se activa ningún conjunto borroso ($G_3 = 0$). Así pues, en el caso de que únicamente haya una regla activa, la selección de la salida corresponde a esa regla.

También se puede dar la circunstancia de que se activen dos reglas. Para este caso, dependiendo del método elegido para el conectivo *and* y la estrategia que implemente el proceso de desborrosificación, se obtendrá un valor diferente en la salida. La figura 7.19 muestra el caso en el que se activan las reglas R_2 y R_3 , y se emplea el método del mínimo para el conectivo *and* y el promedio ponderado para la desborrosificación, resultando un valor de $y = 2.76$. Este valor de y indicará que la salida $u_1(k)$ tendrá un peso del 0% en la señal de control final ($G_1 = 0$), $u_2(k)$ tendrá un peso del 24% ($G_2 = 0.24$) y la $u_3(k)$ un peso del 76% ($G_3 = 0.76$). Si se emplease el método del producto para el conectivo *and*, manteniéndose el promedio ponderado para la desborrosificación, el valor resultante será $y = 2.89$ (figura 7.20), con lo cual la salida $u_2(k)$ tendría un peso del 11% en la señal de control final y la $u_3(k)$ tendrá un peso del 89%.

En la desborrosificación también se puede usar la suma ponderada. Sin embargo, en los experimentos realizados, mediante simulación, se han obtenido mejores resultados con el método del mínimo para el conectivo *and* y el pro-

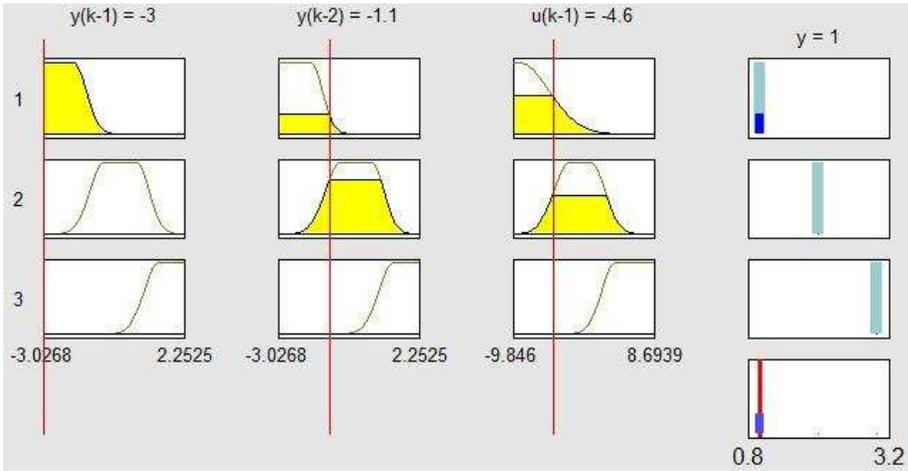


Figura 7.18: Caso en el que se activa la regla 1.

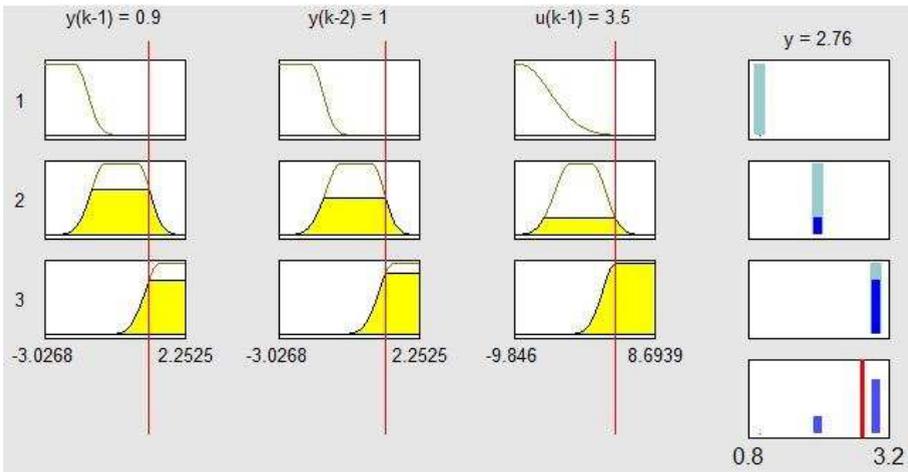


Figura 7.19: Reglas 2 y 3 activas y método del mínimo para la *and*.

medio ponderado para la desborrosificación.

7.1.6. Resultados obtenidos

En esta sección se realizarán simulaciones para estudiar el comportamiento del sistema y comenzar a sacar las primeras conclusiones sobre el control de sistemas no lineales por el método propuesto. Desde la figura 7.2 a la 7.12, se mostraron las respuestas obtenidas en [4] para diversas estrategias de control.

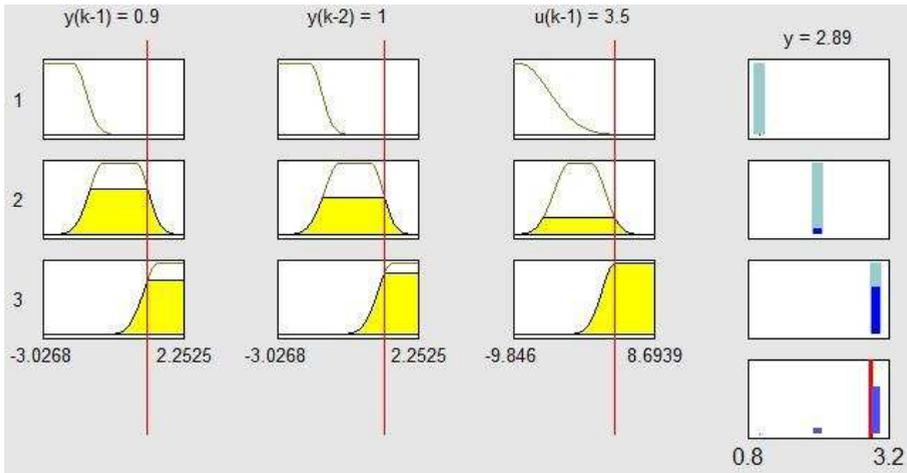


Figura 7.20: Reglas 2 y 3 activas y método del producto para la *and*.

En la figura 7.21 se observa la respuesta obtenida con 3, 6 y 10 clústers para el controlador inverso diseñado. Tal como se puede apreciar, las respuestas son prácticamente idénticas, existiendo un pequeño error en régimen permanente que se muestra en la figura 7.22. Las señales de control necesarias se ven en la figura 7.23.

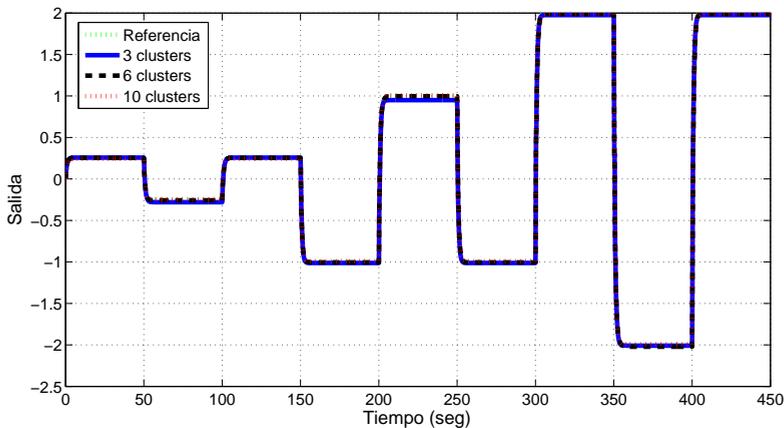


Figura 7.21: Respuestas obtenidas para 3, 6 y 10 clústers.

La amplitud de la señal de control aumenta conforme se pida mayor rapidez al modelo de referencia. Una ampliación de alguna transición (figura 7.24) muestra con mayor detalle la evolución de la señales de control. En todas las

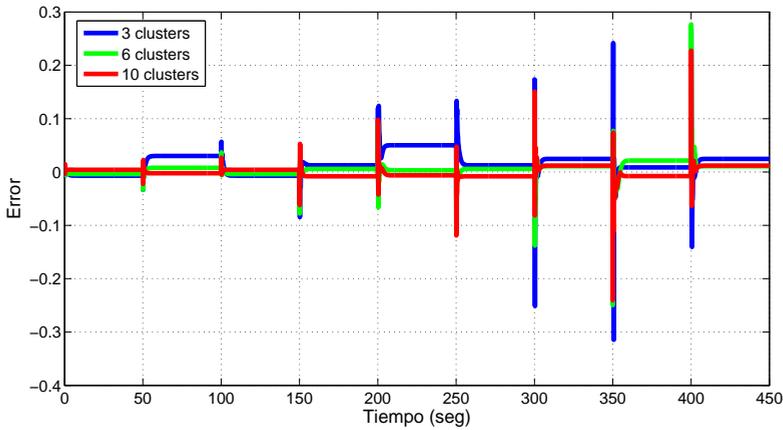


Figura 7.22: Errores para 3, 6 y 10 clústers.

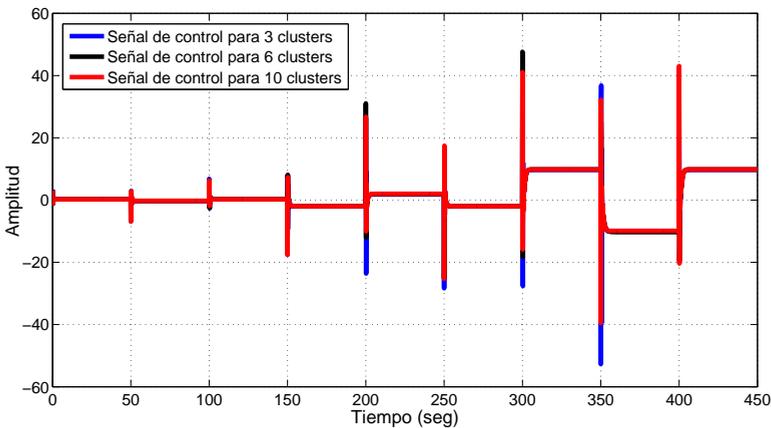


Figura 7.23: Señales de control generadas para 3, 6 y 10 clústers.

transiciones el control con 6 clústers es el que mejor comportamiento ofrece, pero quizás no compense el aumento del número de neuronas con respecto a las prestaciones con 3 clústers.

Analizando la expresión de las señales de control, dadas en (7.7), la magnitud de la señal de control $u_i(k)$ depende del denominador de los diferentes términos y este denominador es el coeficiente de $u(k-1)$ de (7.5). Por lo tanto, aumentar el valor de los coeficientes de $u(k-1)$ produciría el efecto de disminuir la amplitud de la señal de control. Si se utiliza un periodo de muestreo $T = 0.2 s$, el modelo borroso obtenido (7.9) presenta coeficientes mayores para $u(k-1)$

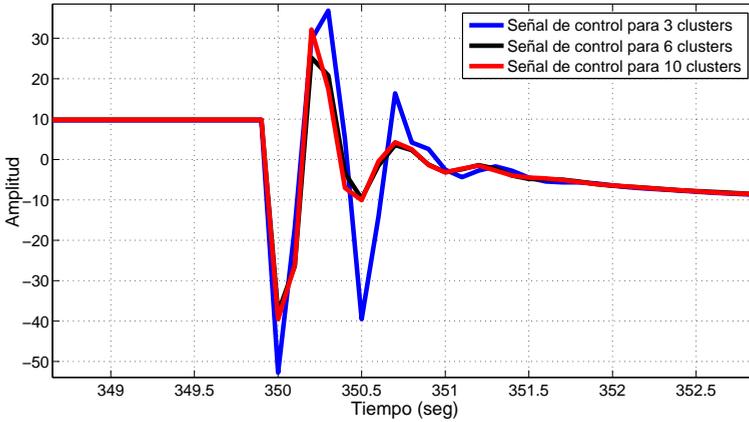


Figura 7.24: Ampliación de la figura 7.23 entorno a los 350 s.

y por lo tanto la señal de control tendrá valores inferiores a costa de perder exactitud en la respuesta. Se realiza la simulación y en la respuesta (figura 7.25) se observa que se produce un mayor error en régimen permanente y el transitorio es peor, aunque admisible (figuras 7.26 y 7.27). En cuanto a la señal de control (figura 7.28) se ve claramente que ha disminuido aunque en las transiciones presenta mayor número de oscilaciones (figura 7.29).

$$\left. \begin{aligned}
 R_1 : & \text{ If } y(k-1) \text{ is } A_{11} \text{ and } y(k-2) \text{ is } A_{12} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{13} \text{ then } y(k) = 1.37 \cdot 10^0 y(k-1) - \\
 & \quad - 7.76 \cdot 10^{-1} y(k-2) + 3.60 \cdot 10^{-2} u(k-1) - 4.58 \cdot 10^{-1} \\
 R_2 : & \text{ If } y(k-1) \text{ is } A_{21} \text{ and } y(k-2) \text{ is } A_{22} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{23} \text{ then } y(k) = 1.80 \cdot 10^0 y(k-1) - \\
 & \quad - 8.66 \cdot 10^{-1} y(k-2) + 2.96 \cdot 10^{-2} u(k-1) + 3.90 \cdot 10^{-3} \\
 R_3 : & \text{ If } y(k-1) \text{ is } A_{31} \text{ and } y(k-2) \text{ is } A_{32} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{33} \text{ then } y(k) = 1.56 \cdot 10^0 y(k-1) - \\
 & \quad - 7.92 \cdot 10^{-1} y(k-2) + 2.72 \cdot 10^{-2} u(k-1) + 2.06 \cdot 10^{-1}
 \end{aligned} \right\} \quad (7.9)$$

Con el modelo (7.9), obtenido con $T = 0.2$ s, se realiza ahora el control con $T = 0.1$ s. El resultado se presenta en la figura 7.30. En la ampliación realizada a los 400 s (figura 7.31), se observa que la transición es más suave que la obtenida para $T = 0.2$ s en el control. En cuanto a la señal de control

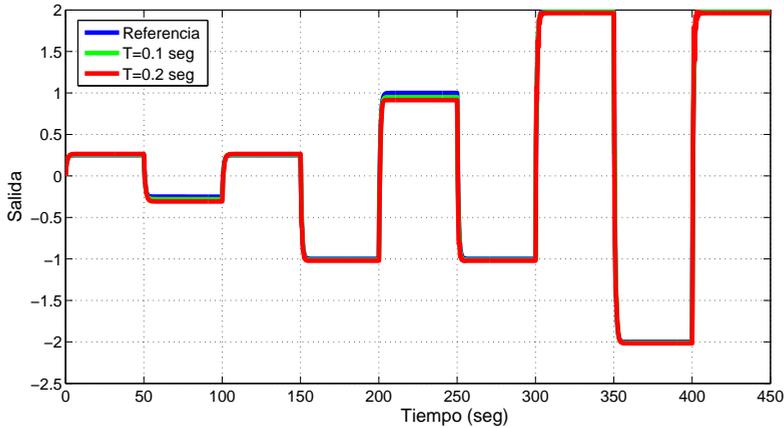


Figura 7.25: Respuestas para $T = 0.1$ s y $T = 0.2$ s.

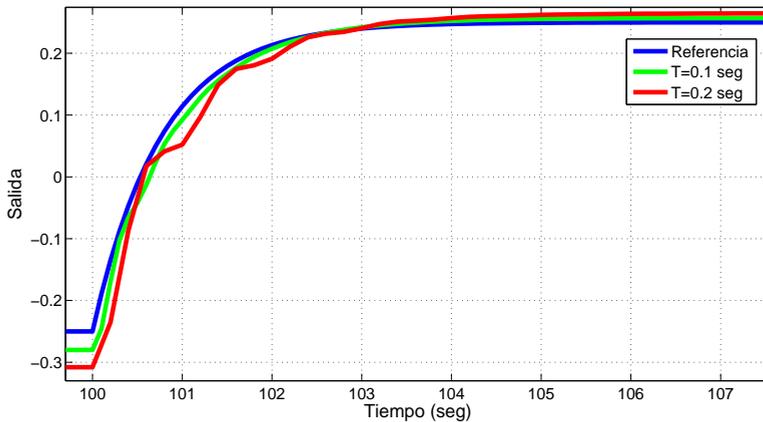


Figura 7.26: Ampliación de la figura 7.25 en $t = 100$ s.

(figura 7.32) se ve claramente (figura 7.33) que es mucho mejor que la obtenida cuando se usa $T = 0.2$ s en el control.

Los mismos experimentos realizados para 3 clústers se llevan a cabo para 6 y 10 clústers. La figura 7.34 presenta una comparativa de las respuestas para 3, 6 y 10 clústers cuando se utiliza $T = 0.2$ s tanto en la identificación como en el posterior control. Se observa que en todas las transiciones y regímenes permanentes la mejor respuesta corresponde a 6 clústers. Las señales de control necesarias (figura 7.35) indican que dependiendo de la transición el comportamiento es mejor para 3 o para 6 clústers, no pudiéndose determinar en general

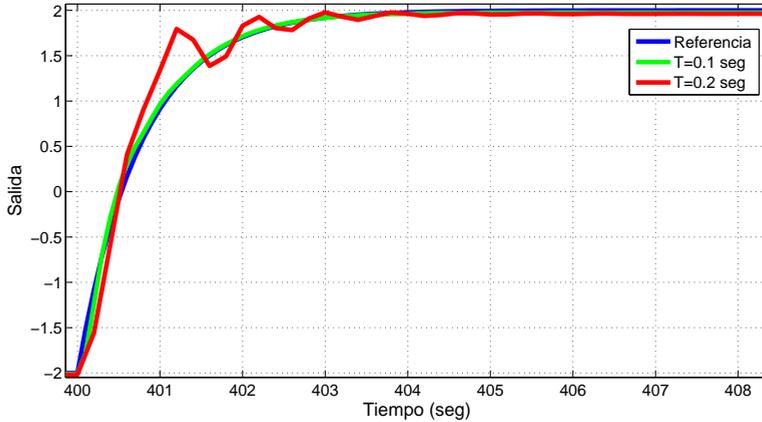


Figura 7.27: Ampliación de la figura 7.25 en $t = 400$ s.

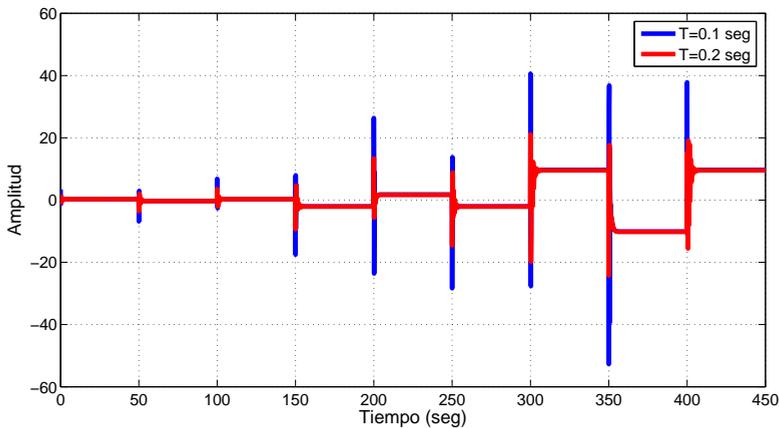


Figura 7.28: Señales de control para $T = 0.1$ s y $T = 0.2$ s.

cual es mejor. Para las transiciones de pequeña amplitud la respuesta es mejor para 10 clústers.

Cuando se usa $T = 0.2$ s en la identificación y $T = 0.1$ s en el control, en general, con 6 clústers (figura 7.36) implica mejor respuesta, salvo en la transición de 350 s donde el error en régimen permanente es mayor para 6 clústers que para 3 clústers. La conducta para 10 clústers es siempre la peor en todas las transiciones. En cuanto a la señal de control generada (figura 7.37), para 3 y 6 clústers son muy similares, mientras que para 10 clústers presenta siempre una mayor amplitud.

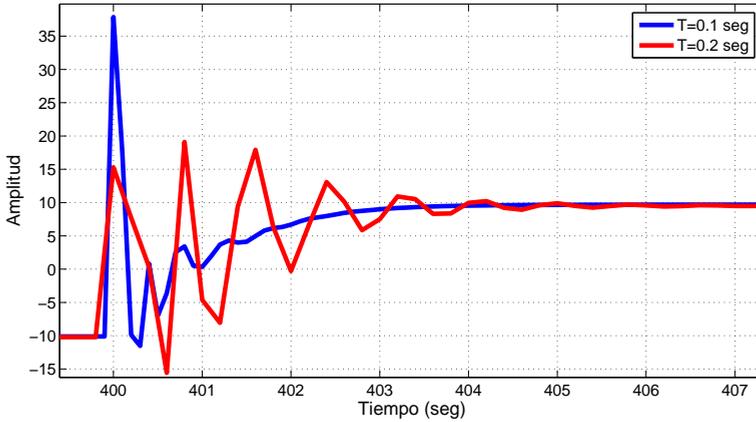


Figura 7.29: Ampliación de la figura 7.28 en $t = 400$ s.

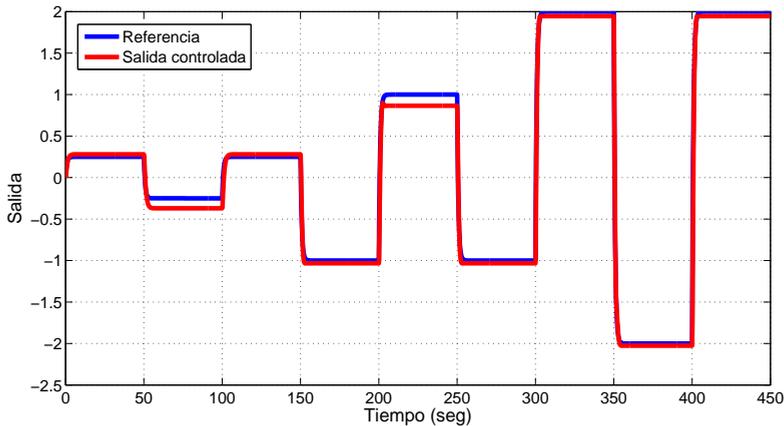


Figura 7.30: Identificación con $T = 0.2$ s y control con $T = 0.1$ s.

De los resultados obtenidos se puede deducir que con 3 clústers y $T = 0.2$ s tanto en la identificación como en el control se consiguen resultados aceptables de forma que se consuman los mínimos recursos hardware. En el siguiente punto se usará esta configuración y se cancelará el error en régimen permanente añadiendo un PI borroso.

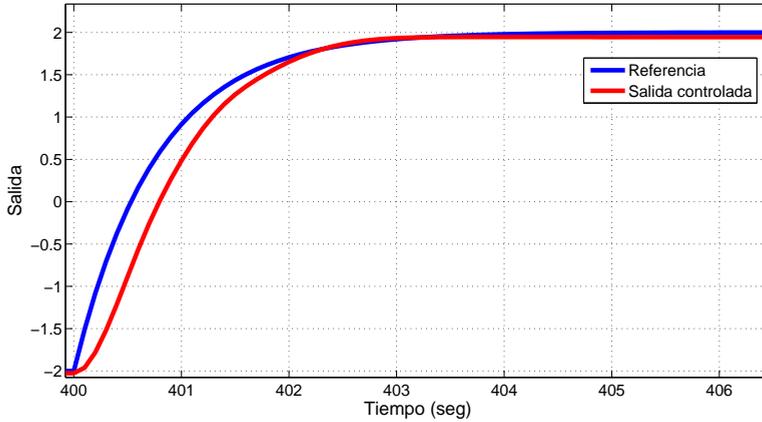


Figura 7.31: Ampliación de la figura 7.30 en $t = 400$ s.

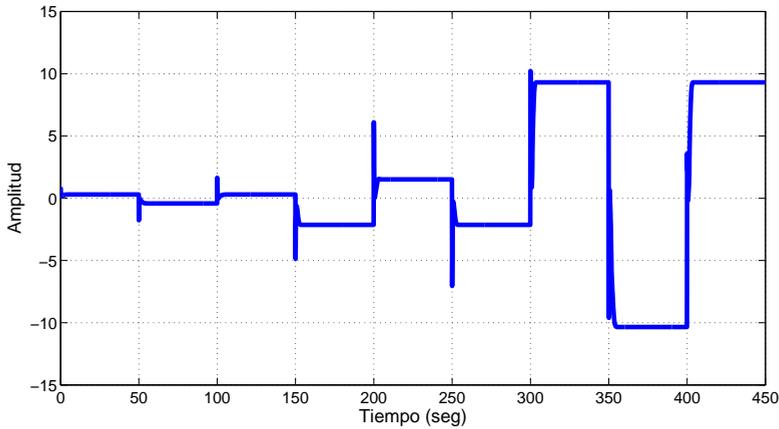


Figura 7.32: Señal de control para diferentes muestreos.

7.1.7. Cancelación de los errores en régimen permanente

Para eliminar los errores en régimen permanente, se opta por la misma solución que la propuesta en la sección 4.6 del capítulo 2, es decir, hacer uso de un PI borroso que procese el error entre la salida deseada del modelo de referencia y la salida real de la planta. El PI borroso no sólo corregirá el error en régimen permanente, sino que intentará ayudar al controlador inverso a mejorar la respuesta transitoria.

Se parte del PI borroso diseñado en el capítulo 2 para un sistema lineal y lo

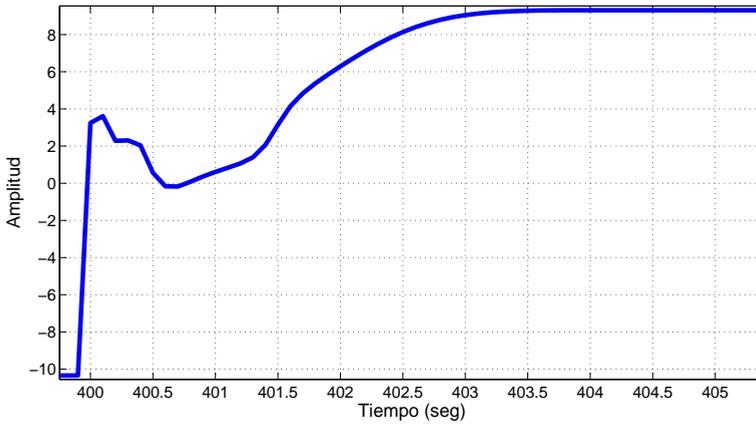


Figura 7.33: Ampliación de la figura 7.32 en $t = 400$ s.

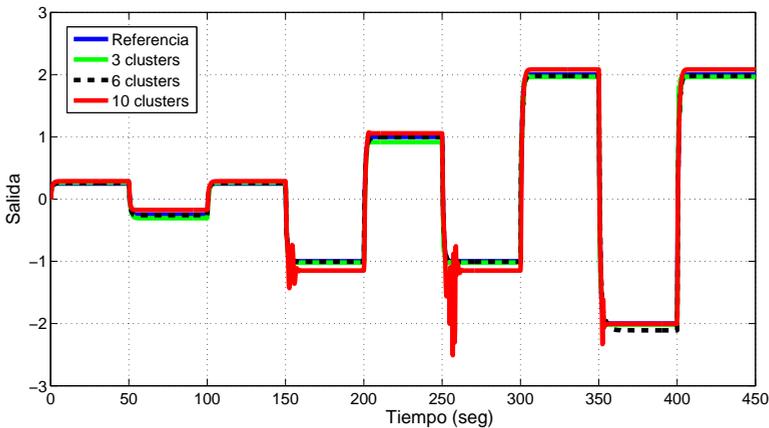


Figura 7.34: Respuestas para $T = 0.2$ s.

único que se va a modificar son los valores de las ganancias G_e , G_{de} y G_u . Con valores iniciales de $G_e = G_{de} = G_u = 1$ y observando la respuesta se modifican estos valores manualmente, observando que no es necesario del uso de un GA para conseguir buenos resultados. En la figura 7.38 se observa la respuesta obtenida para $G_e = 1.5$, $G_{de} = 10$ y $G_u = 1$. El error cometido y la señal de control necesaria se muestran en las figuras 7.39 y 7.40 respectivamente.

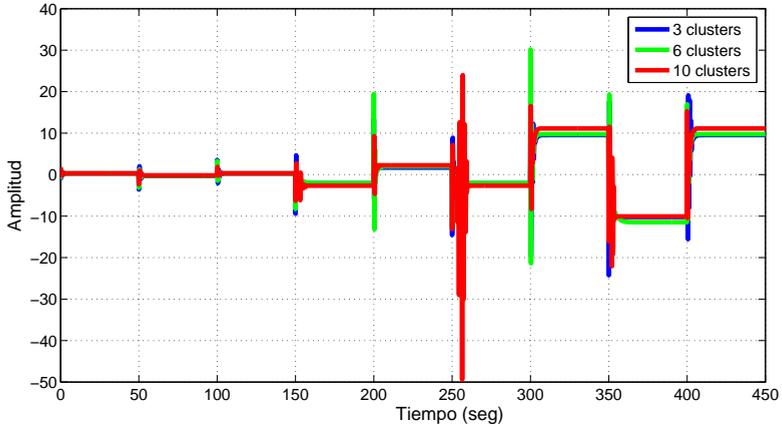
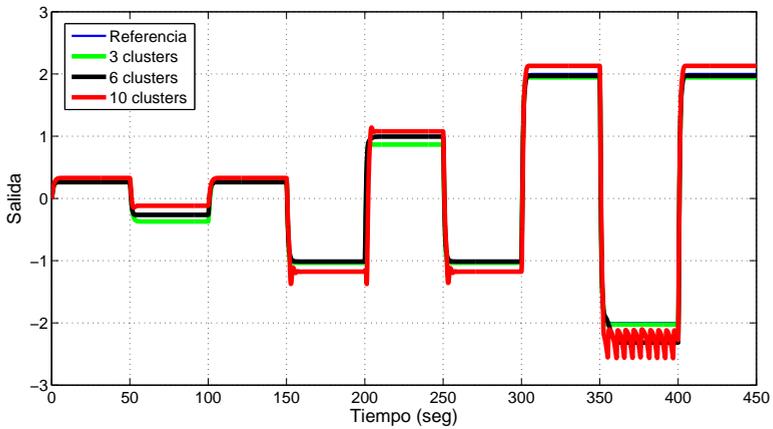
Figura 7.35: Señales de control para $T = 0.2$ s.

Figura 7.36: Respuestas para diferentes muestreos.

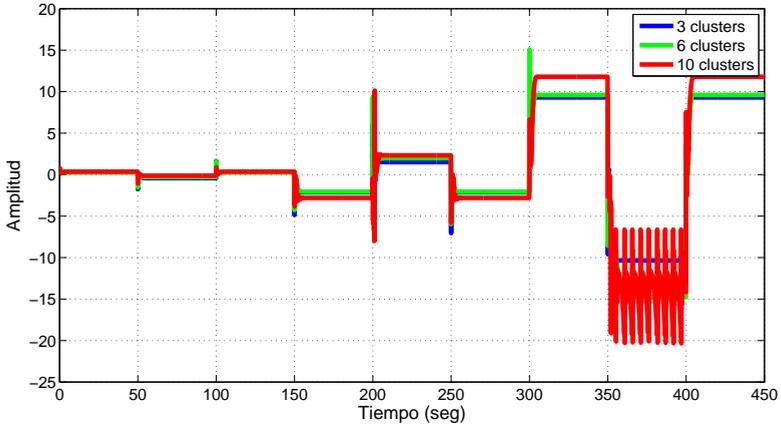


Figura 7.37: Señales de control para diferentes muestreos.

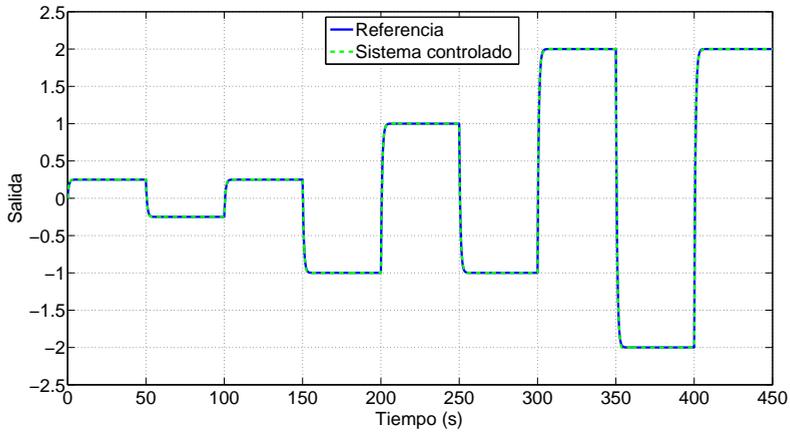


Figura 7.38: Respuesta para 3 clústers y PI borroso con $T = 0.2$ s.

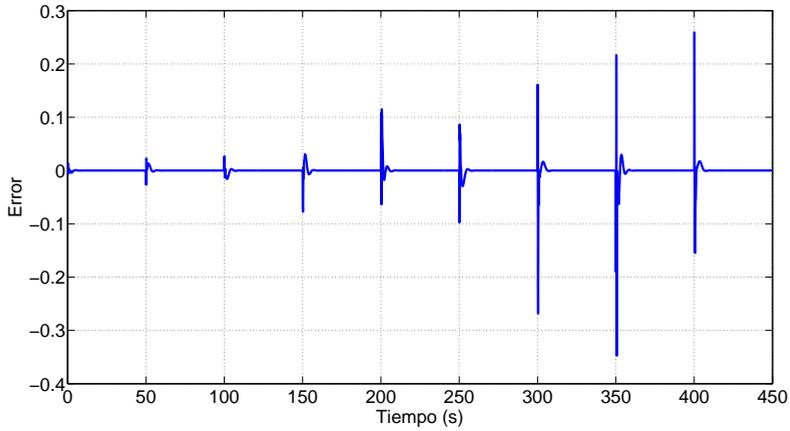


Figura 7.39: Error para 3 clústers y PI borroso con $T = 0.2$ s.

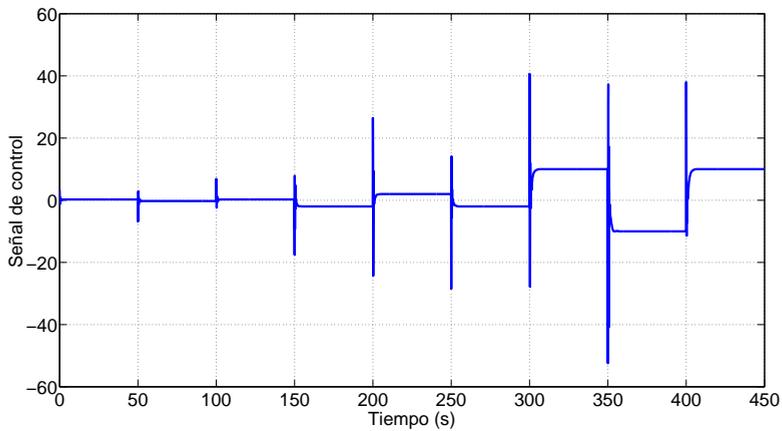


Figura 7.40: Señal de control para 3 clústers y PI borroso con $T = 0.2$ s.

Control del nivel de líquido en tanques cónicos

7.2.1. Introducción

El control de nivel de líquido y el flujo en tanques es un problema básico en las industrias de proceso. Los tanques cónicos encuentran amplias aplicaciones en tales industrias y han sido usados por muchos autores para testear las prestaciones de sus sistemas de control, ya que el tanque cónico con flujo por gravedad es un proceso que presenta una fuerte no linealidad.

Debido a que el proceso de nivel de un tanque cónico tiene características no lineales, normalmente se representa a la planta como varios modelos lineales para un determinado número de puntos operativos. De esta forma, se identifica el proceso mediante un número de funciones de transferencia de primer orden más tiempo de retardo —*First Order Plus Time Delay* (FOPTD)—, representando una aproximación al sistema real [172]. Desde este punto de vista, el control se enfoca sobre plantas FOPTD, también llamadas plantas de primer orden más tiempo muerto —*First Order Plus Dead Time* (FOPDT)— [173].

En [174], los autores identificaron cuatro funciones de transferencia de primer orden, con tiempo muerto igual a cero, en cuatro puntos operativos, mediante el método de la curva de reacción del proceso [175]. Ellos proponen un controlador PI [175] para cada función de transferencia usando el método de síntesis directa —*Direct Synthesis Method* (DSM)— [176] y el método de Skogestad [177]. Los resultados obtenidos mostraron el mejor comportamiento del DSM.

Además, usando el método de la curva de reacción del proceso, en [178] se diseña un controlador PI adaptativo [179] que se sintoniza de forma continua según el punto operativo del tanque. Para ello identificaron cinco modelos FOPTD para cinco puntos operativos. Los parámetros del modelo estimado se usan para calcular el controlador basado sobre los parámetros del controlador y sus puntos operativos, creándose un sistema de sintonización. Se hizo una comparación con un PI convencional, sintonizado por las reglas de Ziegler-Nichols [180], y los resultados mostraron las mejores prestaciones del controlador PI adaptativo.

En [181] realizan una identificación del sistema, usando la curva de reacción del proceso, para obtener cuatro modelos FOPTD para cuatro puntos operativos. Los autores propusieron el diseño de un control borroso —*Fuzzy Control* (FC)— [182], el cual se compara con un controlador PID [175]. El controlador PID está basado en la propuesta de Chien y Fruehauf [183], la

cual es una técnica basada en el control por modelo interno —*Internal Model Controller* (IMC)— [63]. Los resultados en [181] demuestran el mejor comportamiento del FC.

En [184] se propone un controlador PI borroso [157] y se compara con un PI sintonizado por las reglas de Ziegler-Nichols. Los resultados demuestran que el PI borroso presenta mejores prestaciones. El diseño de los controladores está basado en el modelo linealizado en un punto operativo.

Un esquema de FC basado en reglas se desarrolló en [185] y se comparó con un PI sintonizado por DSM. Los resultados son claramente a favor del FC.

La lógica borrosa también se aplica en [186] para diseñar un controlador PI adaptativo borroso [187]. En este tipo de controlador, la lógica borrosa varía los valores de los parámetros de un PI convencional. Los autores compararon las prestaciones del PI adaptativo borroso con un PI sintonizado por las reglas de Ziegler-Nichols y para ello identificaron tres regiones operativas. Los resultados demuestran las mejores prestaciones del PI adaptativo borroso.

En [188], los mismos autores que en [186], diseñan un controlador PI borroso basado en modelo de referencia y comparan las prestaciones con un control adaptativo por modelo de referencia [189]. De los resultados obtenidos se ve claramente que la estrategia propuesta del PI borroso tiene unas prestaciones superiores al control adaptativo.

El GA [30] es usado en [190] para diseñar un PID, un IMC y un controlador basado en función de transferencia. Tres funciones de transferencia FOPTD son identificadas por el método de la curva de reacción del proceso. Las prestaciones de los controladores propuestos se comparan con las de controladores sintonizados por las reglas de Ziegler-Nichols. El mejor controlador fue el IMC sintonizado por un GA.

El GA se usa en [191] para diseñar cuatro controladores PI para cuatro puntos operativos, igual que en [174], y son comparados con cuatro controladores PI diseñados por el método de programación de ganancias —*Gain Scheduling Method* (GSM)— [192]. Aunque GSM es una forma muy efectiva de controlar sistemas cuya dinámica cambia con el punto operativo, el PI diseñado mediante un GA se comportó mejor.

Se han aplicado más técnicas de control para tanques cónicos. Así, en [193] ha sido aplicado un control predictivo basado en modelo —*Model Predictive Control* (MPC)— [63] y se compara con un PID sintonizado por la técnica IMC. Como en casos anteriores, un modelo FOPTD fue identificado para diseñar los dos controladores. El MPC da mejores resultados que el PID.

En [194] un MPC, sintonizado por el método de Sridhar y Cooper [195], es propuesto y comparado con un controlador PI basado en el método de

Skogestad. El tanque es aproximado a un modelo FOPTD. Se observa que el MPC muestra una mejor capacidad de seguimiento que el PI.

También en [196] se usa un GA para sintonizar las funciones de pertenencia de las variables de entrada de un FC tipo Mamdani [197]. Los resultados del FC se comparan con un PI para evaluar sus prestaciones. El PI se diseña usando la técnica IMC. Los resultados obtenidos muestran que el FC se comporta mejor que el PI.

El uso de *Soft Computing* (SC) [198] es una alternativa para controlar sistemas no lineales. Así, en [199] se diseñan un FC, un PI borroso y un controlador sintonizado por un GA. Un PI, sintonizado por el método de Skogestad, se usa como comparativa. Los resultados demostraron que los controladores diseñados usando las técnicas SC trabajan mejor que el PI.

En [200] un PI, sintonizado por las reglas de Ziegler-Nichols, se compara con un controlador PI basado en la optimización por enjambre de partículas —*Particle Swarm Optimization* (PSO)— [201,202] y un PI de orden fraccional [203]. A partir del análisis, el PI basado en PSO y el PI de orden fraccional ofrecen mejores prestaciones que el PI sintonizado por las reglas de Ziegler-Nichols.

PSO es también propuesto y usado en [204] para sintonizar los parámetros de un PI y obtener un controlador óptimo. La eficacia del PI sintonizado por el algoritmo PSO fue validado mediante un estudio comparativo con un PI diseñado por el GSM. El PI sintonizado por PSO dio mejores resultados que el GSM.

En [4] se diseñan y comparan dos controladores predictivos generalizados —*Generalized Predictive Controller* (GPC)— [205,206]. También en [4] se introduce un control inverso directo [55]. Este control inverso directo fue implementado mediante una red neuronal [207]. Según los resultados obtenidos, se demuestra el mejor comportamiento que ofrece el control inverso directo.

En [208] se propone un sistema basado en la estrategia de control de tiempo óptimo [209] y un control adaptativo usando redes neuronales. El control de tiempo óptimo se usa en los cambios de consigna y el control adaptativo se diseña para variaciones en los parámetros del proceso. Se hace una comparativa con las prestaciones que ofrece un PID sintonizado por las reglas de Ziegler-Nichols, usando el método de la curva de reacción del proceso, para seis regiones. Los resultados demuestran la eficacia del control de tiempo óptimo y del control adaptativo.

Como se ha descrito anteriormente, la mayoría de los autores realizan una aproximación del proceso mediante un determinado número de funciones FOPTD y entonces aplican alguna técnica de control. Aquí se realizará una aproxima-

ción por medio de modelos lineales locales que se han obtenido de un modelo Takagi-Sugeno. Una vez se ha obtenido el modelo, se aplicará la técnica de control inverso directo por modelo de referencia para controlar el nivel en un tanque cónico. En [4], los autores aplican la técnica de control inverso, pero sin hacer una aproximación del modelo y entrenando una red neuronal como el inverso exacto. Sin embargo, el controlador obtenido en [4] es más complejo que la propuesta realizada aquí, aunque los resultados obtenidos son similares. También se realizará el diseño de un controlador predictivo no lineal —NGPC— para compararlo con la propuesta planteada aquí.

7.2.2. Modelado del tanque cónico

A continuación se obtendrá la ecuación diferencial no lineal que va a servir para modelar la variación del nivel de líquido en función de los parámetros físicos del tanque y de los flujos de entrada y salida.

La figura 7.41 ilustra la geometría del tanque cónico bajo estudio. Como se muestra, la forma es un cono y por ello la relación geométrica no es lineal.

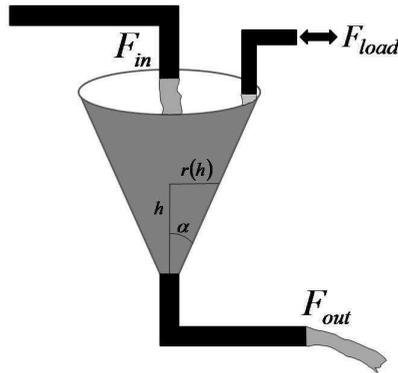


Figura 7.41: Tanque cónico.

Los parámetros del tanque cónico son: flujo de entrada (F_{in}), flujo de salida (F_{out}), carga aplicada al tanque (F_{load}), radio superior (R) y altura total (H). F_{load} es la perturbación que puede aparecer en el sistema [210].

El objetivo del control es mantener el nivel del líquido en un valor h constante, siendo el radio del cono r para este nivel h , y para ello se tiene que controlar el flujo de entrada F_{in} . Se asume que F_{in} se controla de forma directa y que un cambio en el flujo tiene efecto inmediato. El flujo de salida F_{out} es debido a la presión hidrostática y no está controlado.

El volumen de líquido V para un nivel h es dado en (7.10).

$$\left. \begin{aligned} V &= \frac{1}{3} \cdot \pi \cdot r^2 \cdot h \\ \tan \alpha &= \frac{R}{H} = \frac{r}{h} \end{aligned} \right\} V = \frac{1}{3} \cdot \pi \cdot h^3 \cdot \tan^2 \alpha = \frac{1}{3} \cdot \pi \cdot h^3 \cdot \left(\frac{R}{H} \right)^2 \quad (7.10)$$

Por simplicidad, se asume que el flujo es turbulento, el cual se caracteriza por trayectorias circulares erráticas semejantes a remolinos, sin importar la cantidad de líquido en el tanque. La variación del volumen es la diferencia entre el flujo de entrada y el flujo de salida, dependiendo del coeficiente de descarga k_{out} como se indica en (7.11).

$$\frac{dV}{dt} = F_{in} + F_{load} - F_{out} = F_{in} + F_{load} - k_{out} \cdot \sqrt{h} \quad (7.11)$$

Derivando la ecuación (7.10) se obtiene la (7.12).

$$\frac{dV}{dt} = [\pi \cdot \tan^2 \alpha] \cdot h^2 \cdot \frac{dh}{dt} \quad (7.12)$$

Igualando las ecuaciones (7.11) y (7.12) se obtiene la ecuación diferencial (7.13) que indica la variación del nivel de líquido h en función de los parámetros físicos del tanque y de los flujos de entrada y salida.

$$\left. \begin{aligned} \frac{dh}{dt} &= \frac{F_{in} + F_{load}}{\pi \cdot \tan^2 \alpha \cdot h^2} - \frac{k_{out}}{\pi \cdot \tan^2 \alpha \cdot h^{3/2}} \\ &= \frac{F_{in} + F_{load}}{\pi \cdot \left(\frac{R}{H}\right)^2 \cdot h^2} - \frac{k_{out}}{\pi \cdot \left(\frac{R}{H}\right)^2 \cdot h^{3/2}} \end{aligned} \right\} \quad (7.13)$$

Sección 7.3

Tanque cónico con descarga lenta

Para poder realizar comparativas con otras propuestas de diseño, se usará en esta sección el tanque cónico especificado en [4] (pp. 227-233). Las características de este tanque son: $0 \leq F_{in} \leq 4 \cdot 10^{-4} \text{ m}^3/\text{seg}$, $k_{out} = 1 \cdot 10^{-4} \text{ m}^{5/2}/\text{seg}$, $H = 0.5 \text{ m}$ y $\alpha = 20^\circ$.

7.3.1. Respuesta para diversas técnicas

En [4] se diseñan dos GPC, linealizando la planta en dos puntos operativos diferentes: $P_1 = (h_0, F_{in0}) = (0.03 \text{ m}, 1.73 \cdot 10^{-5} \text{ m}^3/\text{s})$ y $P_2 = (h_0, F_{in0}) = (0.2 \text{ m}, 4.47 \cdot 10^{-5} \text{ m}^3/\text{s})$. La figura 7.42 muestra la respuesta del GPC para el punto P_1 y la figura 7.43 para el punto P_2 . Como era de esperar, el primer

controlador trabaja bien para niveles pequeños, mientras que el segundo lo hace para niveles grandes.

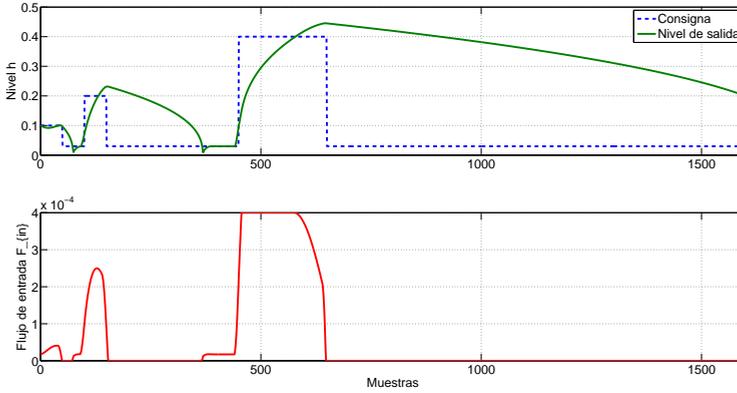


Figura 7.42: Respuesta del GPC para el punto P_1 .

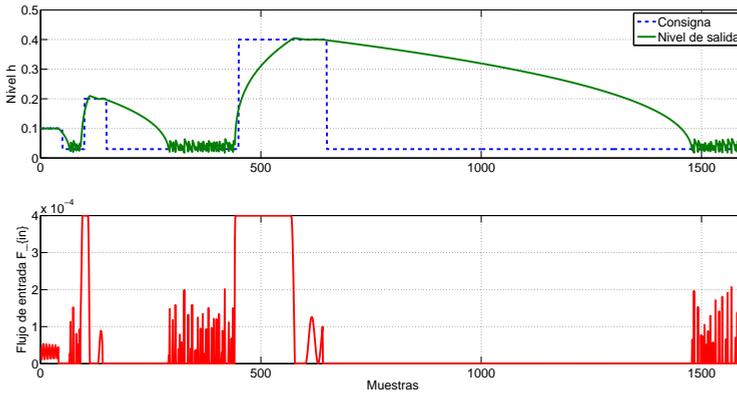


Figura 7.43: Respuesta del GPC para el punto P_2 .

En [4] también se diseña un control inverso directo. La respuesta obtenida (figura 7.44) tiene un mejor comportamiento, y es tan rápida como físicamente es posible, ya que cuando hay un cambio en la referencia, para llenar el tanque, la señal de control es máxima hasta casi conseguir la altura deseada. Para niveles bajos presenta un ligero error y un cierto rizado. El diseño de este control inverso directo fue realizado por el método de prueba y error, implementándose mediante una red neuronal con siete neuronas ocultas. La red fue

entrenada por el algoritmo de Levenberg-Marquardt [211] con 500 iteraciones y el modelo inverso fue usado como controlador.

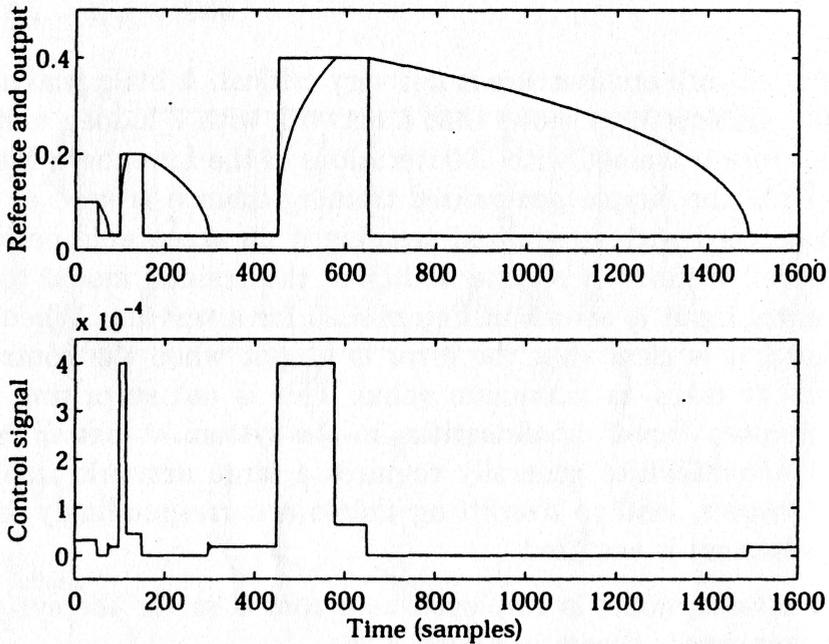


Figura 7.44: Respuesta con el control inverso directo (Fuente: [4]).

7.3.2. Identificación del tanque

Para proceder a la identificación se introduce el modelo de la ecuación diferencial (7.13) en *Simulink*[®] y se aplica al tanque flujos de entrada F_{in} variables, entre 0 y $4 \cdot 10^{-4} \text{ m}^3/\text{seg}$, para así obtener la altura h en cada instante (figura 7.45).

La secuencia del flujo de entrada F_{in} se ha adecuado para que el nivel de líquido no supere los 0.5 m de altura. Con un periodo de muestreo de $T = 0.1 \text{ s}$ se han obtenido los vectores de entrada y salida necesarios para aplicar el algoritmo de agrupamiento borroso. En el algoritmo de agrupamiento se comienza fijando los parámetros expuestos en la tabla 7.3. Ahora se hace un barrido desde 1 clúster hasta 10 clústers, no obteniéndose un resultado satisfactorio en el VAF obtenido. Por ello, se varía el parámetro $.Ny$ a 2 y se realiza el mismo proceso de variar el número de clústers desde 1 hasta 10. De esta forma se deduce que el número de clústers adecuado para este tipo de planta es de 4. Ningún otro número de clústers da una solución que se

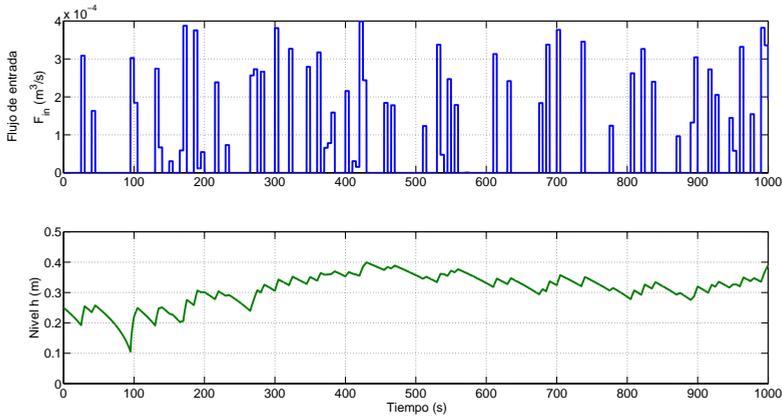


Figura 7.45: Señales para la identificación.

podría considerar como aceptable, ya que el VAF obtenido es muy bajo. Una vez establecido el número de clúster en 4 ($.c = 4$), se varía el coeficiente de borrosificación ($.m$) entorno al valor de 2. La evolución del VAF (figura 7.46) al variar el coeficiente de borrosificación ($.m$) da como resultado que el mejor $VAF = 99.7974\%$ en el proceso de validación (figura 7.47) se obtenga para un coeficiente $.m = 2.265$. La descripción del modelo borroso Takagi-Sugeno obtenido está dada en (7.14).

TABLA 7.3: PARÁMETROS DEL ALGORITMO DE AGRUPAMIENTO

Coficiente de borrosificación	$.m = 2$
Tolerancia de terminación	$.tol = 0.001$
Semilla para la inicialización	$.seed = 10$
Tipo de los antecedentes	$.ante = 2$
Estimación del consecuente	$.cons = 1$
Número de retardos de la salida	$.Ny = 1$
Número de retardos de la entrada	$.Nu = 1$
Número de retardos de transporte	$.Nd = 1$

Uno de los factores de los que depende la calidad obtenida del modelo matemático del proceso es la elección de las señales empleadas en la identificación del sistema. Para realizar una comparativa en el control que se obtendrá posteriormente, se ha generado otro modelo borroso añadiendo una pequeña señal aleatoria a la entrada (figura 7.48), de forma que se asegure que el modelo será

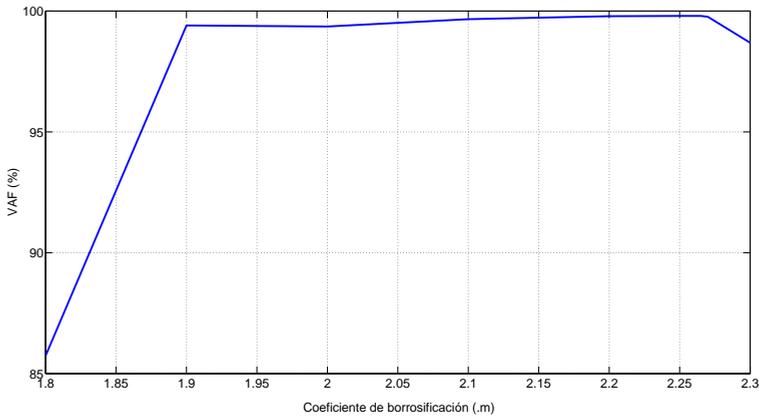


Figura 7.46: VAF al variar el .m con las señales de la figura 7.45.

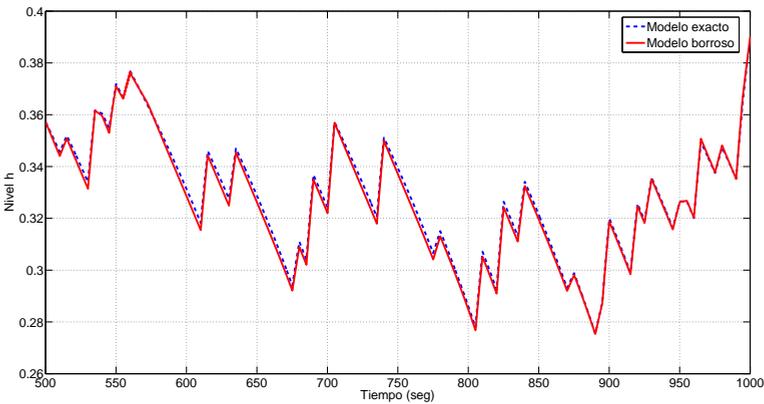


Figura 7.47: Resultado de la validación para el modelo (7.14).

capaz de reproducir de forma exacta salidas de alta frecuencia de magnitud pequeña. Siguiendo el mismo procedimiento explicado anteriormente, se obtienen los parámetros del algoritmo de agrupamiento dados en la tabla 7.4. En la figura 7.49 se puede ver la evolución del VAF al variar el coeficiente de borrosificación y en la figura 7.50 el resultado del proceso de validación. En (7.15) está la descripción del modelo borroso resultante con un $VAF = 96.3911\%$. Aunque el VAF obtenido en este segundo modelo es inferior al del primer modelo, sin embargo, y como se verá posteriormente, la calidad del control obtenido será mayor.

$$\left. \begin{aligned}
 R_1 : & \text{ If } y(k-1) \text{ is } A_{11} \text{ and } y(k-2) \text{ is } A_{12} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{13} \text{ then } y(k) = -1.61 \cdot 10^{-1}y(k-1) + \\
 & + 1.09 \cdot 10^0y(k-2) + 1.56 \cdot 10^1u(k-1) + 1.23 \cdot 10^{-2} \\
 R_2 : & \text{ If } y(k-1) \text{ is } A_{21} \text{ and } y(k-2) \text{ is } A_{22} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{23} \text{ then } y(k) = 8.84 \cdot 10^{-1}y(k-1) + \\
 & + 1.17 \cdot 10^{-1}y(k-2) + 5.07 \cdot 10^0u(k-1) - 3.74 \cdot 10^{-4} \\
 R_3 : & \text{ If } y(k-1) \text{ is } A_{31} \text{ and } y(k-2) \text{ is } A_{32} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{33} \text{ then } y(k) = 1.91 \cdot 10^0y(k-1) - \\
 & - 9.13 \cdot 10^{-1}y(k-2) + 1.46 \cdot 10^0u(k-1) - 2.30 \cdot 10^{-5} \\
 R_4 : & \text{ If } y(k-1) \text{ is } A_{41} \text{ and } y(k-2) \text{ is } A_{42} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{43} \text{ then } y(k) = -2.54 \cdot 10^{-2}y(k-1) + \\
 & + 1.02 \cdot 10^0y(k-2) + 3.15 \cdot 10^0u(k-1) + 4.77 \cdot 10^{-4}
 \end{aligned} \right\} \quad (7.14)$$

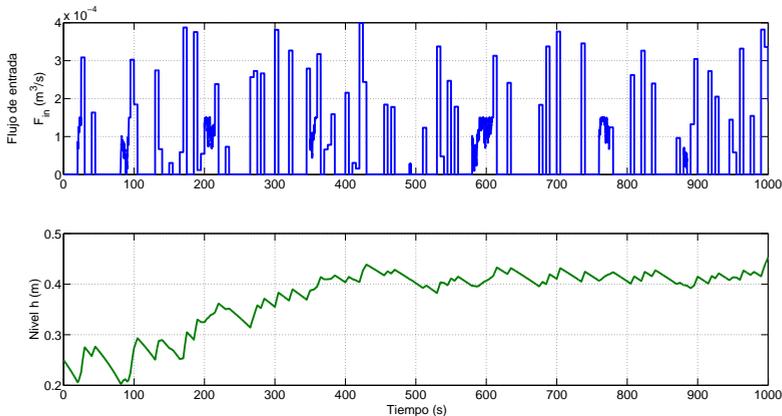


Figura 7.48: Señales para la identificación.

TABLA 7.4: PARÁMETROS DEL ALGORITMO DE AGRUPAMIENTO

Número de clústers	$.c = 4$
Coficiente de borrosificación	$.m = 2.51$
Tolerancia de terminación	$.tol = 0.001$
Semilla para la inicialización	$.seed = 10$
Tipo de los antecedentes	$.ante = 2$
Estimación del consecuente	$.cons = 1$
Número de retardos de la salida	$.Ny = 2$
Número de retardos de la entrada	$.Nu = 1$
Número de retardos de transporte	$.Nd = 1$

$$\left. \begin{aligned}
 R_1 : & \text{ If } y(k-1) \text{ is } A_{11} \text{ and } y(k-2) \text{ is } A_{12} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{13} \text{ then } y(k) = 1.44 \cdot 10^0 y(k-1) - \\
 & \quad - 4.41 \cdot 10^{-1} y(k-2) + 2.44 \cdot 10^0 u(k-1) - 9.10 \cdot 10^{-5} \\
 R_2 : & \text{ If } y(k-1) \text{ is } A_{21} \text{ and } y(k-2) \text{ is } A_{22} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{23} \text{ then } y(k) = 9.27 \cdot 10^{-1} y(k-1) + \\
 & \quad + 6.59 \cdot 10^{-2} y(k-2) + 3.88 \cdot 10^0 u(k-1) + 1.71 \cdot 10^{-3} \\
 R_3 : & \text{ If } y(k-1) \text{ is } A_{31} \text{ and } y(k-2) \text{ is } A_{32} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{33} \text{ then } y(k) = -1.25 \cdot 10^0 y(k-1) + \\
 & \quad + 2.25 \cdot 10^0 y(k-2) + 7.01 \cdot 10^0 u(k-1) - 1.82 \cdot 10^{-4} \\
 R_4 : & \text{ If } y(k-1) \text{ is } A_{41} \text{ and } y(k-2) \text{ is } A_{42} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{43} \text{ then } y(k) = 1.52 \cdot 10^0 y(k-1) - \\
 & \quad - 5.19 \cdot 10^{-1} y(k-2) + 8.02 \cdot 10^{-1} u(k-1) + 9.06 \cdot 10^{-5}
 \end{aligned} \right\} \quad (7.15)$$

7.3.3. Diseño del controlador inverso

Un factor que determinará la respuesta final del sistema, aparte del modelo Takagi-Sugeno obtenido, es la elección del modelo de referencia. Una estrategia que se puede usar es establecer el modelo de referencia como aquel que responda lo más rápido posible, o sea, un retardo puro (7.16), y posteriormente ir haciéndolo más lento.

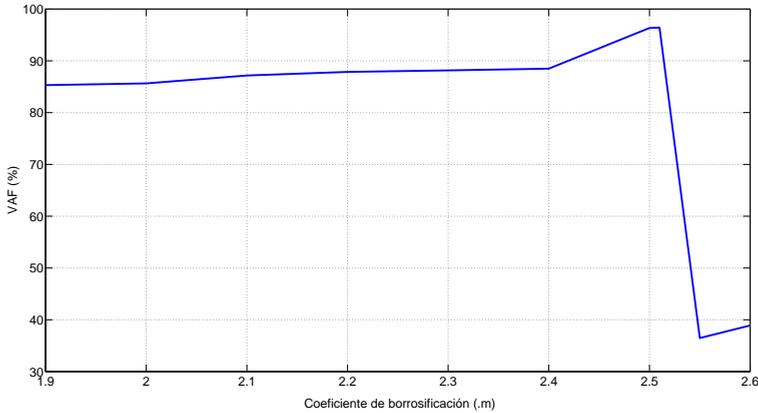


Figura 7.49: VAF al variar el .m con las señales de la figura 7.48.

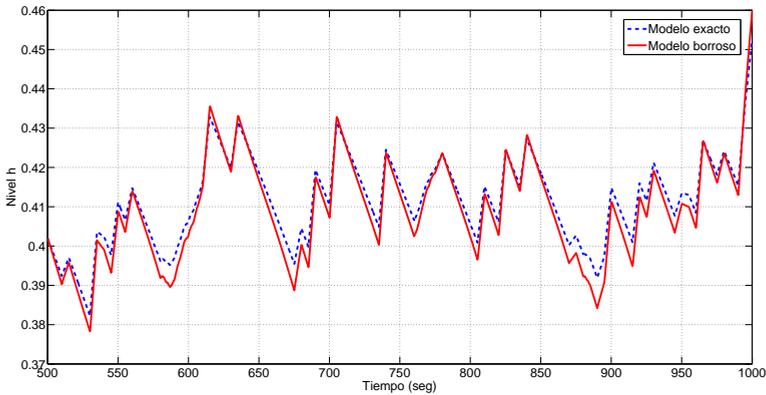


Figura 7.50: Resultado de la validación para el modelo (7.15).

$$G_r(z) = \frac{Y_r(z)}{R(z)} = \frac{1}{z} = z^{-1} \quad (7.16)$$

Hay que tener en cuenta que en algunos diseños el objetivo será conseguir la respuesta más rápida posible, sin sobreimpulsos, y no seguir exactamente la salida del modelo de referencia, siendo este el caso que nos ocupa.

Debido a que son cuatro reglas las que definen el modelo borroso del tanque cónico, como controlador saldrá una red neuronal con cuatro neuronas. Realizando la transformada inversa Z de $Y_r(z)$, en la ecuación (7.16), se obtiene la ecuación en diferencias (7.17) que define la salida $y_r(k)$ del modelo de

referencia.

$$y_r(k) = r(k - 1) \quad (7.17)$$

Igualando $y_r(k)$ de (7.17) a los consecuentes de (7.14) se obtiene la expresión de la señal de control $u(k)$ para cada neurona. Por ejemplo, para la regla R_1 se obtendría la expresión dada en (7.18).

$$u_1(k) = + \left. \begin{aligned} & \frac{1.61 \cdot 10^{-1}}{1.56 \cdot 10^1} \cdot y(k) + \frac{-1.09 \cdot 10^0}{1.56 \cdot 10^1} \cdot y(k - 1) + \\ & + \frac{1}{1.56 \cdot 10^1} \cdot r(k) \frac{-1.23 \cdot 10^{-2}}{1.56 \cdot 10^1} \end{aligned} \right\} \quad (7.18)$$

A continuación se diseña el sistema borroso de decisión el cual genera, a partir de las salidas de las cuatro neuronas, una única salida de control. Este punto fue explicado en la sección 7.1.5.

De la misma forma se procede con el segundo modelo borroso (7.15), generándose otro controlador diferente.

7.3.4. Resultados obtenidos

A continuación se muestran los resultados que se obtienen en el sistema controlado con cada uno de los dos controladores diseñados en la sección anterior.

Se introduce como consigna, o señal de referencia, la misma que en [4] (pp. 227-233). En la figura 7.51 se muestra la respuesta para el controlador diseñado con el modelo borroso (7.14) y en la figura 7.52 el resultado para el controlador diseñado con el modelo borroso (7.15). Ambos controladores trabajan con un periodo de muestreo $T = 1 \text{ ms}$.

Como se puede observar en la figura 7.51, para el modelo (7.14), la señal de control debe variar muy rápido para mantener estable los niveles alcanzados y esto en la práctica es un problema debido a la velocidad de apertura y cierre de la válvula que suministra el flujo de entrada. Analizando el sistema, se observa que este comportamiento de la señal de control es debido a que el sistema diseñado está conmutando entre dos de las reglas para mantener constante un nivel h . Sin embargo, para el diseño realizado con el modelo Takagi-Sugeno (7.15), figura 7.52, la señal de control es constante para mantener el líquido en un determinado nivel h . Esto indica que sólo una neurona actúa cuando se quiere mantener constante el nivel y el modelo borroso (7.15) es el adecuado para esta aplicación.

El resultado obtenido con el modelo (7.15) es similar al obtenido en [4] con el control inverso directo. Tanto en el llenado como en el vaciado no se puede

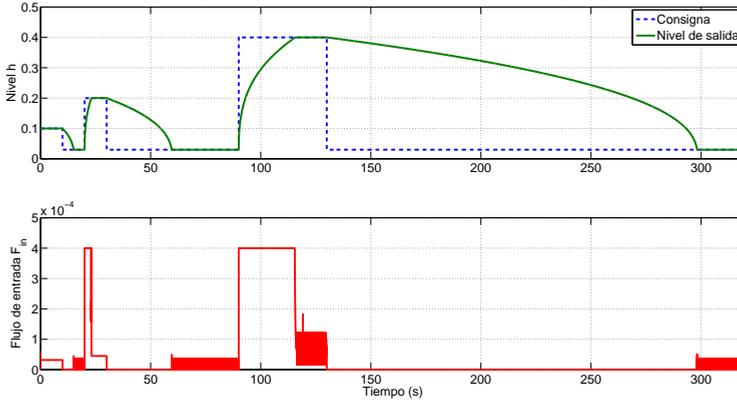


Figura 7.51: Respuesta con el modelo Takagi-Sugeno (7.14).

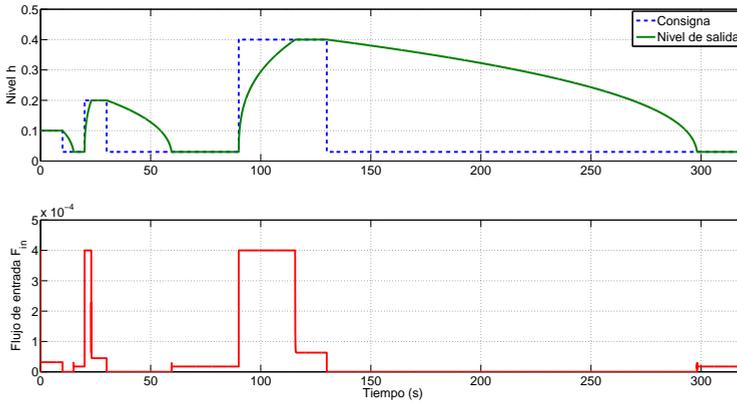


Figura 7.52: Respuesta con el modelo Takagi-Sugeno (7.15).

mejorar la respuesta, ya que la señal de control es máxima mientras se está llenando y nula cuando se está vaciando, por lo que la respuesta es tan rápida como físicamente es posible. Lo que si se mejora es la eliminación del ligero error y el rizado cuando el nivel de líquido es bajo.

Si se aumenta el periodo de muestreo a $T = 0.01$ s, la respuesta del sistema se degrada, produciéndose rizado en los niveles en los que se intenta mantener constante la altura del líquido en el tanque, tal como muestra la figura 7.53, con la consiguiente oscilación de la señal de control.

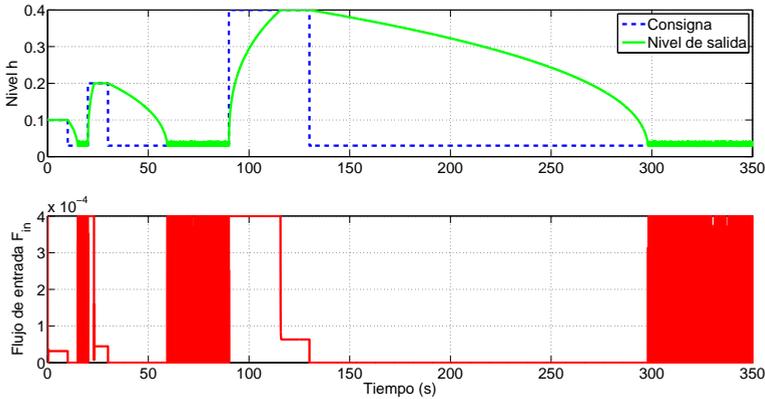


Figura 7.53: Respuesta con el modelo Takagi-Sugeno (7.15) y $T = 0.01$ s.

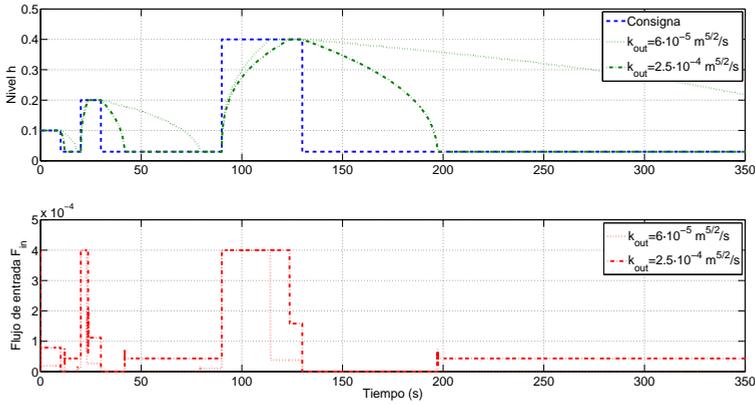
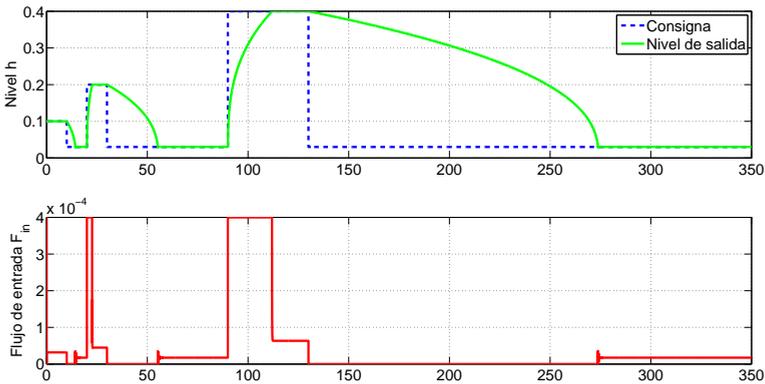
7.3.5. Estabilidad y robustez del sistema

En esta sección, se realiza un estudio sobre la estabilidad y robustez del sistema. En primer lugar se modifica el coeficiente de descarga k_{out} y, aunque los demás parámetros del tanque cónico son físicamente constantes o tienen una tolerancia en la fabricación muy pequeña, en segundo lugar se analiza el sistema cuando se cambia el radio superior (R) y/o la altura total (H). De esta forma se conocerá si el sistema diseñado es válido para tanques con unas dimensiones fijas o se puede aplicar a tanques de diferentes dimensiones.

La figura 7.54 muestra el comportamiento del sistema para variaciones de k_{out} entre $6 \cdot 10^{-5} \text{ m}^{5/2}/\text{s}$ y $2.5 \cdot 10^{-4} \text{ m}^{5/2}/\text{s}$. El sistema sigue siendo estable y robusto, manteniéndose la señal de control ideal sin oscilaciones. Si se varía más el valor de k_{out} , la señal de control comienza a tener oscilaciones aunque el sistema sigue siendo estable.

Ahora se varían las dimensiones del tanque con el objetivo de comprobar que el sistema diseñado es válido para un margen de tamaños de tanques cónicos. Dado que $\tan \alpha = \frac{R}{H}$, aumentar el ángulo α implica aumentar R o disminuir H , y disminuir α supone disminuir R o aumentar H . En las simulaciones realizadas, se puede disminuir el ángulo α hasta en -1.4° sin que la señal de control comience a oscilar para mantener niveles constantes (figura 7.55). Al ir disminuyendo el ángulo α , se va produciendo un rizado cada vez mayor para mantener niveles pequeños constantes. Por ejemplo, la figura 7.56 muestra la respuesta cuando se disminuye el ángulo α en -10° . El sistema permanece estable hasta una disminución de -17° en α .

El sistema se presenta más robusto ante aumentos positivos de α . Por ejem-

Figura 7.54: Respuesta al variar k_{out} .Figura 7.55: Respuesta al variar α en -1.4° .

plo, en la figura 7.57 se muestra el comportamiento ante una variación de $+4^\circ$. Como se ve, aparte de tardar más en el llenado y vaciado, normal ya que se ha aumentado el volumen del tanque, la señal de control es la adecuada. La figura 7.58 muestra la respuesta del sistema para $\alpha = 40^\circ$ y $\alpha = 60^\circ$. Como se puede ver en la figura el sistema permanece estable y robusto. Incluso variando α a un ángulo ligeramente inferior a 90° el sistema responde perfectamente.

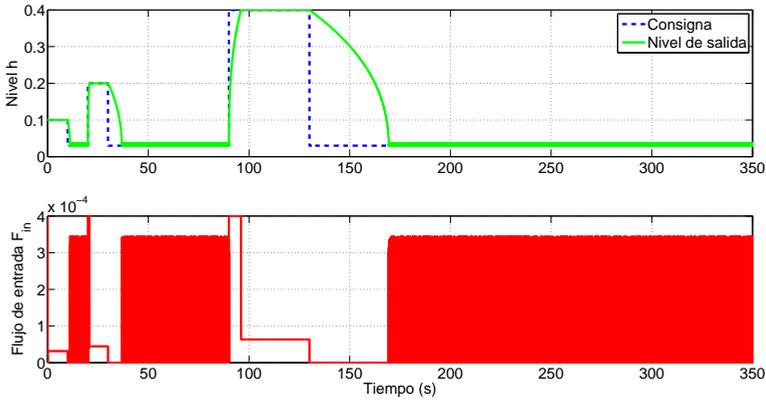


Figura 7.56: Respuesta al variar α en -10° .

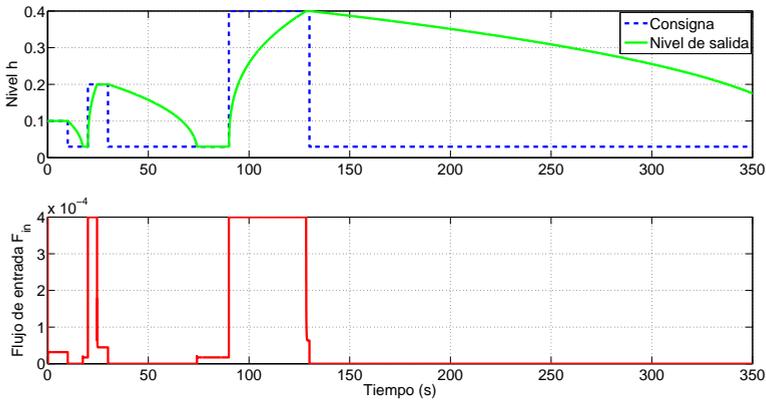
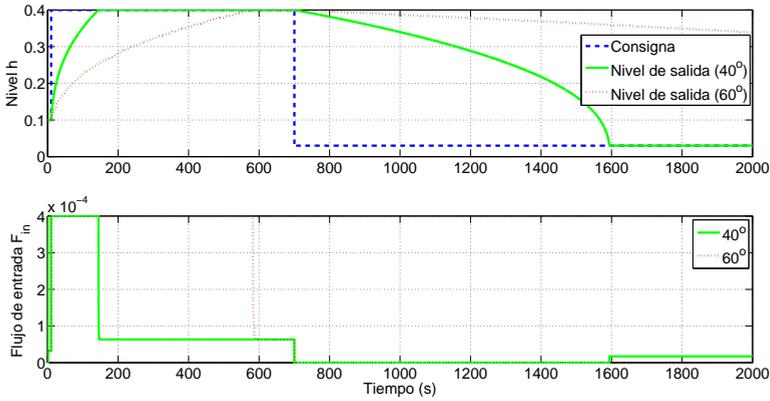
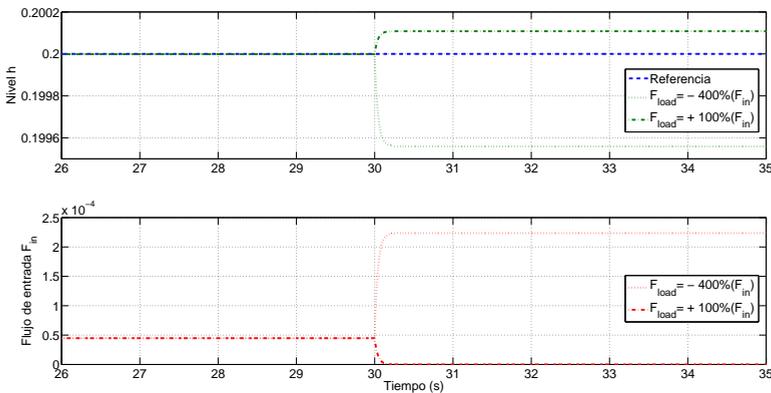


Figura 7.57: Respuesta al variar α en $+4^\circ$.

7.3.6. Comportamiento frente a perturbaciones

Ahora se verá el efecto de las perturbaciones en el sistema diseñado en los puntos operativos $h = 0.2 \text{ m}$ y $h = 0.4 \text{ m}$. En el instante $t = 30 \text{ s}$ se aplica una carga en forma de escalón entorno al flujo de entrada F_{in} . Para el nivel $h = 0.2 \text{ m}$ (figura 7.59) el sistema permanece estable si se varía la carga entre el -400% de F_{in} ($F_{load} = -1.7888 \cdot 10^{-4} \text{ m}^3/\text{s}$) y el $+100\%$ de F_{in} ($F_{load} = +0.4472 \cdot 10^{-4} \text{ m}^3/\text{s}$). Para el nivel $h = 0.4 \text{ m}$ (figura 7.60) el sistema permanece estable si se varía la carga entre el -500% de F_{in} ($F_{load} = -3.1623 \cdot 10^{-4} \text{ m}^3/\text{s}$) y el $+100\%$ de F_{in} ($F_{load} = +0.6325 \cdot 10^{-4} \text{ m}^3/\text{s}$).

Figura 7.58: Respuesta al variar α a 40° y 60° .Figura 7.59: Respuesta para perturbación entorno a $h = 0.2m$.

Como se puede observar en las figuras 7.59 y 7.60, el nivel h varía mínimamente cuando aparece la perturbación. Con esto, queda demostrado que el sistema de control propuesto rechaza a las perturbaciones de carga en un amplio rango.

7.3.7. Diseño de un controlador predictivo no lineal

Usando el paquete de herramientas de identificación de sistemas basado en redes neuronales (*Neural Network Based System Identification Toolbox*) [212] y el kit de herramientas de diseño de sistemas de control basado en

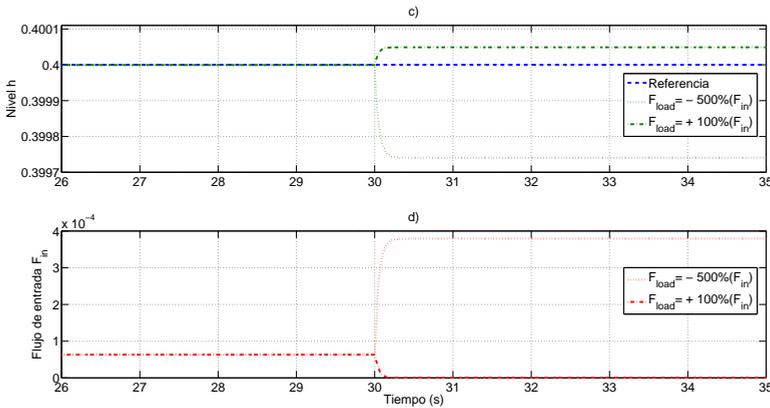


Figura 7.60: Respuesta para perturbación entorno a $h = 0.4m$.

redes neuronales (*Neural Network Based Control System Design Toolkit*) [213], desarrollados por Magnus NØrgaard, se ha diseñado un control predictivo no lineal —*Nonlinear Generalized Predictive Control* (NGPC).

Para el proceso de identificación se ha usado el mismo conjunto de datos que se utilizó para la determinación del modelo Takagi-Sugeno y que se presentó en la figura 7.48. En primer lugar, se ha de determinar el orden del modelo, es decir, el número de señales anteriores o pasadas usadas como regresores. Si no hay ruido o este es pequeño, el paquete de herramientas [212] proporciona un método para determinar el orden del modelo. La figura 7.61 muestra el resultado de la simulación, para órdenes desde 1 a 10, para el mismo número de retardos en la entrada y salida. En la figura 7.62 se visualiza el índice del orden frente al espacio de retardos de la entrada y salida.

El índice del orden es el número de Lipschitz el cual está basado en los coeficientes de Lipschitz [214]. El objetivo es que el número de Lipschitz sea tan pequeño como sea posible. En la figura 7.61 se puede observar que para 6 retrasos hay un mínimo, sin embargo, para 2 retrasos el índice del orden no es mucho mayor, con la ventaja de que la arquitectura de la red es mucho más simple. Por ello, se eligen 2 retrasos para la entrada y la salida. En la tabla 7.5 se puede ver el valor del índice del orden para todas las combinaciones de los retrasos de la entrada y la salida, m y n respectivamente, correspondiente a la figura 7.62.

Una vez que se ha elegido el orden del modelo, se debe determinar, por el método de prueba y error, la arquitectura de la red. Para las neuronas de la capa oculta se elegirá la función de activación tangente hiperbólica y para las

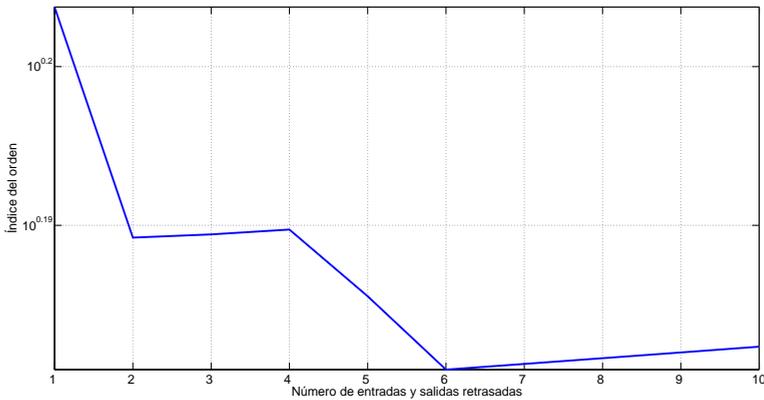


Figura 7.61: Índice del orden para retrasos de entrada y salida iguales.

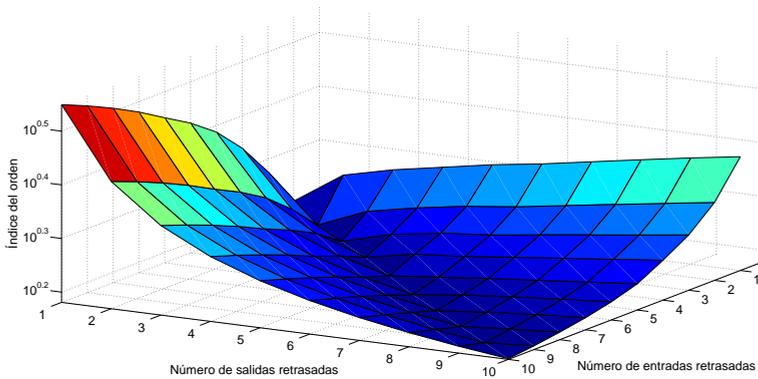


Figura 7.62: Índice del orden para diferentes retrasos de entrada y salida.

neuronas de la capa de salida la función de activación lineal. Se investigarán los resultados desde 1 hasta 10 neuronas en la capa oculta. Las tablas 7.6 y 7.7 muestran la suma normalizada de los errores al cuadrado —*Normalized Sum of Squared Errors* (NSSE)— que se han obtenido en el proceso de validación desde 4 a 10 neuronas, ya que los resultados desde 1 a 3 neuronas han sido muy malos. El entrenamiento de la red ha sido realizado por el método de Levenberg-Marquardt [211].

Si se analizan las tablas 7.6 y 7.7, se puede deducir que la arquitectura no lineal autorregresiva con entrada externa —*Nonlinear AutoRegressive eXternal input* (NARX)— es mejor que el modelo con error de la salida no lineal —

TABLA 7.5: ÍNDICE DEL ORDEN PARA DIFERENTES RETRASOS m Y n

m \ n	1	2	3	4	5	6	7	8	9	10
1	1.59	1.84	1.93	2.01	2.08	2.15	2.23	2.31	2.39	2.47
2	1.89	1.55	1.69	1.76	1.82	1.87	1.93	1.99	2.05	2.12
3	2.18	1.73	1.55	1.64	1.69	1.72	1.77	1.82	1.87	1.92
4	2.44	1.89	1.67	1.55	1.61	1.64	1.68	1.72	1.76	1.80
5	2.65	2.02	1.77	1.62	1.53	1.57	1.60	1.63	1.66	1.70
6	2.83	2.14	1.85	1.69	1.59	1.52	1.56	1.59	1.61	1.64
7	3.02	2.27	1.95	1.78	1.66	1.58	1.52	1.55	1.57	1.60
8	3.21	2.39	2.05	1.85	1.73	1.64	1.57	1.52	1.54	1.57
9	3.38	2.51	2.14	1.93	1.79	1.70	1.62	1.57	1.52	1.54
10	3.54	2.62	2.23	2.00	1.86	1.75	1.67	1.61	1.56	1.52

TABLA 7.6: NSSE PARA NOE Y DIFERENTE NÚMERO DE ITERACIONES

Neuronas \ Iteraciones	Iteraciones		
	500	1000	2000
4	$1.07 \cdot 10^{-7}$	$2.94 \cdot 10^{-5}$	$1.18 \cdot 10^{-4}$
5	$5.29 \cdot 10^{-9}$	$1.02 \cdot 10^{-8}$	$1.17 \cdot 10^{-8}$
6	$3.05 \cdot 10^{-8}$	$6.24 \cdot 10^{-9}$	$2.54 \cdot 10^{-9}$
7	$8.02 \cdot 10^{-8}$	$8.39 \cdot 10^{-10}$	$2.05 \cdot 10^{-9}$
8	$2.74 \cdot 10^{-8}$	$3.21 \cdot 10^{-10}$	$3.24 \cdot 10^{-10}$
9	$2.78 \cdot 10^{-8}$	$1.15 \cdot 10^{-8}$	$9.74 \cdot 10^{-9}$
10	$2.06 \cdot 10^{-4}$	$1.06 \cdot 10^{-8}$	$3.50 \cdot 10^{-9}$

Nonlinear Output Error (NOE). Concretamente, para 6 neuronas en la capa oculta, y con 1000 iteraciones, el NSSE es mínimo e igual a $2.01 \cdot 10^{-12}$. La figura 7.63 muestra el proceso de validación para 6 neuronas y el error de predicción obtenido.

Una vez que se ha definido el modelo de predicción de la planta, se diseñará el algoritmo de control predictivo. En primer lugar se elige un periodo de muestreo $T = 0.01$ s. Además, se deben fijar los parámetros del algoritmo de predicción, es decir, el horizonte de predicción N , el horizonte de control N_u y el parámetro λ que pondera el esfuerzo de control. La figura 7.64 muestra la respuesta para 6 neuronas, $N = 7$, $N_u = 5$ y $\lambda = 0.01$.

Se han variado los valores de predicción (N , N_u y λ) y también se han usado

TABLA 7.7: NSSE PARA NARX Y DIFERENTE NÚMERO DE ITERACIONES

Neuronas \ Iteraciones	500	1000	2000
	4	$2.56 \cdot 10^{-12}$	$2.20 \cdot 10^{-12}$
5	$2.54 \cdot 10^{-12}$	$4.31 \cdot 10^{-12}$	$1.20 \cdot 10^{-11}$
6	$2.33 \cdot 10^{-12}$	$2.01 \cdot 10^{-12}$	$2.01 \cdot 10^{-12}$
7	$3.94 \cdot 10^{-12}$	$2.50 \cdot 10^{-12}$	$2.26 \cdot 10^{-12}$
8	$2.34 \cdot 10^{-12}$	$2.42 \cdot 10^{-12}$	$2.10 \cdot 10^{-12}$
9	$6.47 \cdot 10^{-12}$	$2.18 \cdot 10^{-12}$	$2.10 \cdot 10^{-12}$
10	$2.83 \cdot 10^{-12}$	$3.59 \cdot 10^{-12}$	$2.21 \cdot 10^{-12}$

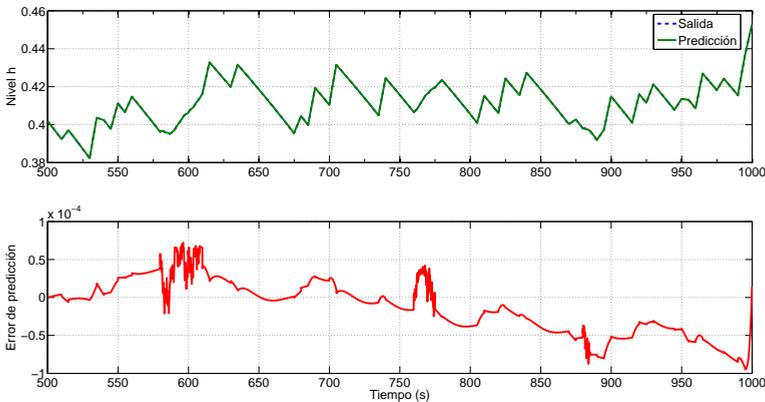


Figura 7.63: Validación para 6 neuronas y 1000 iteraciones.

diferente número de neuronas, pero no se han obtenido mejores respuestas. Por ejemplo, en la figura 7.65 se puede ver la respuesta para 9 neuronas, entrenadas con 2000 iteraciones, $N = 30$, $N_u = 20$ y $\lambda = 0.1$. La mejor respuesta obtenida es la mostrada en la figura 7.64 y como se puede observar, para mantener estables los niveles alcanzados, la señal de control debe variar muy rápido, y en la práctica esto es un problema debido a la velocidad de apertura y cierre de la válvula que suministra el flujo de entrada.

Para mejorar la respuesta del sistema es necesario disminuir el periodo de muestreo. Si se elige el periodo de muestreo $T = 1 \text{ ms}$, con 6 neuronas, $N = 7$, $N_u = 5$ y $\lambda = 0.01$, se obtiene la respuesta mostrada en la figura 7.66. Aquí, existe un ligero error en régimen permanente y se produce un rizado de 0.0024 m en el nivel de 0.3 m .

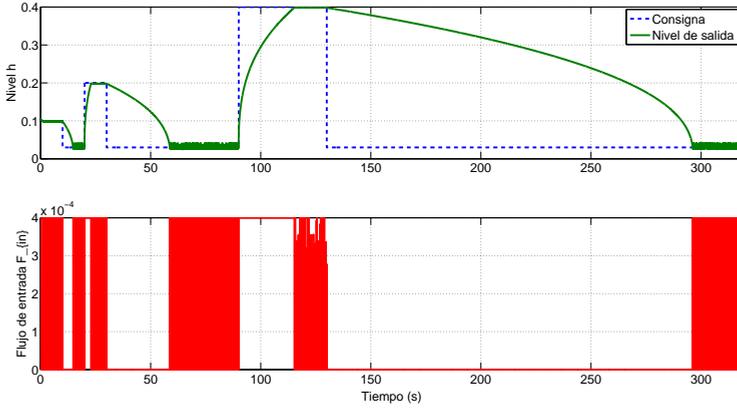


Figura 7.64: Respuesta con 6 neuronas, $N = 7$, $N_u = 5$ y $\lambda = 0.01$.

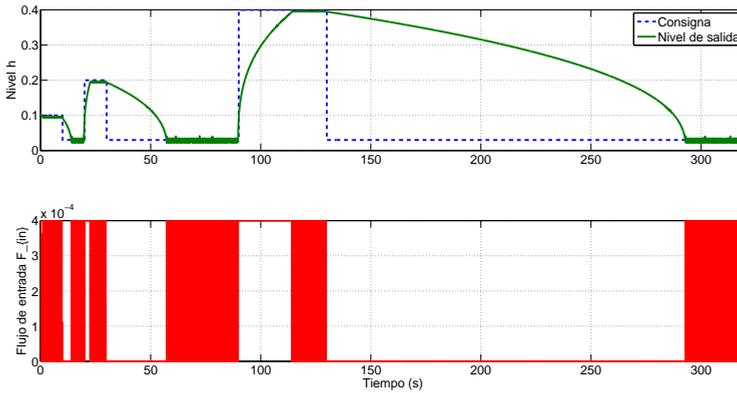


Figura 7.65: Respuesta con 9 neuronas, $N = 30$, $N_u = 20$ y $\lambda = 0.1$.

Comparando las respuestas, se ve claramente que la del control inverso con modelo de referencia propuesto es mejor que la que ofrece el NGPC. Además, el costo computacional y el hardware asociado sería de mayor coste para el NGPC.

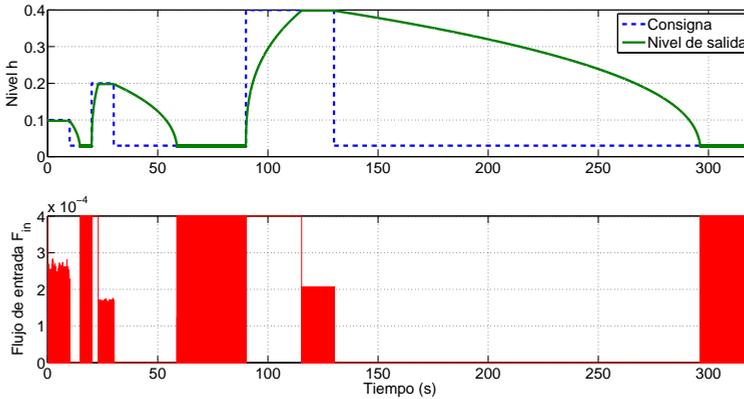


Figura 7.66: Respuesta con 6 neuronas y $T = 1$ ms.

Sección 7.4

Tanque cónico con descarga rápida

Para poder realizar comparativas con otras estrategias de control para este tipo de tanque, se usará el tanque cónico especificado en [174]. Las características de este tanque son: $0 \leq F_{in} \leq 400$ cm³/seg, $k_{out} = 55$ cm^{5/2}/seg, $H = 73$ cm y $R = 19.25$ cm.

7.4.1. Respuesta con controladores PI

En [174] los autores identificaron un modelo de primer orden para cuatro puntos operativos diferentes: 1.44 cm, 5.76 cm, 12.83 cm y 23.04 cm de altura. Para cada punto operativo, diseñaron un controlador PI por el método de síntesis directa [176] y por el método de Skogestad [177]. La figura 7.67 muestra los resultados obtenidos.

7.4.2. Identificación del tanque

Para proceder a la identificación, igual que en el ejemplo anterior, se introduce el modelo de la ecuación diferencial (7.13) en *Simulink*[®], aplicándose flujos de entrada variables para obtener la altura del líquido en cada instante. Una vez sintonizados los parámetros del algoritmo de agrupamiento quedan con los valores reflejados en la tabla 7.8. El VAF obtenido en el proceso de validación ha sido del 98.6315% y el modelo borroso Takagi-Sugeno el dado en (7.19).

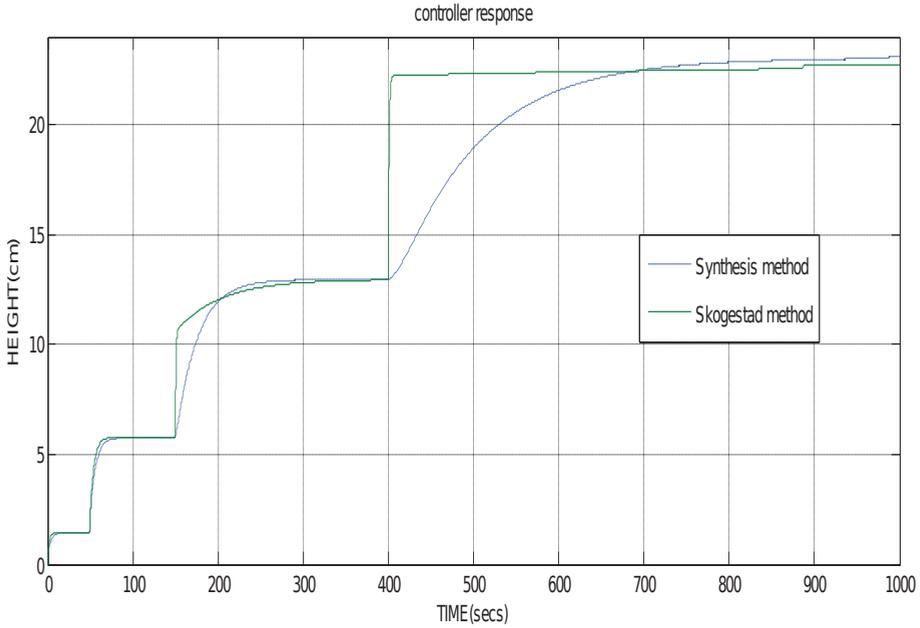


Figura 7.67: Respuesta con los dos PI (Fuente: [174]).

$$\left. \begin{aligned}
 R_1 : & \text{ If } y(k-1) \text{ is } A_{11} \text{ and } y(k-2) \text{ is } A_{12} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{13} \text{ then } y(k) = 1.55 \cdot 10^0 y(k-1) - \\
 & \quad - 5.48 \cdot 10^{-1} y(k-2) + 2.77 \cdot 10^{-4} u(k-1) - 3.21 \cdot 10^{-2} \\
 R_2 : & \text{ If } y(k-1) \text{ is } A_{21} \text{ and } y(k-2) \text{ is } A_{22} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{23} \text{ then } y(k) = 4.47 \cdot 10^{-1} y(k-1) - \\
 & \quad - 4.05 \cdot 10^{-1} y(k-2) + 7.57 \cdot 10^{-2} u(k-1) + 4.03 \cdot 10^1 \\
 R_3 : & \text{ If } y(k-1) \text{ is } A_{31} \text{ and } y(k-2) \text{ is } A_{32} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{33} \text{ then } y(k) = -6.69 \cdot 10^{-1} y(k-1) + \\
 & \quad + 1.67 \cdot 10^0 y(k-2) + 4.69 \cdot 10^{-4} u(k-1) - 5.28 \cdot 10^{-2} \\
 R_4 : & \text{ If } y(k-1) \text{ is } A_{41} \text{ and } y(k-2) \text{ is } A_{42} \text{ and} \\
 & \text{ and } u(k-1) \text{ is } A_{43} \text{ then } y(k) = 1.77 \cdot 10^0 y(k-1) - \\
 & \quad - 7.72 \cdot 10^{-1} y(k-2) + 3.40 \cdot 10^{-5} u(k-1) - 1.38 \cdot 10^{-2}
 \end{aligned} \right\} \quad (7.19)$$

TABLA 7.8: PARÁMETROS DEL ALGORITMO DE AGRUPAMIENTO

Número de clústers	$.c = 4$
Coficiente de borrosificación	$.m = 2.215$
Tolerancia de terminación	$.tol = 0.001$
Semilla para la inicialización	$.seed = 10$
Tipo de los antecedentes	$.ante = 2$
Estimación del consecuente	$.cons = 1$
Número de retardos de la salida	$.Ny = 2$
Número de retardos de la entrada	$.Nu = 1$
Número de retardos de transporte	$.Nd = 1$

7.4.3. Diseño del controlador inverso

Igual que en el ejemplo de la sección 7.3.3 se establece como modelo de referencia el equivalente a un retardo puro (7.16), con el cual se conseguirá una señal de control ideal para la aplicación en cuestión.

Con el modelo borroso Takagi-Sugeno obtenido (7.19) y el modelo de referencia (7.16) se diseña la red neuronal, que actuará como control inverso directo, siguiendo el mismo procedimiento al señalado en la sección 7.3.3. También se diseña el sistema borroso de decisión el cual genera a partir de las cuatro salidas de control una única salida. Este punto fue explicado en la sección 7.1.5.

Una vez diseñado el controlador, se realizarán las simulaciones pertinentes y analizarán los resultados obtenidos, decidiendo lo que se habría que modificar en el diseño si los resultados no fueran los deseados.

7.4.4. Resultados obtenidos

A continuación se muestran los resultados obtenidos con el controlador diseñado y se comparan con los obtenidos en [174] utilizando la misma consigna.

La figura 7.68 muestra la respuesta obtenida y tal como se puede observar, el controlador diseñado tiene un comportamiento con una respuesta mucha más rápida, comparada con la obtenida por el método de síntesis directa y por el método de Skogestad (figura 7.67). Ello es debido a que el controlador inverso impone el máximo flujo de líquido posible de entrada cuando hay un cambio en la consigna, mientras que los otros controles no usan ese flujo máximo de entrada y además el cambio de flujo de entrada es gradual. La figura 7.68 muestra también que el tiempo de llenado es tan rápido como físicamente es

posible, no pudiéndose mejorar, ya que el flujo de entrada es máximo cuando el tanque se está llenando, hasta que alcanza el nivel deseado, momento en el cual la señal de control es constante para mantener el líquido en el nivel deseado.

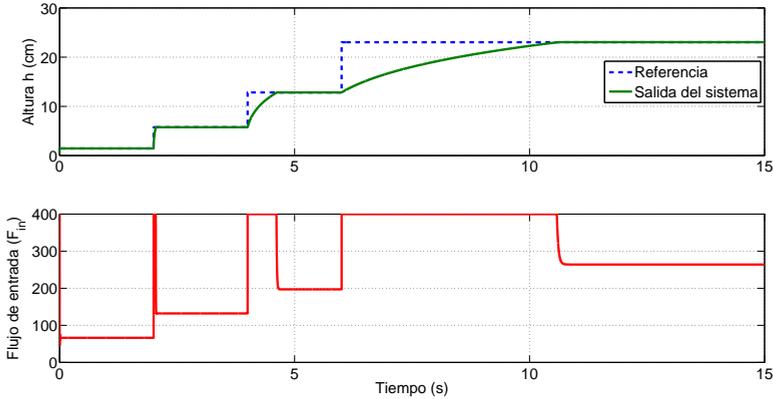


Figura 7.68: Respuesta del control inverso con modelo de referencia.

En la tabla 7.9 se resume el tiempo de establecimiento en segundos, el cual es medido entorno al $\pm 2\%$ del valor final, para las gráficas de las figuras 7.67 y 7.68.

TABLA 7.9: TIEMPO DE ESTABLECIMIENTO EN SEGUNDOS

Altura (cm)	Síntesis directa	Método de Skogestad	Control inverso directo
0.2 → 1.44	9.00	4.50	0.02
1.44 → 5.76	31.00	20.00	0.05
5.76 → 12.83	137.00	226.00	0.60
12.83 → 23.04	600.00	>600.00	4.44

7.4.5. Estabilidad y robustez del sistema

Seguidamente se realiza un estudio sobre la estabilidad y robustez del sistema al variar el coeficiente de descarga k_{out} y las dimensiones del tanque.

En la figura 7.69 se muestra el comportamiento del sistema cuando se varía k_{out} en un $\pm 50\%$ de su valor de $55 \text{ cm}^{5/2}/\text{s}$, es decir, entre $27.5 \text{ cm}^{5/2}/\text{s}$ y

$82.5 \text{ cm}^5/2/s$. Como se puede ver, el sistema sigue siendo estable y robusto, manteniéndose la señal de control ideal sin oscilaciones. Incluso se puede variar más el valor de k_{out} y el sistema sigue siendo estable y se comporta de forma ideal. En la figura 7.70 se ve el comportamiento para $k_{out} = 0$, es decir, sin descarga y para $k_{out} = 100 \text{ cm}^5/2/s$.

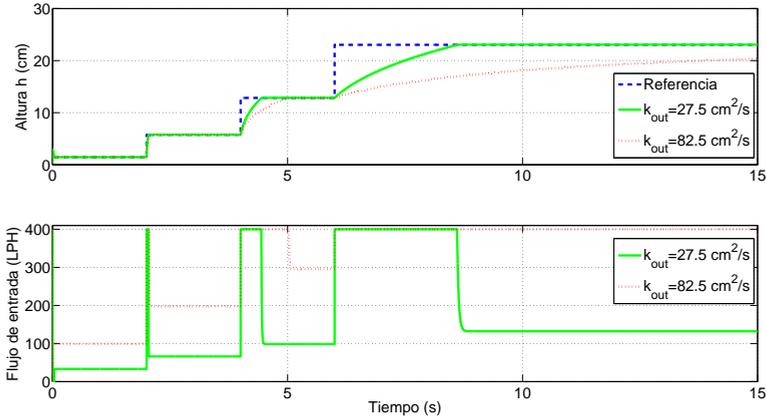


Figura 7.69: Respuesta al variar k_{out} un $\pm 50\%$.

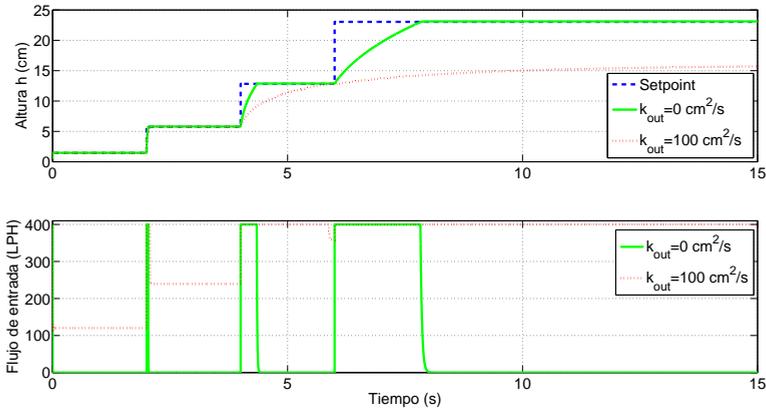


Figura 7.70: Respuesta al variar k_{out} entre 0 y $100 \text{ cm}^2/s$.

Ahora se varían las dimensiones del tanque. Ante la disminución de la altura H , o el aumento del radio R , el sistema sigue respondiendo perfectamente. Por ejemplo, en la figura 7.71 se puede observar la respuesta ante una disminución

de la altura H en -55 cm y ante un aumento del radio R en $+50\text{ cm}$ ($H = 18\text{ cm}$, $R = 69.25\text{ cm}$). La figura 7.72 muestra la respuesta cuando se aumenta la altura H en $+55\text{ cm}$ y ante una disminución del radio R en -10 cm ($H = 128\text{ cm}$, $R = 9.25\text{ cm}$). Tal como se observa, la respuesta del sistema sigue siendo ideal. Si se disminuye más el radio R , comienza a aparecer rizado ante niveles bajos del nivel. Según los resultados obtenidos, se verifica que el sistema diseñado es robusto para un amplio margen de tamaños del tanque.

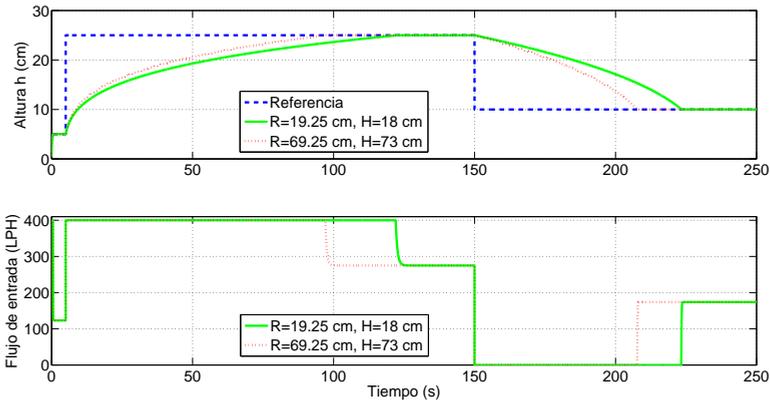


Figura 7.71: Respuesta al variar R a 69.25 cm y H a 18 cm .

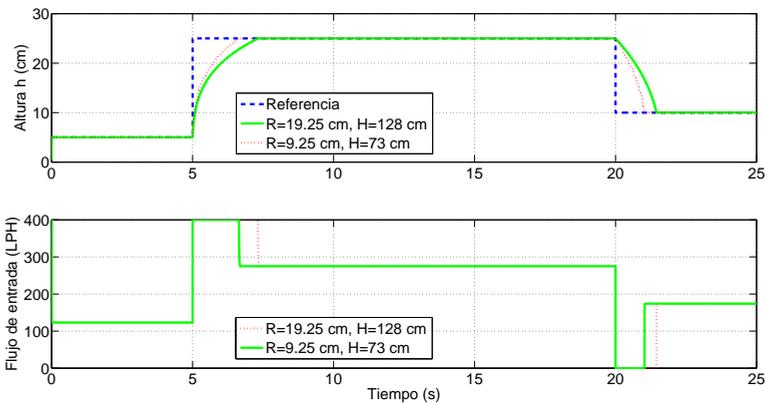


Figura 7.72: Respuesta al variar R a 9.25 cm y H a 128 cm .

7.4.6. Comportamiento frente a perturbaciones

Se analizará el efecto de las perturbaciones en el sistema diseñado en dos puntos operativos: $h = 5 \text{ cm}$ y $h = 25 \text{ cm}$. En el instante 10 s se aplica una carga en forma de escalón entorno al flujo de entrada F_{in} .

Para el nivel $h = 5 \text{ cm}$, figura 7.73, el sistema permanece estable si se varía la carga entre aproximadamente el -200% de F_{in} ($F_{load} = -246.1442 \text{ cm}^3/s$) y el $+100\%$ de F_{in} ($F_{load} = +123.0721 \text{ cm}^3/s$).

Para el nivel $h = 25 \text{ cm}$, figura 7.74, el sistema permanece estable si se varía la carga entre el -45% de F_{in} ($F_{load} = -123.7533 \text{ cm}^3/s$) y el $+100\%$ de F_{in} ($F_{load} = +275.0073 \text{ cm}^3/s$).

Como se puede observar en las figuras 7.73 y 7.74, el nivel h varía mínimamente cuando aparece la perturbación. Con esto, queda demostrado que el sistema de control propuesto minimiza los efectos de las perturbaciones de carga en un amplio rango.

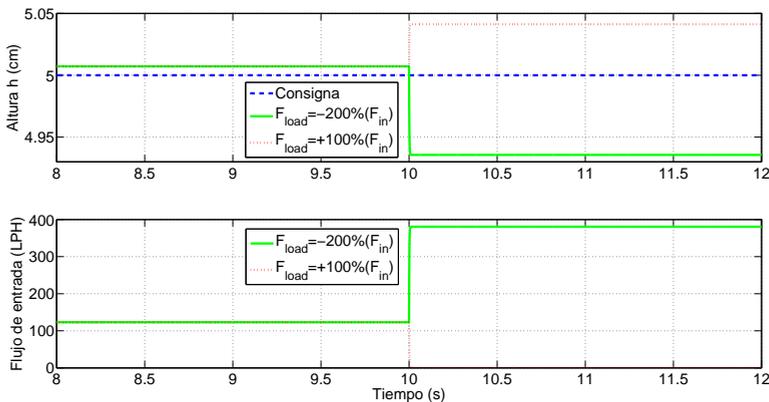


Figura 7.73: Respuesta con perturbación entorno a $h = 5 \text{ cm}$.

Sección 7.5

Aportaciones

En este capítulo se ha mostrado el trabajo realizado sobre sistemas que poseen no linealidades continuas y que dan lugar a distintos modelos locales del proceso. De la investigación llevada a cabo, a continuación se exponen las aportaciones realizadas.

Con respecto al sistema con no linealidades suaves:

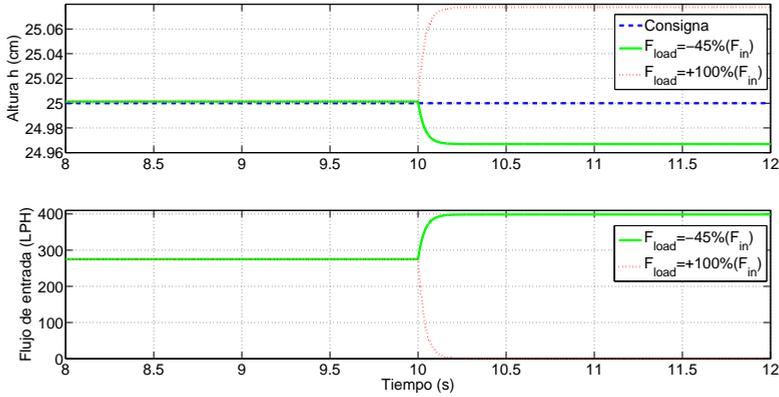


Figura 7.74: Respuesta con perturbación entorno a $h = 25$ cm.

1. Se han establecido las pautas a seguir para realizar una identificación adecuada en sistemas que posean no linealidades suaves. En este sentido, se muestra un patrón a seguir a la hora de obtener un buen modelo Takagi-Sugeno que identifique a la planta.
2. Para el bloque supervisor, se diseñó una lógica borrosa que genera, a partir de los antecedentes del modelo borroso de la planta, el peso de cada neurona y finalmente el valor de la señal de control.
3. Para el bloque supervisor, encargado de la integración de los subsistemas, se ha deducido, después de diversas pruebas, que los mejores resultados se obtienen con el método del mínimo para el conectivo *and* y el promedio ponderado para la desborrosificación.
4. Para mejorar la señal de control resultante, se observó que se pueden usar diferentes periodos de muestreo en la identificación y en el posterior sistema controlado.
5. Para anular los errores en régimen permanente, debidos a los errores de modelado, únicamente es necesario partir del PI borroso diseñado en capítulos anteriores y resintonizar las ganancias en función del proceso en cuestión.

Con respecto al sistema con no linealidades fuertes:

1. Se ha demostrado que la técnica de control inverso con modelo de referencia, no aplicada hasta ahora, es ideal para esta aplicación.

2. Establecer en cuatro el número de modelos locales lineales que modelan a un tanque cónico, independientemente de sus dimensiones y de su dinámica de descarga.
3. Fijar en un retardo puro el modelo de referencia para conseguir la mejor respuesta.
4. Obtención de un controlador sencillo formado por una red neuronal de cuatro neuronas con función de activación lineal y de una lógica borrosa para la integración de los subsistemas.
5. El control obtenido ha resultado ser estable y robusto para una variedad de cambios en los parámetros del sistema, no siendo necesario añadir el PI borroso.

La siguiente publicación es parte del trabajo presentado en este capítulo de la tesis:

C. S. Betancor, J. A. Montiel, and A. Vega, “*Direct Inverse Control for a Conical Tank by Using Takagi-Sugeno Fuzzy Model*”, IJAREEIE, Vol. 2, Issue 11, pp. 5327-5342, 2013.

Control de una no linealidad discontinua

El capítulo anterior se dedicó al control de sistemas que presentan no lineales continuas, tanto suaves como fuertes. Se realizó el diseño del sistema de control para una planta representada por una ecuación diferencial no lineal de segundo orden y para el control de nivel de líquido en tanques cónicos.

En este capítulo se trata el caso de una no linealidad discontinua que aparece comúnmente en muchos procesos industriales y que se denomina zona muerta (*dead zone*). Concretamente se estudiará y diseñará el sistema para la compensación de servosistemas de posicionamiento que presentan esta no linealidad. Para ello, se expondrán dos casos de diseño.

En el primer ejemplo la planta posee una zona muerta simétrica, siendo el objetivo del control el alcanzar la posición angular deseada en el mínimo tiempo posible y sin sobreimpulsos. Para conseguir esto, se diseñará el controlador inverso siguiendo pautas establecidas en capítulos anteriores y se compararán los resultados con un controlador predictivo generalizado. Para ver la influencia del proceso de identificación en la calidad del control resultante, se generarán diferentes modelos borrosos de la planta y se diseñará un controlador para cada modelo. Debido a que la respuesta obtenida ofrece un ligero error en régimen permanente, en lugar de añadir un PI borroso para corregirlo, se ha observado que una sintonización fina de los pesos de la red neuronal es suficiente para conseguir una completa exactitud. Por ello, se ajustan los parámetros de un algoritmo genético, para llevar a cabo dicha sintonización y optimizar las prestaciones del controlador inverso diseñado.

En el segundo ejemplo la planta posee una zona muerta asimétrica y el objetivo es seguir una trayectoria de referencia de la forma más exacta posible.

Para ello se diseñará el control inverso, siguiendo las pautas establecidas, y se realizará una comparativa con un controlador PD con compensación adaptativa robusta.

En los dos ejemplos se analizará la estabilidad y robustez del sistema diseñado, variando diferentes parámetros de la planta, entre ellos el ancho de la zona muerta, y también se observará la compensación que ofrece el controlador al producirse perturbaciones en el sistema.

En el siguiente capítulo, y último, se abordará el diseño del sistema de control aplicado a convertidores DC/DC conmutados. Estos convertidores pertenecen a una clase de sistemas no lineales variantes en el tiempo.

El capítulo se organiza como sigue. En la sección 8.1 se comienza introduciendo el concepto de zona muerta y se citan una serie de trabajos dedicados al control de esta no linealidad. A continuación, en la sección 8.2, se explica el modelado matemático de la zona muerta y el de un servosistema DC que posee esta no linealidad. El primer ejemplo de control se presenta en la sección 8.3, para un motor con zona muerta simétrica. En primer lugar, se describe al sistema y se presentan los resultados con un GPC propuesto en la literatura. Posteriormente, para realizar el diseño del controlador inverso, se realiza la identificación de la planta y mediante un modelo de referencia se diseña el controlador. Seguidamente se muestran los resultados obtenidos y se procede a entrenar la red neuronal, mediante un algoritmo genético, para eliminar los errores en régimen permanente. También se analiza la estabilidad y robustez del sistema. El segundo ejemplo se presenta en la sección 8.4, y consiste en un motor DC con zona muerta asimétrica. Se describe a la planta y se muestran los resultados, propuestos en la literatura, con un PD sin compensación y con un PD con compensación adaptativa robusta. Como en el primer ejemplo, se realiza la identificación de la planta y el diseño del controlador inverso. Se presentan los resultados obtenidos y se analiza la estabilidad y robustez del sistema. Finalmente, las aportaciones realizadas son presentadas en la sección 8.5.

Sección 8.1

Zona muerta

Las no linealidades discontinuas, no suaves o duras, también se las conoce como no linealidades rápidas, significando con ello que el modo de operar cambia rápidamente comparándolo con el tiempo de respuesta del sistema. Las no linealidades discontinuas que más habitualmente se presentan en los

sistemas son: fricción de Coulomb, saturación, todo-nada, holgura, histéresis y zona muerta. Estas no linealidades son comunes en los sensores y actuadores.

La zona muerta (*dead zone*) es una de las no linealidades discontinua más importantes en muchos procesos industriales, lo que puede limitar severamente el rendimiento del sistema. En el caso de un motor eléctrico, la zona muerta es el intervalo de tensión de entrada al motor que hace que no se produzca movimiento.

Dentro de los motores, el motor DC es uno de los actuadores que normalmente aparecen en los sistemas controlados. En el motor DC, al considerar un modelo ideal, se asume que cualquier voltaje de entrada aplicado causará movimiento de rotación, por lo que con pequeño voltaje se origina pequeño movimiento. En realidad, debido a la fricción estática del eje del motor, la rotación ocurrirá solo si el par proporcionado por el motor es suficientemente grande como para romper el rozamiento estático. La zona muerta en motores DC está relacionada de manera directa con el par necesario para iniciar el movimiento partiendo del reposo. Al voltaje necesario para sobrepasar la fuerza de fricción estática, se le conoce como voltaje de rompimiento. Normalmente en un motor esta zona muerta es simétrica, es decir, para que comience a girar en sentido horario la tensión positiva mínima a aplicar es la misma que la negativa para que gire en sentido antihorario. Por otra parte, el tiempo de respuesta de un motor es casi instantáneo, es decir, desde el instante que le aplicamos tensión hasta que el eje comienza a girar pasa un tiempo despreciable, por lo que se considera que el tiempo muerto es nulo y por lo tanto no aparece en el modelado el efecto del tiempo muerto.

Análogamente, cuando se transmite movimiento por conexión de componentes mecánicos, aparecen zonas muertas por los espacios debidos a la fabricación. En los actuadores neumáticos de válvulas controladas y en componentes hidráulicos ocurren fenómenos similares a la zona muerta. Las zonas muertas tienen un número de posibles efectos en los sistemas de control, empobreciendo las prestaciones del sistema. Su efecto más común es disminuir la precisión de la salida, con un excesivo error en régimen permanente y una pobre respuesta en el régimen transitorio. También pueden dar lugar a inestabilidad en el sistema debido a la ausencia de respuesta en la zona muerta. En algunos casos, sin embargo, pueden estabilizar un sistema o eliminar las autooscilaciones. Por ejemplo, si se añade una zona muerta a un relé se disminuye las conmutaciones de sus contactos, reduciendo el desgaste de los mismos. Otro ejemplo se puede encontrar en los sistemas hidráulicos, que a menudo tienen pequeñas zonas muertas, con el fin de evitar fugas de líquidos y reducir el desgaste.

A continuación se exponen diferentes técnicas que se han propuesto para la

compensación de este tipo de no linealidad. No se trata de hacer una descripción exhaustiva de todos los trabajos publicados sobre el tema, sino de dar una visión general sobre las estrategias más aplicadas para su compensación.

Un inverso de la zona muerta es una práctica que muchos autores han usado para diseñar sus controladores. Por ejemplo, en [215] se diseña un PI en serie con este inverso. Otro ejemplo se encuentra en [216], donde un *Proportional Derivative* (PD) se pone en serie con el inverso.

En [217] una red neuronal se encarga de estimar la zona muerta desconocida y otra proporciona compensación adaptativa. Una red neuronal adaptativa es usada en [218] para compensar la no linealidad en un sistema con motor DC. Usando redes neuronales recurrentes y con compensación adaptativa de la zona muerta, en [219] se controla un sistema SISO no lineal. Para reducir el efecto de no linealidades discontinuas que se presentan en los servomotores DC, en [220] se diseña un controlador basado en red neuronal. Otro ejemplo en el que se usa un control neuronal adaptativo se encuentra en [221].

La lógica borrosa también ha sido usada y se sigue aplicando para la compensación de este tipo de no linealidad. Así, en [222] se aplica a un motor DC, siendo el esquema de compensación adaptativo y estimándose de forma *online* el ancho de la zona muerta. Un controlador borroso adaptativo se diseña en [223] para aplicar a un sistema electro-hidráulico. Otro control borroso adaptativo para un sistema de posicionamiento industrial se aplica en [224]. También se combinan las redes neuronales y lógica borrosa, como en [225].

Un estudio del control adaptativo se puede encontrar en [226]. Un clásico lo encontramos en [227], donde se diseña un controlador no lineal adaptativo. El control adaptativo se aplica en [228] para controlar un motor DC con zona muerta incierta en la entrada. Otro ejemplo de aplicación del control adaptativo para un sistema con zona muerta desconocida de entrada se encuentra en [229].

Para la compensación en servosistemas DC, en [230] se aplica un esquema adaptativo robusto. La magnitud de la zona muerta es calculada *online* y con ello se consigue una alta precisión en el control. También, el control adaptativo robusto se aplica en [231], asegurando la estabilidad y la precisión de seguimiento.

Los controladores en modo deslizante también han sido aplicados, como en [232], para estabilizar de forma robusta a plantas no lineales con incertidumbre y ante la presencia de zonas muertas u holguras.

Por último, una estrategia de control intuitivo se implementa en [233]. Esta técnica es simple, pero se requiere una buena estimación del ancho de la zona muerta.

Modelado de la zona muerta

La relación entre la entrada y la salida, en algunos sistemas, puede presentar la forma de la figura 8.1. El término zona muerta se refiere a la zona plana en la que las señales de entrada no provocan efecto alguno en la salida del sistema.

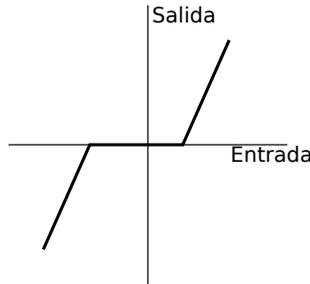


Figura 8.1: Zona muerta en un sistema.

En el motor DC se produce por la fricción de Coulomb, afectando tanto a la estabilidad como a las prestaciones del sistema. En los servosistemas de posición, el control de alta precisión está expuesto a comportamientos indeseados debido a esta no linealidad, siendo sus mayores efectos un excesivo error en régimen permanente y una respuesta pobre en el régimen transitorio.

Para obtener una representación matemática se puede partir de la figura 8.2. El par motor, T , está relacionado con la tensión de entrada aplicada al motor, u , mediante la expresión (8.1), donde m_+ y m_- son las pendientes en el primer y tercer cuadrante respectivamente. Esto se clarifica en la figura 8.3.

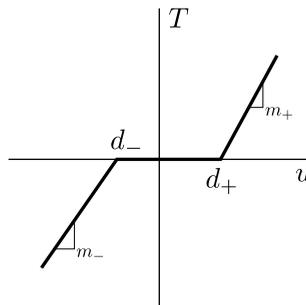


Figura 8.2: Función no lineal de la zona muerta.

$$T = \begin{cases} m_+ \cdot (u - d_+), & u \geq d_+ \\ 0, & d_- < u < d_+ \\ m_- \cdot (u - d_-), & u \leq d_- \end{cases} \quad (8.1)$$

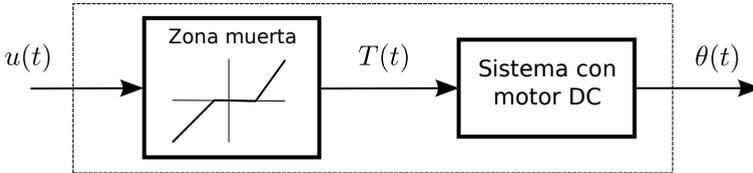


Figura 8.3: Servosistema de posicionamiento con zona muerta.

En la práctica, los valores de d_+ y d_- pueden ser desconocidos, por lo que resulta dificultosa su compensación. La mayoría de los esquemas de control cubren el caso en el que hay simetría, es decir, $|d_+| = |d_-|$. Aquí, se aplicará la metodología propuesta para el caso general en el que $|d_+| \neq |d_-|$.

En un servosistema de posicionamiento DC, la entrada al proceso es la tensión aplicada al motor $u(t)$ y la salida es la posición del eje del motor $\theta(t)$ (figura 8.3). Bien es conocido, que si se tiene en cuenta la inercia y la fricción de la carga, la ecuación diferencial de la que se obtiene el par desarrollado por el motor es la dada en (8.2), donde θ es la posición angular del motor, J es la inercia, B es el coeficiente de fricción viscosa, T_d representa la fricción no lineal más la perturbación externa y T es el par de control del actuador. Por simplicidad, no se incluye el efecto de vibración.

$$J \cdot \frac{d^2\theta}{dt^2} + B \cdot \frac{d\theta}{dt} + T_d = T \quad (8.2)$$

Un servosistema de posicionamiento presenta un efecto integral dado que la posición crece indefinidamente mientras el motor está alimentado con cierta tensión. La función de transferencia final se puede aproximar según (8.3), siendo K_m la ganancia y τ_m la constante de tiempo, dependiendo sus valores de las características electromecánicas del motor.

$$\frac{\theta(s)}{U_p(s)} = \frac{K_m}{s \cdot (\tau_m \cdot s + 1)} \quad (8.3)$$

Motor con zona muerta simétrica

Para realizar una comparativa de prestaciones, se emplearán los datos del motor que se refleja en el apartado 5.4.3. de la referencia [234], donde se diseña un GPC [235].

8.3.1. Descripción del sistema

Este motor tiene una ganancia $K_m = 2.5$ y una constante de tiempo $\tau_m = 0.9$ s, por lo que la función de transferencia es la dada en (8.4). La zona muerta tiene respuesta simétrica con $d_+ = 0.7$ V y $d_- = -0.7$ V. El motor es de 24 V de tensión nominal y 1.3 A de corriente nominal, por lo que se establece una salida del amplificador de potencia limitada en ± 24 V (figura 8.4). En (8.5) se describe la tensión $u_p(t)$ que se aplicaría al bloque definido en (8.4).

$$\frac{\theta(s)}{U_p(s)} = \frac{2.5}{s \cdot (0.9 \cdot s + 1)} \quad (8.4)$$

$$u_p(t) = \begin{cases} -23.3 \text{ V}, & u(t) < -24 \text{ V} \\ u(t) + 0.7 \text{ V}, & -0.7 \text{ V} \geq u(t) \geq -24 \text{ V} \\ 0 \text{ V}, & -0.7 \text{ V} < u(t) < 0.7 \text{ V} \\ u(t) - 0.7 \text{ V}, & 0.7 \text{ V} \leq u(t) \leq 24 \text{ V} \\ 23.3 \text{ V}, & u(t) > 24 \text{ V} \end{cases} \quad (8.5)$$



Figura 8.4: Modelo empleado para el motor.

8.3.2. Resultados con un GPC

En la figura 8.5 se puede observar la respuesta obtenida en [234] con un GPC para varios cambios en la referencia. Realizando medidas en la gráfica se obtiene un tiempo de establecimiento de aproximadamente 1.5 s en los cambios de consigna, siendo el sobreimpulso despreciable. Se produce un ligero error en régimen permanente y la señal de control no es muy buena al presentar cierto rizado.

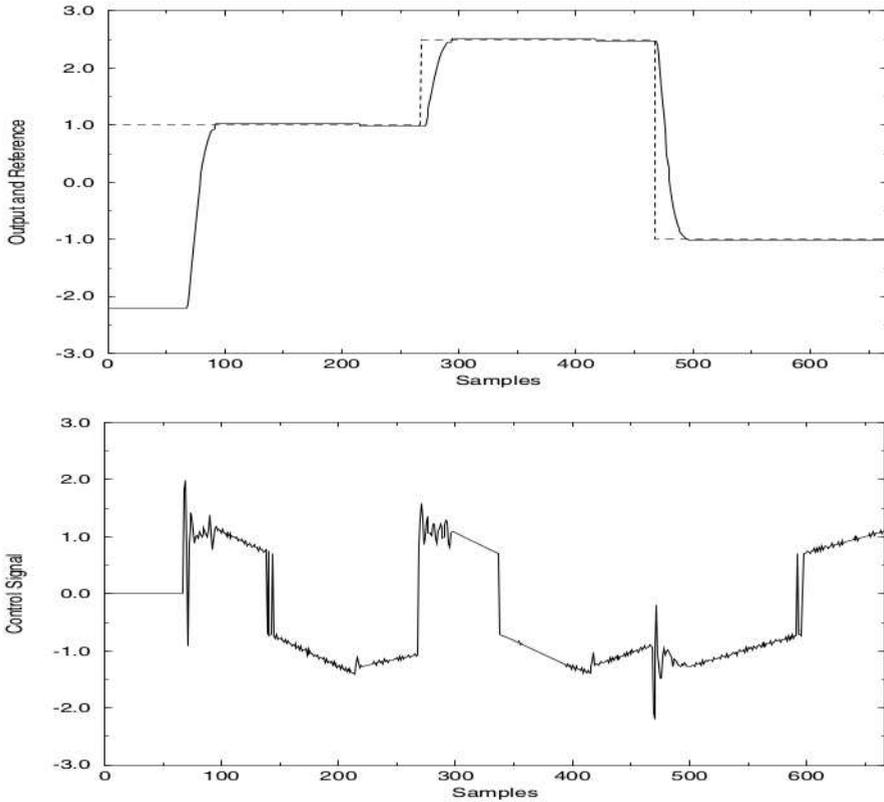


Figura 8.5: Respuesta del motor con un GPC (Fuente: [234]).

8.3.3. Identificación de la planta

En primer lugar se procede a la identificación y así obtener un modelo borroso Takagi-Sugeno. Para ello se aplican tensiones de entrada variables entre $\pm 3 V$ y se obtiene el ángulo que gira el eje del motor con un periodo de muestreo $T = 0.01 s$ (figura 8.6).

Con los parámetros en el algoritmo de identificación reflejados en la tabla 8.1, se consigue un $VAF = 90.4293 \%$, con el consecuente del modelo borroso dado en (8.6) —modelo borroso 1. Aumentando el orden del denominador a $.Ny = 3$ se consigue un $VAF = 96.9138 \%$ y el consecuente del modelo borroso definido en (8.7) —modelo borroso 2. Si se aumenta el parámetro $.Ny$ a 4 el VAF disminuye y al variar el coeficiente $.m$ no se consigue mejorarlo.

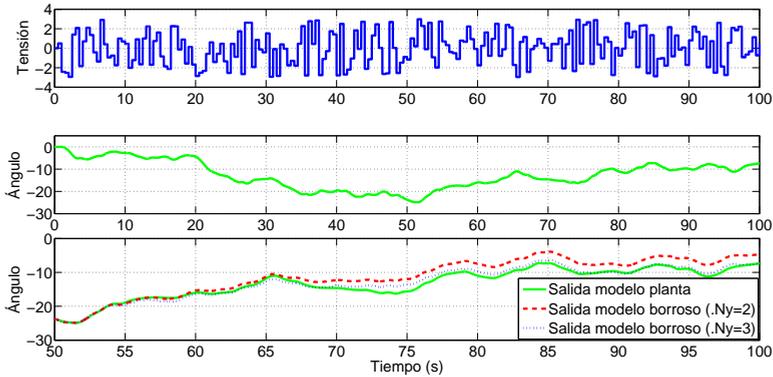


Figura 8.6: Señales para la identificación.

TABLA 8.1: PARÁMETROS PARA LA IDENTIFICACIÓN DEL MOTOR CON ZONA MUERTA SIMÉTRICA

Número de clústers	$.c = 1$
Coefficiente de borrosificación	$.m = 2$
Tolerancia de terminación	$.tol = 0.001$
Semilla para la inicialización	$.seed = 1$
Tipo de los antecedentes	$.ante = 2$
Estimación del consecuente	$.cons = 1$
Número de retardos de la salida	$.Ny = 2$
Número de retardos de la entrada	$.Nu = 1$
Número de retardos de transporte	$.Nd = 1$

$$y(k) = \left. \begin{aligned} &+ 1.99y(k-1) - 0.99y(k-2) + \\ &+ 1.79 \cdot 10^{-4}u(k-1) - 1.49 \cdot 10^{-5} \end{aligned} \right\} \quad (8.6)$$

$$y(k) = \left. \begin{aligned} &+ 2.66y(k-1) - 2.32y(k-2) + 0.661y(k-3) + \\ &+ 6.36 \cdot 10^{-5}u(k-1) - 6.43 \cdot 10^{-6} \end{aligned} \right\} \quad (8.7)$$

Ahora se varía la tensión de entrada en todo el rango dinámico disponible de $\pm 24 V$ y se procede a deducir otros modelos borrosos (figura 8.7). El objetivo de esto es diseñar diversos controladores para diferentes modelos borrosos y analizar posteriormente cual ofrece mejores prestaciones. Con los mismos parámetros presentados en la tabla 8.1 se consigue un $VAF = 99.4315\%$ con

el consecuente del modelo borroso (8.8) —modelo borroso 3. Aumentando el orden del denominador a $Ny = 3$ se consigue un $VAF = 99.9670\%$ y como consecuente del modelo borroso el dado en (8.9) —modelo borroso 4. En este caso tampoco se consigue mejorar el VAF al variar el coeficiente m .

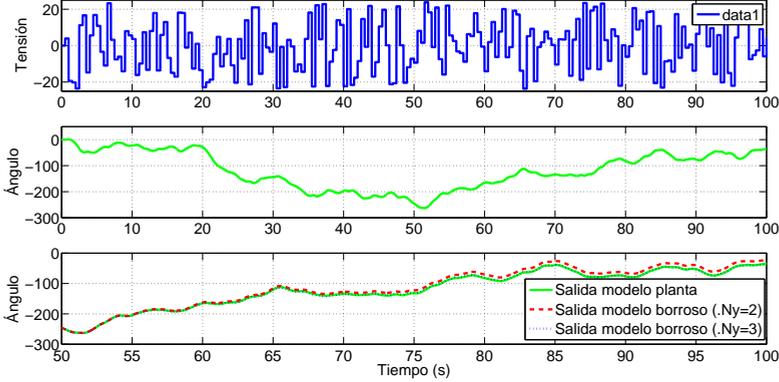


Figura 8.7: Señales para la identificación.

$$y(k) = \left. \begin{aligned} &+ 1.99y(k-1) - 0.989y(k-2) + \\ &+ 2.60 \cdot 10^{-4}u(k-1) - 1.10 \cdot 10^{-5} \end{aligned} \right\} \quad (8.8)$$

$$y(k) = \left. \begin{aligned} &+ 2.4y(k-1) - 1.80y(k-2) + 0.406y(k-3) + \\ &+ 1.57 \cdot 10^{-4}u(k-1) - 1.25 \cdot 10^{-5} \end{aligned} \right\} \quad (8.9)$$

8.3.4. Diseño del controlador inverso

Para comenzar con el diseño del controlador inverso hay que fijar el modelo de referencia. Para ello, y partiendo de los experimentos realizados en [234] (figura 8.5), se supone que la desviación angular máxima va a ser de $\pm 2.5 \text{ rad}$. Si se parte de la posición angular $\theta = -2.5 \text{ rad}$ y se aplican $+24 \text{ V}$ al motor, el tiempo que tarda en alcanzar la posición $\theta = +2.5 \text{ rad}$ es aproximadamente de 0.4 s . Por lo tanto, se establecerá a este como el tiempo de establecimiento del modelo de referencia, o sea, $t_e = 0.4 \text{ s}$. Siguiendo las pautas establecidas en el capítulo 2, se deduce la ecuación en diferencias (8.10) del modelo de referencia para $T = 0.01 \text{ s}$.

$$y_r(k) = 0.9068 \cdot y_r(k-1) + 0.0932 \cdot r(k-1) \quad (8.10)$$

Ahora se calculan cuatro controladores inversos para los cuatro consecuentes, (8.6), (8.7), (8.8) y (8.9), de los cuatro modelos identificados. Por ejemplo, para el consecuente (8.6) la señal de control tiene la expresión (8.11).

$$u(k) = + \left. \begin{aligned} & \frac{0.9068 - 1.99}{1.79 \cdot 10^{-4}} \cdot y(k) + \frac{0.99}{1.79 \cdot 10^{-4}} \cdot y(k - 1) + \\ & + \frac{0.0932}{1.79 \cdot 10^{-4}} \cdot r(k) + \frac{1.49 \cdot 10^{-5}}{1.79 \cdot 10^{-4}} \end{aligned} \right\} \quad (8.11)$$

8.3.5. Resultados obtenidos

Se realiza la simulación para un cambio en escalón de amplitud igual a 1 rad . En la figura 8.8 se observa que el transitorio difiere ligeramente del modelo de referencia, aunque el tiempo de establecimiento prácticamente es el impuesto de 0.4 s . Por otra parte, la respuesta es muy similar para los cuatro controladores.

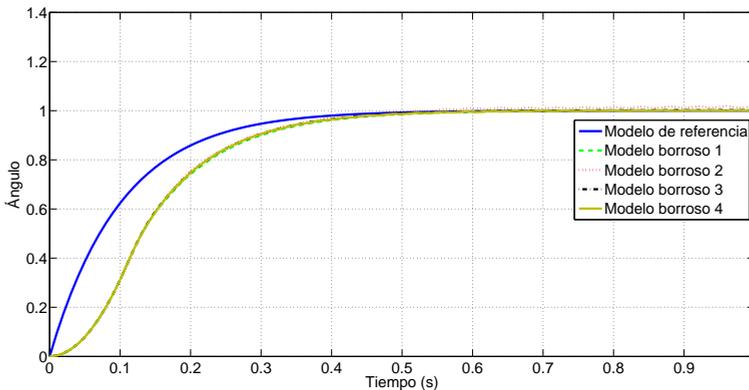


Figura 8.8: Respuestas a un cambio escalón unitario.

La figura 8.9 ilustra la respuesta en régimen permanente entre 1.7 s y 2 s . De esta figura se puede deducir que el mejor comportamiento se logra con los controladores 1 y 3, ya que presentan el menor error en régimen permanente sin fluctuaciones, siendo estos de -0.0015 rad y -0.0021 rad respectivamente.

La señal de control generada hacia la planta (figura 8.10) es constante una vez alcanzado el régimen permanente y en el transitorio presenta un comportamiento aceptable.

Ahora se verá el comportamiento del sistema ante errores de modelado. En principio se varía el ancho de la zona muerta entre $\pm 0 \text{ V}$ y $\pm 21 \text{ V}$. La figura 8.11 muestra que el sistema sigue siendo estable, aunque a medida que

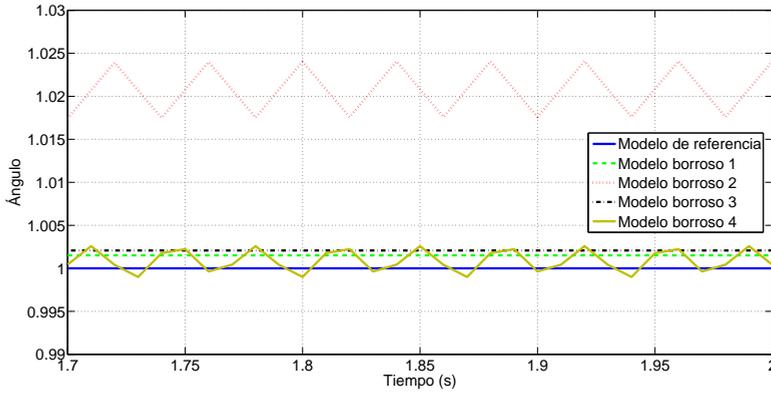


Figura 8.9: Respuestas en régimen permanente.

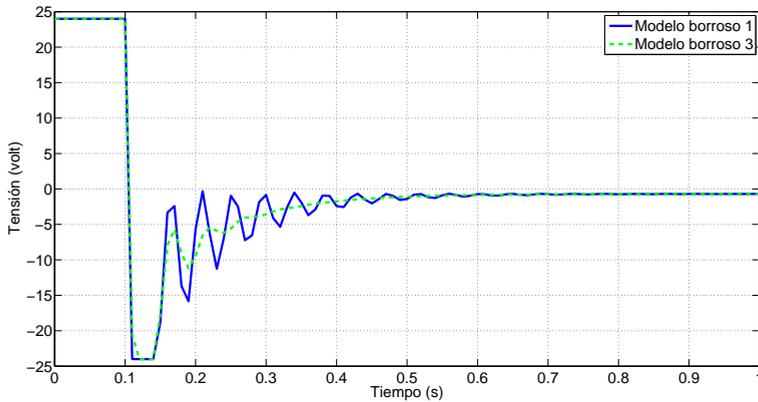


Figura 8.10: Señales de control para los controladores 1 y 3.

el ancho de la zona muerta es mayor, aumenta el error en régimen permanente y el tiempo de establecimiento. Sin embargo, la conclusión más importante es que el sistema es estable hasta que el ancho de la zona muerta es $d_+ < 24 V$ y $d_- > -24 V$.

Seguidamente, usando el controlador diseñado con el modelo borroso 1 (8.6), se aplica la misma referencia que en [234] (sección 8.3.2). Aunque en la figura 8.12 no se aprecie, se produce un pequeño error en régimen permanente. En la tabla 8.2 se muestra el valor final alcanzado para cada posición angular.

Si la aplicación no soporta estos ligeros errores, se puede plantear la incorporación del PI borroso. Sin embargo, si se quiere que el hardware final sea el

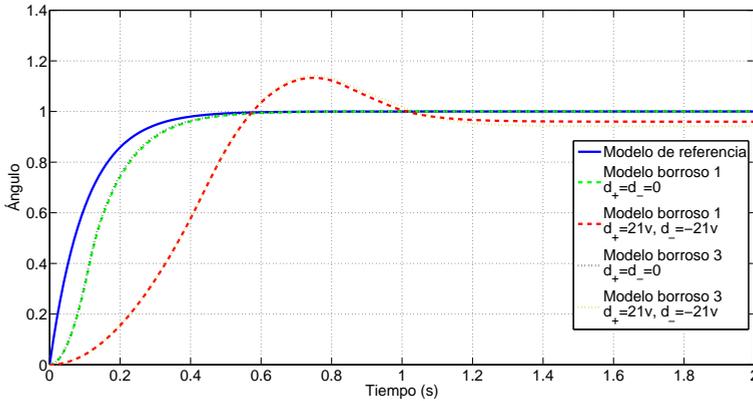


Figura 8.11: Respuesta al variar el ancho de la zona muerta.

TABLA 8.2: VALORES FINALES ALCANZADOS EN RADIANES

Consigna	-2.2000	1.0000	2.5000	-1.0000
Valor final	-2.2012	1.0015	2.5015	-1.0012

más simple posible, para esta aplicación se puede realizar un ajuste más fino de los pesos de la red neuronal y con ello anular los errores producidos por incertidumbres en el modelado, tal como se muestra en el siguiente apartado.

8.3.6. Entrenamiento de la red neuronal

En esta sección, se partirá del controlador obtenido con el modelo borroso 1, dado en (8.6), y se aplicará y sintonizará un GA para lograr que el transitorio se parezca más al del modelo de referencia y el régimen permanente presente un error igual a cero.

De la información del modelo borroso Takagi-Sugeno, obtenido con el paquete de herramientas FMID, una característica que se puede extraer es la desviación típica de los parámetros del consecuente. Esta desviación típica es la que ayudará a formar la matriz de la población inicial del GA. Para ello, se forman vectores de números aleatorios de media 0 y desviación típica la dada por el paquete de herramientas FMID para cada coeficiente del modelo borroso. El número de vectores es igual al número de coeficientes del modelo borroso. Por ejemplo, para el modelo (8.6), el número de vectores va a ser igual a 4. El número de componentes de cada vector va a ser igual al número

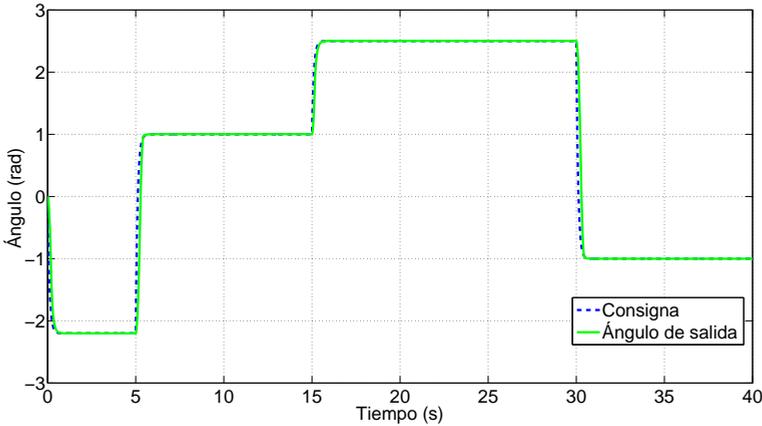


Figura 8.12: Respuesta para varios cambios de ángulo.

de individuos con los que se va a trabajar en el GA. Ahora se suman estos vectores a los coeficientes del modelo borroso, con lo cual se obtienen tantos modelos borrosos como individuos se quiera que tenga la población para el GA. Finalmente, se deducen las señales de control para cada modelo borroso según el modelo de referencia elegido. La matriz de la población inicial estará formada por los coeficientes de las señales de control deducidas, siendo estos coeficientes los pesos de la red neuronal que se va a entrenar.

Antes de empezar la sintonización de los parámetros del GA hay que elegir la función objetivo y la función aptitud. Dado que lo que se quiere es una mejora tanto en el transitorio como en el régimen permanente, se puede elegir la función objetivo IAE, dada en (2.30), como la función a minimizar. En cuanto a la función aptitud se elige la dada en (2.33).

Ahora se analiza la influencia del tamaño de la población. Primero se fija la tasa de mutación en el 10%, la tasa de reinserción en el 50%, se elige el muestreo universal estocástico (*sus*) como el método de selección y la recombinación intermedia (*recint*) como el método de cruce con una tasa del 50%. A continuación se varía el número de individuos en 10, 30, 50 y 100. En la figura 8.13 se observan los valores mínimos obtenidos para la función IAE. Para 9000 evaluaciones ya se ha conseguido el valor mínimo con los tamaños de 50 y 100, mientras que para 30 individuos hay que esperar hasta las 21000 evaluaciones. Con 10 individuos hay que realizar unas 28000 evaluaciones.

Así pues, los experimentos realizados a partir de ahora se centrarán en 50 y 100 individuos, usando 10000 evaluaciones como criterio de parada. Partiendo de los mismos parámetros en el GA que el experimento anterior, se analiza

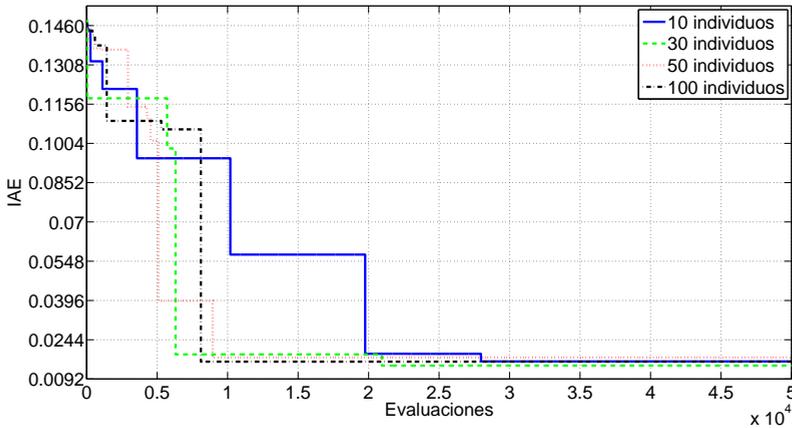


Figura 8.13: Mínimos de IAE según el tamaño de la población.

el efecto de modificar la tasa de mutación. En la tabla 8.3 se muestran los valores mínimos de la función objetivo para las diferentes tasas de mutación. Para 50 individuos el mínimo se alcanza con una tasa de mutación sobre el 30 %, aunque entorno al 50 % se obtiene un valor parecido. Con 100 individuos y 50 % de mutación se alcanza el valor más favorable. Según estos resultados, se seguirá manteniendo el 50 % de mutación para el resto de experimentos.

TABLA 8.3: VALORES MÍNIMOS DE IAE PARA DIFERENTES TASAS DE MUTACIÓN

	50 individuos	100 individuos
10 %	$9.3391 \cdot 10^{-3}$	$9.3201 \cdot 10^{-3}$
30 %	$9.1858 \cdot 10^{-3}$	$9.3224 \cdot 10^{-3}$
50 %	$9.2008 \cdot 10^{-3}$	$9.1804 \cdot 10^{-3}$
70 %	$9.1913 \cdot 10^{-3}$	$9.2662 \cdot 10^{-3}$
90 %	$9.2296 \cdot 10^{-3}$	$9.2982 \cdot 10^{-3}$

Ahora se estudia el efecto de modificar la tasa de reinserción. Partiendo de los parámetros en el GA del primer experimento, en la tabla 8.4 se reflejan los valores mínimos de la función objetivo. En este caso una tasa de reinserción alrededor del 30 % es la que mejores resultados ha dado tanto para 50 como para 100 individuos. A partir de ahora, el resto de simulaciones se realizará con un 30 % de reinserción.

Con una tasa de mutación del 50 % y una tasa de reinserción del 30 %, se

TABLA 8.4: VALORES MÍNIMOS DE IAE PARA DIFERENTES TASAS DE REINSERCIÓN

	50 individuos	100 individuos
10 %	$9.2148 \cdot 10^{-3}$	$9.2790 \cdot 10^{-3}$
30 %	$9.1704 \cdot 10^{-3}$	$9.1628 \cdot 10^{-3}$
50 %	$9.2558 \cdot 10^{-3}$	$9.1804 \cdot 10^{-3}$
80 %	$9.2969 \cdot 10^{-3}$	$9.2608 \cdot 10^{-3}$
100 %	$9.2137 \cdot 10^{-3}$	$9.1720 \cdot 10^{-3}$

estudia la influencia de las distintas metodologías de cruce¹. En la tabla 8.5 se observan los resultados obtenidos para una tasa de cruce igual al 50 %. Los mejores resultados se han obtenido para la recombinación lineal (*reclin*) y para el cruce de doble punto con sustituto reducido (*xovdprs*), tanto para 50 como 100 individuos.

TABLA 8.5: VALORES MÍNIMOS DE IAE PARA DIFERENTES METODOLOGÍAS DE CRUCE

Metodología de cruce	50 individuos	100 individuos
<i>recdis</i>	$9.2650 \cdot 10^{-3}$	$9.2702 \cdot 10^{-3}$
<i>recint</i>	$9.1704 \cdot 10^{-3}$	$9.1628 \cdot 10^{-3}$
<i>reclin</i>	$9.1647 \cdot 10^{-3}$	$9.1665 \cdot 10^{-3}$
<i>xovdp</i>	$9.1729 \cdot 10^{-3}$	$9.2640 \cdot 10^{-3}$
<i>xovdprs</i>	$9.1655 \cdot 10^{-3}$	$9.1630 \cdot 10^{-3}$
<i>xovsh</i>	$9.1887 \cdot 10^{-3}$	$9.1731 \cdot 10^{-3}$
<i>xovshrs</i>	$9.2099 \cdot 10^{-3}$	$9.1623 \cdot 10^{-3}$
<i>xovsp</i>	$9.1733 \cdot 10^{-3}$	$9.2637 \cdot 10^{-3}$
<i>xovsprs</i>	$9.2084 \cdot 10^{-3}$	$9.1925 \cdot 10^{-3}$

Ahora se varía la tasa de cruce para *reclin* con 50 individuos y *xovdprs* con 100 individuos. La tabla 8.6 muestra los resultados para 30 %, 50 % y 70 % de tasa de cruce. De la tabla se deduce que entorno al 50 % es la que mejores resultados ofrece.

En la figura 8.14 se observa la respuesta para un escalón de 1 *rad*, una vez sintonizados los pesos de la red neuronal. Se observa que el error en régimen permanente se ha anulado.

Ahora se analiza la estabilidad y robustez del sistema. En la figura 8.15 se

¹Una descripción detallada de los diferentes métodos y técnicas se encuentra en la referencia [165].

TABLA 8.6: VALORES MÍNIMOS DE IAE PARA DIFERENTES TASAS DE CRUCE

Tasa de cruce	50 individuos (<i>reclin</i>)	100 individuos (<i>xovdprs</i>)
30 %	$9.1826 \cdot 10^{-3}$	$9.1692 \cdot 10^{-3}$
50 %	$9.1647 \cdot 10^{-3}$	$9.1630 \cdot 10^{-3}$
70 %	$9.1708 \cdot 10^{-3}$	$9.2116 \cdot 10^{-3}$

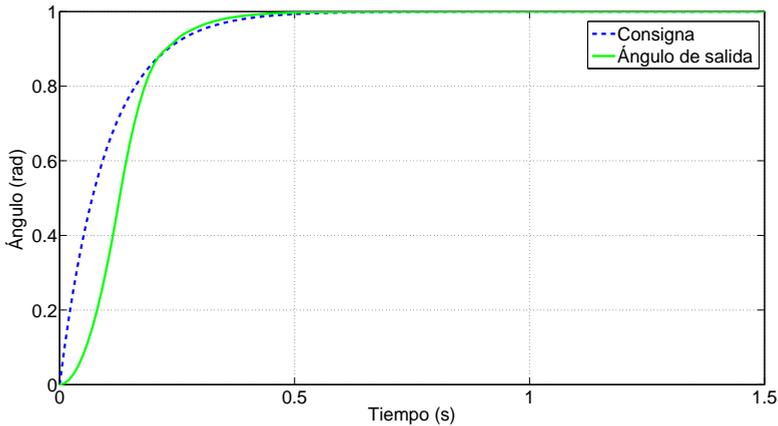


Figura 8.14: Respuesta con la red neuronal sintonizada.

muestra el comportamiento cuando se modifica el ancho de la zona muerta en un $\pm 50\%$ ($d_+ = 1.05 V$ y $d_- = -1.05 V$). También se simula el comportamiento cuando se modifica el ancho de la zona muerta y la función de transferencia del motor cambia a $[1.25/(0.45s^2 + s)]$ y a $[3/(0.7s^2 + s)]$. Como se ve, el sistema sigue respondiendo de forma adecuada con error en régimen permanente igual a cero.

Sección 8.4

Motor con zona muerta asimétrica

En este segundo ejemplo se hace uso del servosistema DC descrito en [230], el cual tiene una zona muerta asimétrica. Los autores presentan una comparativa entre un PD sin compensación de la zona muerta y el PD con compensación adaptativa robusta.

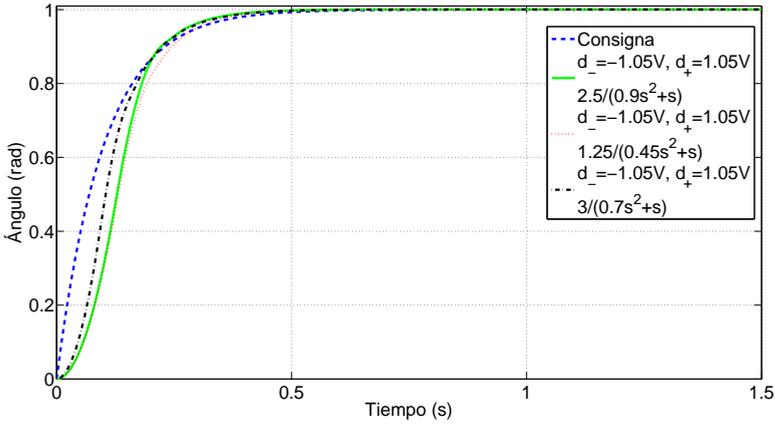


Figura 8.15: Respuesta ante errores de modelado.

8.4.1. Descripción del sistema

Para este servosistema DC se establece el valor del par motor T dado en la ecuación (8.12), donde $sat_d(u)$ es la función de saturación no simétrica y que está definida según la expresión (8.13).

$$T = u - sat_d(u) \tag{8.12}$$

$$sat_d(u) = \begin{cases} d_+, & u > d_+ \\ u, & d_- < u \leq d_+ \\ d_-, & d_- \geq u \end{cases} \tag{8.13}$$

Según (8.12) y (8.13), el par motor generado sigue la gráfica de la figura 8.2, siendo $m_+ = m_- = 1$. El amplificador lineal, entre el controlador y el motor, tiene una tensión de saturación de $\pm 0.8 V$ y el resto de parámetros del servosistema son: $J = 7.876 \cdot 10^{-3} V/rad \cdot s^{-2}$, $B = 8.666 \cdot 10^{-3} V/rad \cdot s^{-1}$, $d_+ = 0.08 V$ y $d_- = -0.2 V$. El diagrama de bloques de la figura 8.16 representa al conjunto formado por el amplificador y motor.

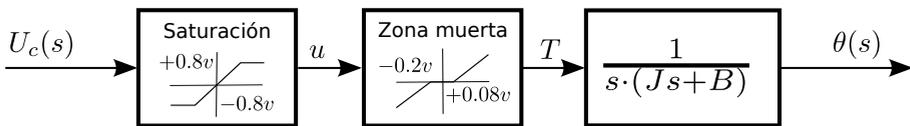


Figura 8.16: Modelo empleado para el motor.

8.4.2. Resultados con PD y controlador adaptativo robusto

En [230] la trayectoria $r(t)$ que se desea que siga el servosistema DC es la dada en (8.14).

$$r(t) = 2 + \sin t + 0.5 \sin 5t \quad (8.14)$$

En la figura 8.17 se puede observar el error que se produce con un PD sin ninguna compensación de la zona muerta. Realizando medidas en la gráfica, este error oscila aproximadamente entre $+0.12 \text{ rad}$ y -0.18 rad . Este error se reduce a $\pm 0.1 \cdot 10^{-3} \text{ rad}$, aproximadamente, cuando se aplica el controlador adaptativo robusto (figura 8.18), siendo la salida del sistema la mostrada en la figura 8.19. La señal de control necesaria tiene una amplitud máxima aproximada entre $+0.1 \text{ V}$ y -0.22 V (figura 8.20).

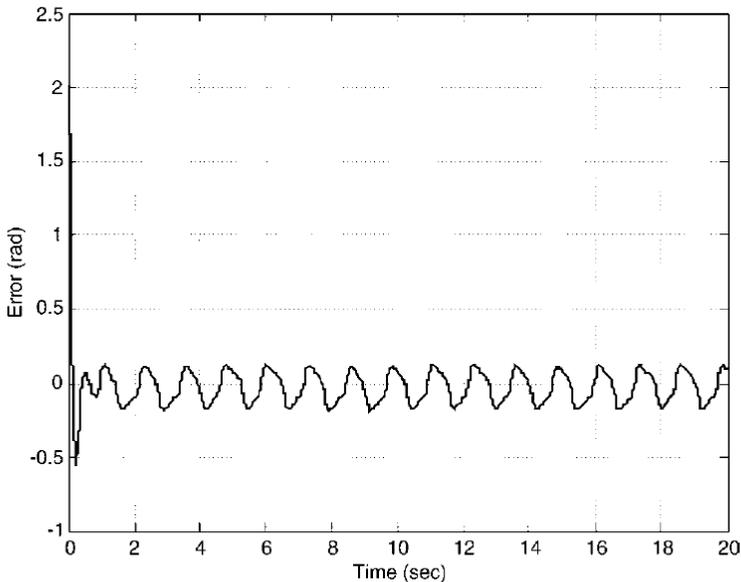


Figura 8.17: Error con un PD sin compensación de la zona muerta (Fuente: [230]).

Cuando se le añade al sistema una perturbación de -0.4 V , en el instante 2 s , se produce un error de media -0.05 rad (figura 8.21). El tiempo de muestreo que se utiliza en estas simulaciones es de 0.1 ms .

8.4.3. Identificación de la planta

El primer paso es identificar al servosistema DC para obtener un modelo borroso Takagi-Sugeno. Para ello se aplican tensiones de entrada variables u

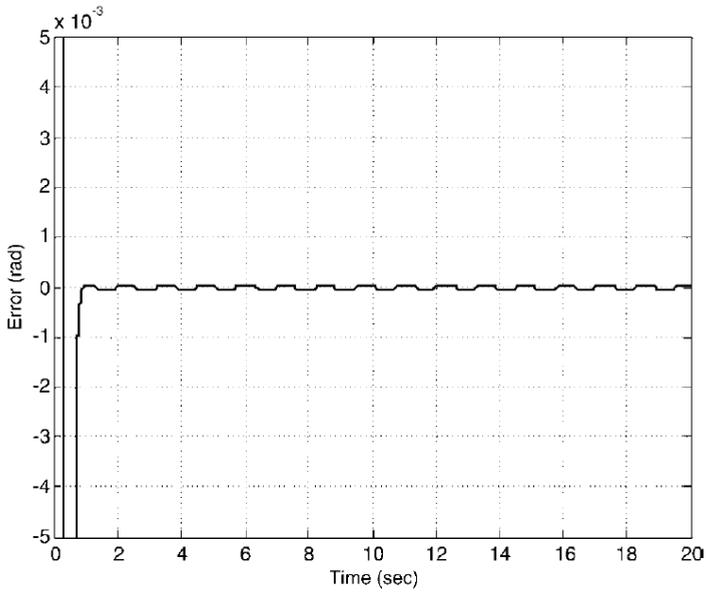


Figura 8.18: Error con un compensador adaptativo robusto (Fuente: [230]).

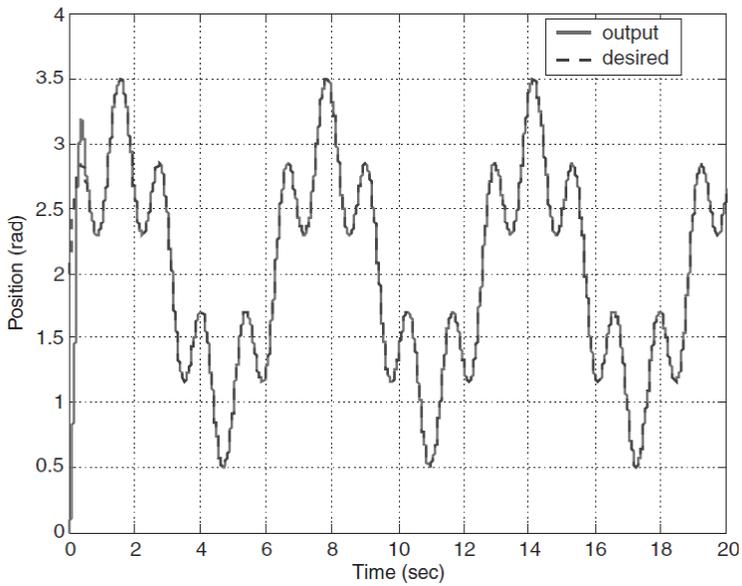


Figura 8.19: Salida con el controlador adaptativo robusto (Fuente: [230]).

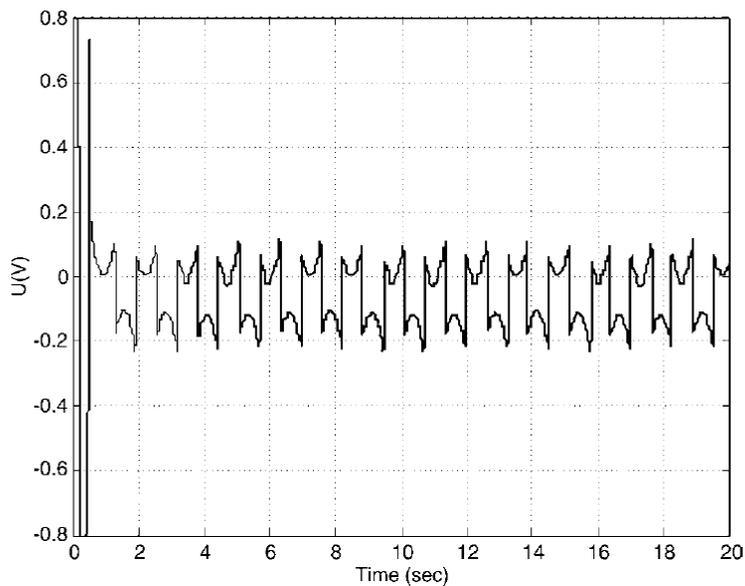


Figura 8.20: Señal de control con el controlador adaptativo robusto (Fuente: [230]).

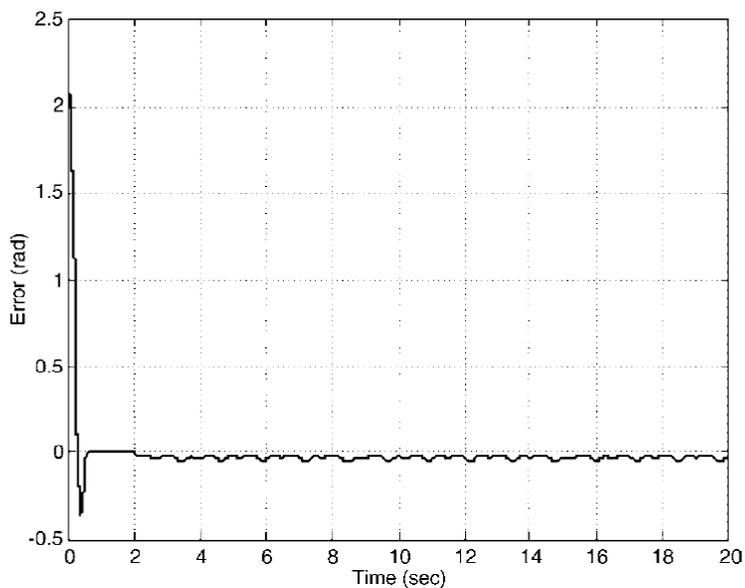


Figura 8.21: Error para una perturbación de $-0.4 V$ en el instante $2 s$ (Fuente: [230]).

entre $\pm 0.8 V$ y se obtiene el ángulo que gira el eje en cada instante (figura 8.22).

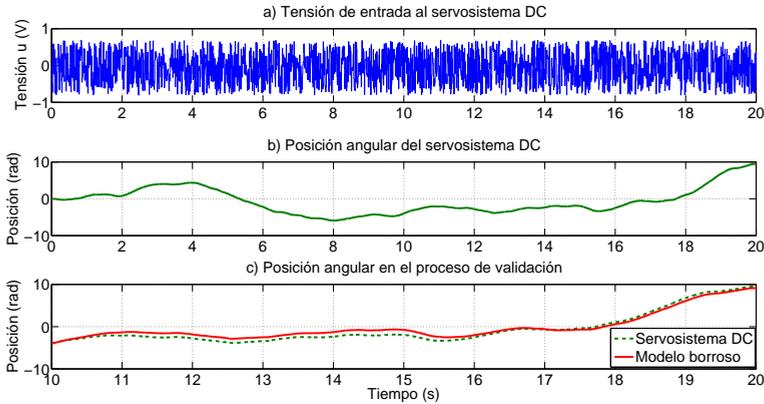


Figura 8.22: Señales para la identificación.

Con los parámetros en el algoritmo de identificación mostrados en la tabla 8.7, se consigue un $VAF = 96.9556\%$ y el consecuente del modelo dado en (8.15). Aumentando el orden del denominador a $.Ny = 3$ el VAF disminuye a $VAF = 90.9305\%$. En este caso al variar el coeficiente de borrosificación ($.m$) no se consigue mejorar el VAF .

TABLA 8.7: PARÁMETROS PARA LA IDENTIFICACIÓN DEL MOTOR CON ZONA MUERTA ASIMÉTRICA

Número de clústers	$.c = 1$
Coefficiente de borrosificación	$.m = 2$
Tolerancia de terminación	$.tol = 0.001$
Semilla para la inicialización	$.seed = 1$
Tipo de los antecedentes	$.ante = 2$
Estimación del consecuente	$.cons = 1$
Número de retardos de la salida	$.Ny = 2$
Número de retardos de la entrada	$.Nu = 1$
Número de retardos de transporte	$.Nd = 1$

$$\left. \begin{aligned}
 y(k) = & + 2y(k-1) - 0.999y(k-2) + \\
 & + 8.64 \cdot 10^{-5}u(k-1) + 5.10 \cdot 10^{-6}
 \end{aligned} \right\} \quad (8.15)$$

8.4.4. Diseño del controlador inverso

Para el diseño del controlador inverso, primero se fija el modelo de referencia con un tiempo de establecimiento de 0.3 s y se calcula el controlador siguiendo el proceso explicado en el capítulo 2. Mediante simulación se ha visto que si se disminuye el tiempo de establecimiento del modelo de referencia, el comportamiento de la respuesta transitoria del sistema es subamortiguada. Igualando la salida $y_r(k)$ del modelo de referencia (8.16) a la salida $y(k)$ del modelo borroso (8.15), se obtiene la señal de control $u(k)$ (8.17) que debe generar la red neuronal.

$$y_r(k) = 0.9870y_r(k-1) + 0.0130r(k-1) \quad (8.16)$$

$$u(k) = \left. \begin{aligned} &+ \frac{0.9870 - 2}{8.64 \cdot 10^{-5}} \cdot y(k-1) + \frac{0.999}{8.64 \cdot 10^{-5}} \cdot y(k-2) + \\ &+ \frac{0.0130}{8.64 \cdot 10^{-5}} \cdot r(k-1) - \frac{5.10 \cdot 10^{-6}}{8.64 \cdot 10^{-5}} \end{aligned} \right\} \quad (8.17)$$

8.4.5. Resultados obtenidos

Introduciendo la misma señal de referencia (8.14) que en [230] se obtiene el resultado de la figura 8.23. Una vez que ha pasado el transitorio, la salida del sistema sigue a la salida del modelo de referencia con un error de seguimiento máximo de $\pm 1.8 \cdot 10^{-4}\text{ rad}$ (figura 8.24). La señal de control necesaria es mostrada en la figura 8.25.

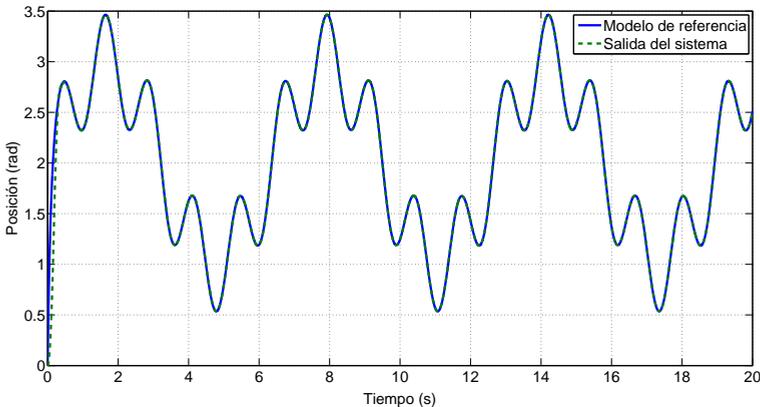


Figura 8.23: Salidas del modelo de referencia y del sistema.

En las figuras 8.23, 8.24 y 8.25, en el instante 10 s , y como en [230], se añade una perturbación de -0.4 V . Tal como queda reflejado en la figura 8.24, se

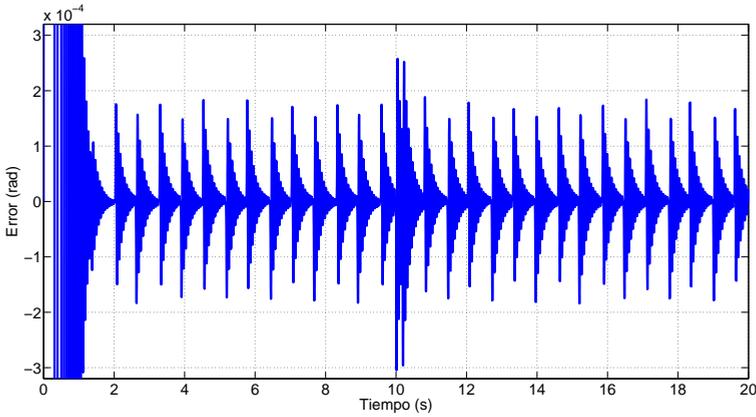


Figura 8.24: Error en el seguimiento de la trayectoria.

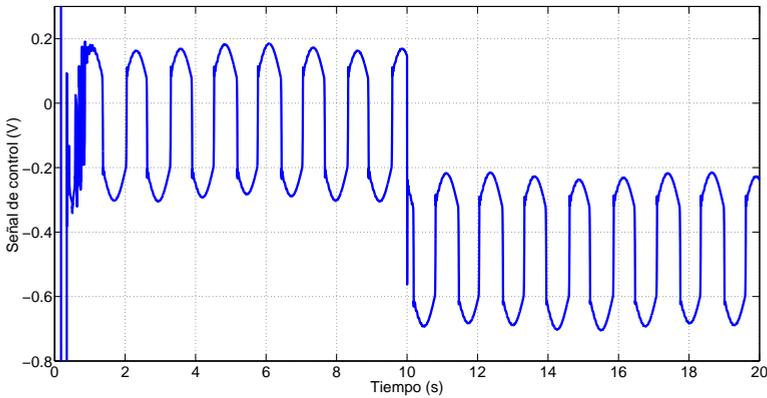


Figura 8.25: Señal de control para minimizar el error.

obtiene un error de seguimiento de media cero. El periodo de muestreo usado es de 1 ms , diez veces mayor que el usado en [230].

A continuación se realizan algunos experimentos que no se llevan a cabo en [230], pero con ellos se dará una mejor idea del comportamiento, estabilidad y robustez del sistema. En primer lugar se varían los límites de la zona muerta en tres rangos: a) $d_- = 0\text{ V}$ y $d_+ = 0\text{ V}$, b) $d_- = -0.4\text{ V}$ y $d_+ = 0.16\text{ V}$ y finalmente c) $d_- = -0.6\text{ V}$ y $d_+ = 0.24\text{ V}$. Aquí, también se aplica la perturbación de -0.4 V en el instante 10 s . La figura 8.26 muestra el error de seguimiento para estos tres casos. Cuando la zona muerta es nula, el error de seguimiento se reduce a un mínimo de $\pm 2 \cdot 10^{-6}\text{ rad}$. Aumentando en

un $\pm 100\%$ los valores de d_+ y d_- , el error aumenta a $\pm 2.5 \cdot 10^{-4} \text{ rad}$ sin perturbación y a $\pm 5 \cdot 10^{-4} \text{ rad}$ cuando se produce la perturbación, pero con media igual a cero. En este caso no hay un aumento significativo con respecto al de $\pm 1.8 \cdot 10^{-4} \text{ rad}$ que se producía sin error de modelado. Cuando se triplica el ancho de la zona muerta, el error aumenta a $\pm 0.5 \cdot 10^{-3} \text{ rad}$ sin perturbación y a $\pm 0.8 \cdot 10^{-3} \text{ rad}$ aplicando la perturbación, no habiendo tampoco un aumento drástico del error de seguimiento. Con esto se ha demostrado que el diseño es estable y robusto para incertidumbres en el modelado de la zona muerta.

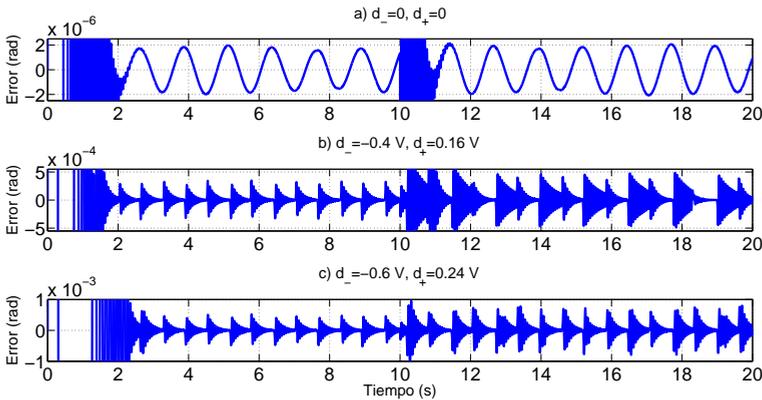


Figura 8.26: Error para incertidumbre de la zona muerta.

El próximo estudio se realiza para perturbaciones de mayor amplitud. En la figura 8.27 se muestra el comportamiento del sistema cuando una perturbación de -22 V se aplica en el instante 10 s . Como se puede observar, una vez pasado el transitorio, el sistema presenta el mismo error.

En el último experimento se modifican los valores de J y B entorno al 50% . En la figura 8.28 aparece el error de seguimiento para tres casos: a) $J = 0.0118 \text{ V/rad} \cdot \text{s}^{-2}$ y $B = 0.013 \text{ V/rad} \cdot \text{s}^{-1}$, b) $J = 0.0118 \text{ V/rad} \cdot \text{s}^{-2}$ y $B = 0.0043 \text{ V/rad} \cdot \text{s}^{-1}$ y finalmente c) $J = 0.0039 \text{ V/rad} \cdot \text{s}^{-2}$ y $B = 0.0043 \text{ V/rad} \cdot \text{s}^{-1}$. La perturbación de -22 V se aplica a los 10 s . Según la figura 8.28, el sistema es estable y robusto ante cambios de J y B .

Aportaciones

Este capítulo se ha centrado en la compensación de la no linealidad discontinua denominada zona muerta. Se pueden resumir las aportaciones realizadas

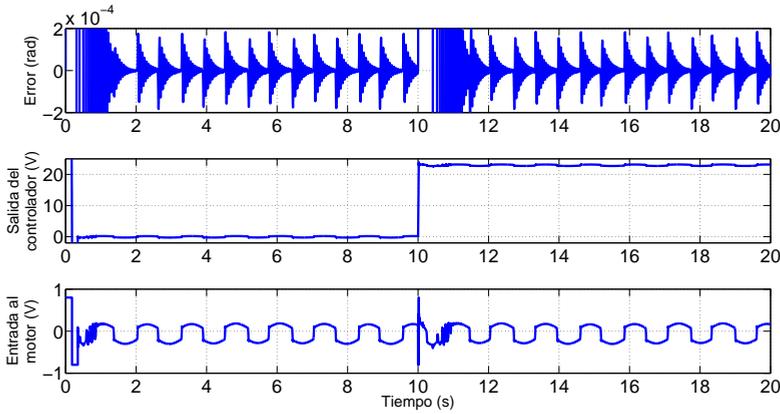


Figura 8.27: Respuesta para perturbación de -22 V a los 10 s .

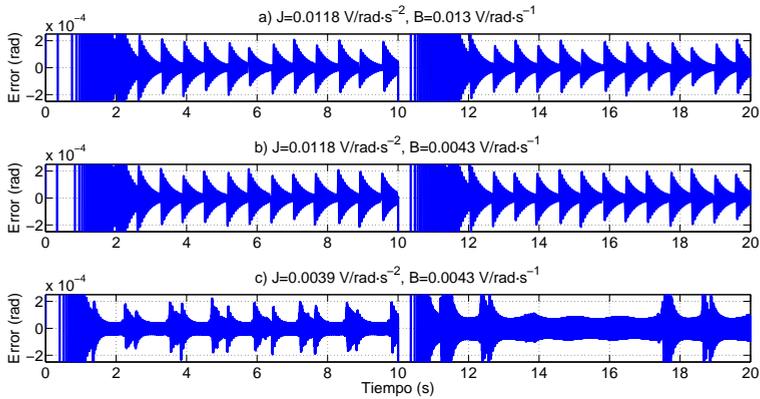


Figura 8.28: Errores para diferentes valores de J y B .

en los siguientes puntos:

1. El controlador final consiste de una única neurona, lo cual muestra la simplicidad para la posterior implementación hardware.
2. Se han definido los parámetros del algoritmo genético para sintonizar de una forma más óptima los pesos de la neurona, evitando el tener que agregar un PI borroso, simplificando al máximo el controlador.
3. Para el algoritmo genético se ha establecido la forma de fijar la población inicial a partir de la información disponible después del proceso de

identificación.

4. Al variar parámetros de la planta, incluyendo el ancho de la zona muerta, ha resultado un sistema controlado de una alta estabilidad y robustez.
5. El sistema resultante compensa perfectamente las perturbaciones.
6. El sistema diseñado muestra un mejor comportamiento frente a otras técnicas de control más complejas propuestas en la literatura.

La siguiente publicación es parte del trabajo presentado en este capítulo de la tesis:

Betancor-Martin, C.S. and Montiel-Nelson, J.A. and Vega-Martinez, A., “*Dead-zone Compensation in Motion Control Systems Using Model Reference Direct Inverse Control*”, Circuits and Systems (MWSCAS), 2014 IEEE 57th International Midwest Symposium on, pp. 165-168, 3-6 Aug. 2014.

Control de un convertidor conmutado DC/DC

En el capítulo anterior se aplicó la estrategia planteada a sistemas con no linealidades discontinuas, concretamente a sistemas que poseen la no linealidad no continua denominada zona muerta.

En este último capítulo se abordará el diseño para un tipo particular de plantas no lineales y además variantes en el tiempo, como es el caso de los convertidores conmutados DC/DC, más concretamente, se aplicará a convertidores conmutados reductores DC/DC.

Para el control de este tipo de convertidor se seguirá la misma filosofía planteada en capítulos anteriores, es decir, se realizará la identificación, para obtener un modelo Takagi-Sugeno, se establecerá un modelo de referencia y se diseñará el controlador inverso. Hay que tener en cuenta que como la aplicación en cuestión es un regulador, las referencias serán tipo escalón y lo importante será que la salida alcance el valor impuesto en la consigna de la forma más rápida posible y sin sobreimpulsos, de forma que la carga no tenga que soportar picos de tensión. Esto significa que la tensión de salida del regulador no debe porqué seguir exactamente a la salida del modelo de referencia. Debido a que en el convertidor se van a producir tanto variaciones en el valor de la carga, como variaciones en la tensión de entrada a regular, será necesario introducir el PI borroso que compense dichas variaciones que se producen en el circuito. Para su diseño se seguirá la misma estrategia que la planteada en el capítulo 5. Para observar y analizar las prestaciones, estabilidad y robustez del sistema, se realizará el diseño con un convertidor al que se le han aplicado diferentes estrategias del control. De los resultados obtenidos, se verá que con la propuesta presentada se obtiene un mejor comportamiento que con otras

técnicas más complejas propuestas en la literatura.

El capítulo se organiza como sigue. En la sección 9.1 se hace una introducción sobre los convertidores DC/DC. Se sigue, en la sección 9.2, exponiendo el modelo matemático del convertidor reductor. En la siguiente sección 9.3 se diseña el sistema de control, tanto del controlador inverso como del PI borroso, para un convertidor DC/DC reductor conmutado. Se demuestra la validez del control diseñado y se compara la respuesta con otras técnicas presentadas en la literatura. Además, se estudiará la estabilidad y robustez del sistema variando diferentes parámetros del convertidor. Para finalizar, en la sección 5.6, se expondrán las aportaciones realizadas.

Sección 9.1

El convertidor DC/DC

Los convertidores DC/DC —*Direct Current* (DC)— [236] son circuitos electrónicos de gran interés, siendo su control fundamental para obtener las máximas prestaciones. Hay tres tipos básicos de convertidores conmutados DC/DC: el convertidor reductor (*buck converter*), el convertidor elevador (*boost converter*) y el convertidor reductor-elevador (*buck-boost converter*). Este capítulo se centrará en el convertidor reductor.

El convertidor reductor DC/DC es un circuito cuyo objetivo es conseguir una tensión regulada de un valor inferior a la tensión de entrada, la cual normalmente no está regulada. Estas fluctuaciones en la tensión de entrada, o las variaciones en la carga de la salida, generan transitorios en la tensión de salida que el sistema de control debería minimizar. El convertidor reductor tiene muchas aplicaciones tales como fuentes de alimentación conmutadas, sistemas fotovoltaicos o cargadores de baterías. En general, este convertidor se aplica a cualquier sistema electrónico que requiera una tensión estabilizada.

Los convertidores conmutados DC/DC son sistemas no lineales y variantes en el tiempo, debido a que contienen elementos no lineales y además su estructura cambia en el tiempo por la acción de los interruptores de que dispone. Los diseñadores han intentado mejorar el comportamiento de estos convertidores modificando la estrategia de control o modificando su estructura.

En el trabajo que se presenta en este capítulo se usa la topología básica del convertidor conmutado reductor DC/DC y se diseña un control inverso directo que es complementado con un PI borroso para rechazar perturbaciones, tales como las variaciones de carga y las variaciones de la tensión de entrada, además de corregir los errores de modelado.

Publicaciones recientes como [237–245] dan idea de la gran cantidad de técnicas que se han desarrollado y se siguen investigando para el control de este tipo de convertidores.

Sección 9.2

Modelo del convertidor conmutado reductor DC/DC

En este apartado no se va a realizar un desarrollo exhaustivo del modelado del convertidor reductor DC/DC sino que se explicarán brevemente los conceptos básicos. Para una descripción detallada sobre esta temática se puede consultar, entre otras, las referencias [246–249].

La figura 9.1 ilustra el esquema de un convertidor conmutado reductor (*Switched Buck Converter*) DC/DC. La señal por modulación de ancho de pulso —*Pulse Width Modulation* (PWM)—, mostrada en la figura 9.2, con periodo T_s , tiene dos estados: estado en el que presenta una tensión V_{ON} , durante un tiempo t_{ON} , y estado en el que está a una tensión V_{OFF} , durante un tiempo t_{OFF} . El ciclo de trabajo d (*duty cycle*) es la relación entre el tiempo en el que la señal está en el estado *ON* y el periodo de la señal PWM: $d = t_{ON}/T_s$. El sistema de control será el encargado de generar la señal PWM.

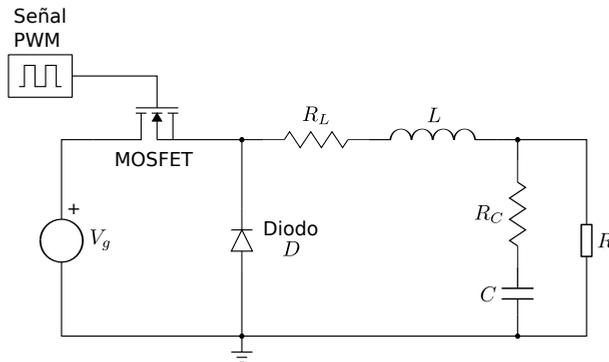


Figura 9.1: Convertidor conmutado reductor DC/DC.

El convertidor puede trabajar en dos modos diferentes: modo de conducción continuo —*Continuous Conduction Mode* (CCM)— y modo de conducción discontinuo —*Discontinuous Conduction Mode* (DCM)—. La diferencia entre ambos modos es que en CCM la corriente por la bobina no se hace cero mientras que en DCM la corriente por la bobina llega a ser cero en el subintervalo *OFF*.

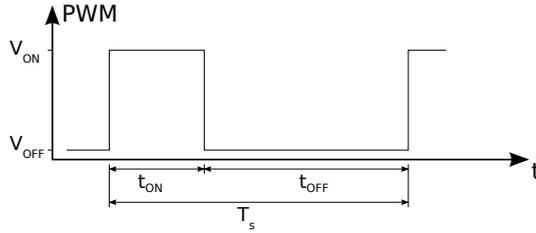


Figura 9.2: Señal PWM.

En CCM y cuando la señal PWM está en V_{ON} el MOSFET —*Metal Oxide Semiconductor Field Effect Transistor*— está en estado de conducción, presentando el modelo de la figura 9.3. Para conseguir un modelo más aproximado, se incluye la resistencia entre drenador y fuente del MOSFET (R_{on}), la resistencia serie equivalente de la bobina (R_L) y la resistencia equivalente serie del condensador (R_C). La figura 9.4 muestra el modelo del convertidor cuando la señal PWM está en V_{OFF} , donde se incluye la resistencia del diodo R_d .

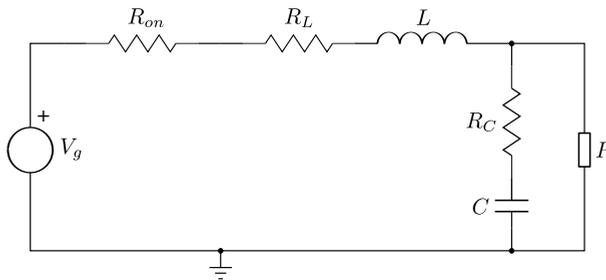


Figura 9.3: Modelo para CCM y estado ON.

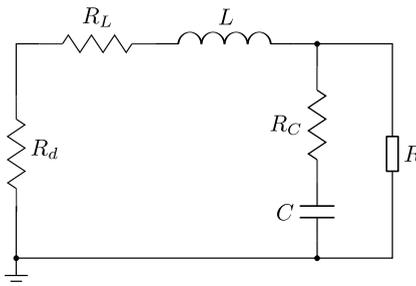


Figura 9.4: Modelo para CCM y estado OFF.

En DCM, aparte de los modelos de las figuras 9.3 y 9.4, se produce otro circuito equivalente cuando la corriente por la bobina es cero (figura 9.5).

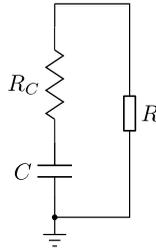


Figura 9.5: Modelo para DCM cuando la corriente es cero.

A partir de los modelos de las figuras 9.3, 9.4 y 9.5, se deducen las ecuaciones que modelan al convertidor. Normalmente lo que se deduce es un modelo promediado, ya que es fácil de analizar aunque se pierda cierta información. Para el caso del modo CCM, las ecuaciones que definen el modelo promedio son dadas en (9.1). El caso del modo DCM no se tratará aquí.

$$\left. \begin{aligned} \frac{di_L}{dt} &= \frac{1}{L} \cdot (V_g \cdot d - i_L \cdot R_L - v_R) \\ \frac{dv_C}{dt} &= \frac{1}{C} \cdot (i_L - i_R) \\ v_o &= v_C + R_C \cdot (i_L - i_R) \end{aligned} \right\} \quad (9.1)$$

En este trabajo no se usarán las ecuaciones del convertidor, sino que se introducirá el circuito eléctrico con el paquete de herramientas *SimPowerSystems* en *Simulink*[®] y mediante un proceso de identificación se sacará un modelo Takagi-Sugeno que define el comportamiento del convertidor.

Sección 9.3

Convertidor DC/DC reductor

En este ejemplo se trabajará con el convertidor descrito en [250], donde los autores presentan una comparativa entre cinco estrategias de control: PID Continuo (PIDC), PID Discreto (PIDD), control óptimo empleando el índice de prestaciones cuadrático —*Quadratic Performance Index* (QPI)—, control borroso —FC— y redes neuronales artificiales —*Artificial Neural Network* (ANN).

Los valores de los parámetros del convertidor son: $f_s = 25 \text{ KHz}$, $V_g = 12 \text{ V}$, $L = 30 \text{ mH}$, $R_L = 0.3 \Omega$, $C = 4700 \mu\text{F}$, $R_C = 0 \Omega$ y $R = 2 \Omega$.

9.3.1. Diseño del sistema de control

Para el diseño del controlador inverso, el primer paso es identificar al convertidor para obtener un modelo Takagi-Sugeno. Para ello se varía el ciclo de trabajo d , de la señal PWM, entre 0 y 1 y se obtiene la tensión de salida (figura 9.6).

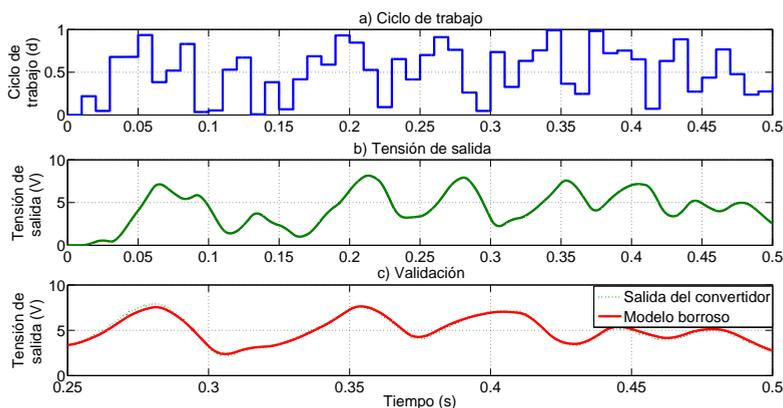


Figura 9.6: Señales para la identificación.

Para los parámetros del algoritmo de agrupamiento dados en la tabla 9.1 se consigue un $VAF = 98.7445\%$. Si se aumenta el parámetro $.Ny$ a 3 el VAF disminuye ligeramente ($VAF = 98.7218\%$).

TABLA 9.1: PARÁMETROS PARA LA IDENTIFICACIÓN DEL CONVERTIDOR

Número de clústers	$.c = 1$
Coefficiente de borrosificación	$.m = 2$
Tolerancia de terminación	$.tol = 0.001$
Semilla para la inicialización	$.seed = 1$
Tipo de los antecedentes	$.ante = 2$
Estimación del consecuente	$.cons = 1$
Número de retardos de la salida	$.Ny = 2$
Número de retardos de la entrada	$.Nu = 1$
Número de retardos de transporte	$.Nd = 1$

El consecuente del modelo borroso Takagi-Sugeno obtenido para $.Ny = 2$

está dado en (9.2).

$$y(k) = \left. \begin{aligned} &+ 2y(k-1) - y(k-2) + \\ &+ 1.41 \cdot 10^{-6}u(k-1) - 1.21 \cdot 10^{-7} \end{aligned} \right\} \quad (9.2)$$

Para el diseño del controlador inverso, y tal como se ha explicado en los capítulos anteriores, hay que fijar el modelo de referencia. Para esta aplicación, no interesa tanto que la salida siga a la trayectoria del modelo de referencia, sino que el valor final de la tensión de salida se alcance lo antes posible y sin sobreimpulsos. Por ello, mediante simulación, se ha establecido el valor mínimo del tiempo de establecimiento del modelo de referencia como aquel para el cual la salida no presenta sobreimpulsos ante cambios de la consigna, es decir, aquel para el cual el transitorio presenta respuesta monótona. Este tiempo de establecimiento ha resultado ser de 8 ms y, por lo tanto, la ecuación en diferencias del modelo de referencia es la dada en (9.3).

$$y_r(k) = 0.99805y_r(k-1) + 0.00195r(k-1) \quad (9.3)$$

Dado que para la identificación ha sido necesario únicamente un clúster, el modelo obtenido es lineal. El consecuente del modelo está dado en 9.4.

$$u(k) = \left. \begin{aligned} &+ \frac{0.99805 - 2}{1.41 \cdot 10^{-6}} \cdot y(k) + \frac{1}{1.41 \cdot 10^{-6}} \cdot y(k-1) + \\ &+ \frac{0.00195}{1.41 \cdot 10^{-6}} \cdot r(k) + \frac{1.21 \cdot 10^{-7}}{1.41 \cdot 10^{-6}} \end{aligned} \right\} \quad (9.4)$$

Para esta aplicación no es conveniente que se produzcan sobreimpulsos cuando, por ejemplo, hay una variación del valor de la carga o cuando se producen variaciones en la tensión de entrada a regular. Por lo tanto, es necesario insertar otro controlador que complemente al controlador inverso para que fuerce a que en las posibles variaciones de la tensión de salida no se produzcan oscilaciones. Por este motivo se diseñará un PI borroso siguiendo las pautas establecidas en el capítulo 5.

Por lo tanto, se partirá de la estructura del PI borroso presentado en el apartado 5.4.2 y mediante un GA se sintonizarán las ganancias G_e , G_{de} y G_u . En el apartado 5.4.4 se analizó de forma detallada la influencia del tamaño de la población, de los diferentes métodos de cruce y de las distintas tasas de cruce, mutación y reinsertión en el resultado final. El modelo de rechazo de perturbación se debe crear específicamente para cada planta. En este caso, se calcula para una perturbación máxima de cambio del valor de la carga de 2 Ω. Por simulación se obtiene que el valor de pico del modelo de rechazo de

perturbación será de aproximadamente 5.5 mV con un tiempo de establecimiento de unos $4 \mu\text{s}$. Con estos datos y según lo explicado en el apartado 5.3 se calcula el modelo de rechazo de perturbación que será usado para sintonizar las ganancias del PI borroso. Finalmente, como función objetivo se usa la ISEOISRAE explicada en el apartado 5.2.

9.3.2. Simulaciones y resultados

En la figura 9.7 se puede observar la respuesta para un cambio en la consigna de 0 V a 6 V , en el instante $t = 1 \text{ s}$, junto con el error que se produce. Como se ve, la respuesta es muy rápida y sin sobreimpulsos.

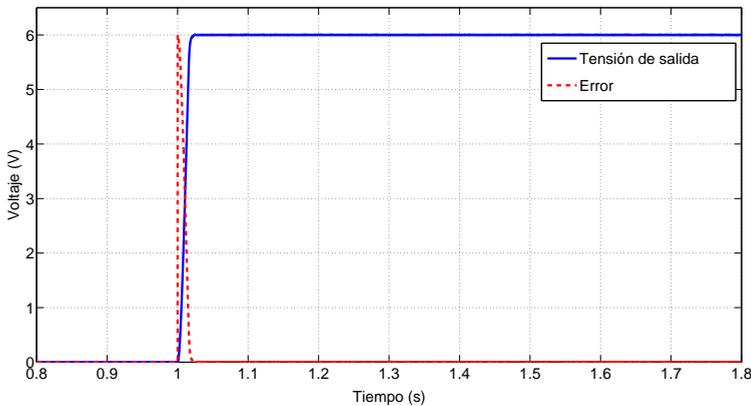


Figura 9.7: Respuesta para un cambio de 0 V a 6 V .

La tabla 9.2 muestra el tiempo de establecimiento (t_e) y el sobreimpulso (M_p) obtenido en [250] frente a la propuesta presentada aquí y medida en la figura 9.7. El tiempo de establecimiento se mide entorno al $\pm 2\%$ del valor final. Como se refleja en la tabla 9.2, la propuesta presentada es más rápida, $t_e = 0.0181 \text{ s}$, y presenta un sobreimpulso igual a cero. Así, se obtiene el mismo sobreimpulso que con el PIDC, mientras que en términos del tiempo de establecimiento la propuesta presentada decremanta el tiempo requerido alrededor de 1 ms con respecto al control QPI y 506 ms respecto al PIDC cuando se requiere 0% de sobreimpulso.

La figura 9.8 muestra la evolución de los índices de prestaciones ISE, IAE e ITSE. En la tabla 9.3 quedan resumidos sus valores y tal como se observa, con la propuesta planteada se consiguen valores inferiores.

A continuación se analiza el comportamiento del sistema controlado cuando la señal de referencia cambia a los valores 6 V , 2 V , 4 V , 8 V y 3 V , en los

TABLA 9.2: TIEMPO DE ESTABLECIMIENTO Y SOBREPULSO

	PIDC	PIDD	QPI	FC	ANN	Propuesta presentada
t_e (s)	0.5240	0.1620	0.0190	0.1430	0.2357	0.0181
M_p (%)	0	21.6	3.96	2.59	5.73	0

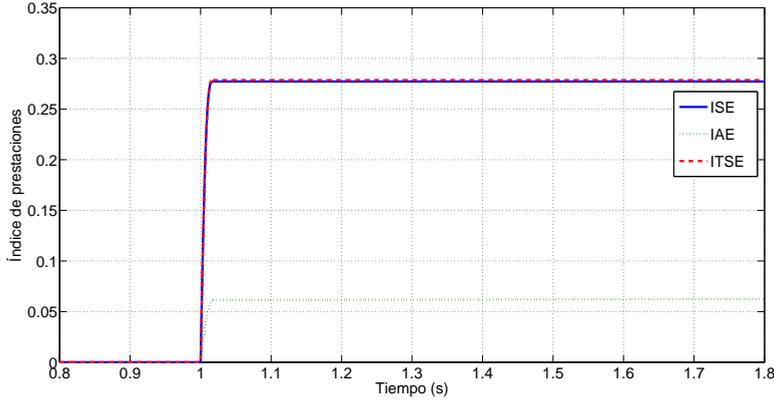


Figura 9.8: Evolución de los índices de prestaciones.

TABLA 9.3: COMPARACIONES DE LOS ÍNDICES DE PRESTACIONES

	PIDC	PIDD	QPI	FC	ANN	Propuesta presentada
ISE	1.2319	1.1462	0.2981	0.5232	0.6499	0.2772
IAE	0.5642	0.2883	0.0856	0.1745	0.2495	0.0625
ITSE	1.2974	1.1647	0.2986	0.5276	0.6620	0.2786

instantes 0.1 s, 0.5 s, 1 s, 1.5 s y 2 s respectivamente, para cargas de 2 Ω y de 50 Ω . En la figura 9.9 se observa que el sistema sigue a la referencia perfectamente y en un tiempo mínimo. Para una carga R de 50 Ω , por el proceso de descarga, tarda más en pasar de un nivel de tensión mayor a otro menor y con un ligero sobreimpulso. Sin embargo, si se compara con los resultados obtenidos en [250] (figuras 9.10, 9.11, 9.12, 9.13, 9.14, 9.15), la propuesta presentada aquí responde mejor, demostrando la estabilidad y robustez del sistema cuando se varía la carga. El QPI no tiene un buen comportamiento robusto, mientras que FC y ANN muestran un comportamiento más robusto.

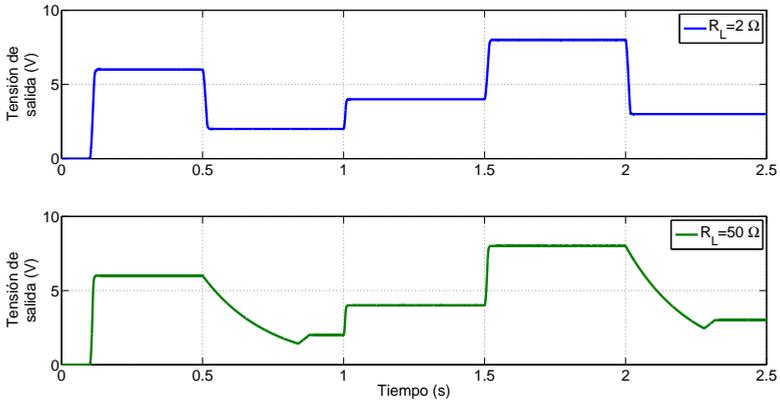


Figura 9.9: Respuesta para variaciones en la consigna y carga.

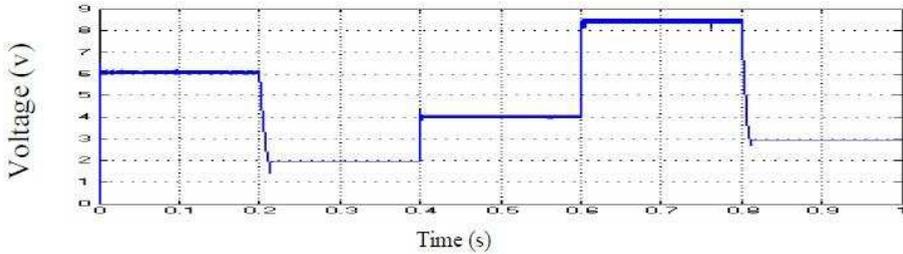


Figura 9.10: Respuesta con QPI y carga de 2Ω (Fuente: [250]).

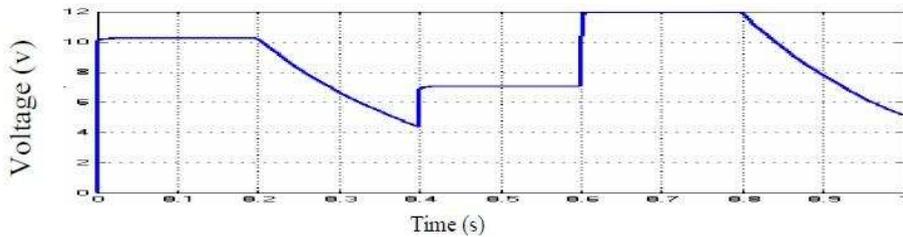


Figura 9.11: Respuesta con QPI y carga de 50Ω (Fuente: [250]).

Además, se realizarán algunos experimentos extras para dar una mejor visión del comportamiento, la estabilidad y la robustez del sistema. La figura 9.16 ilustra la salida del sistema cuando la carga se variaba a 1Ω y 0.7Ω . Aparece un rizado para las tensiones de salida más elevadas, sin embargo, para las tensiones más bajas el rizado desaparece. Para cargas inferiores a 0.7Ω el sistema se hace inestable.

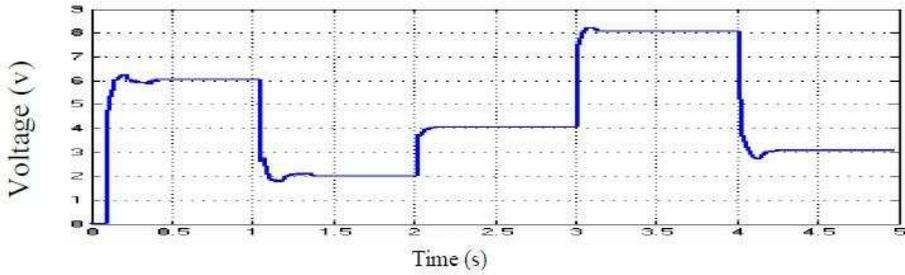


Figura 9.12: Respuesta con FC y carga de 2Ω (Fuente: [250]).

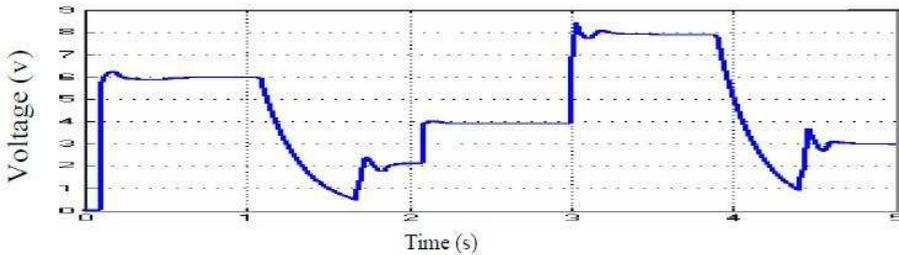


Figura 9.13: Respuesta con FC y carga de 50Ω (Fuente: [250]).

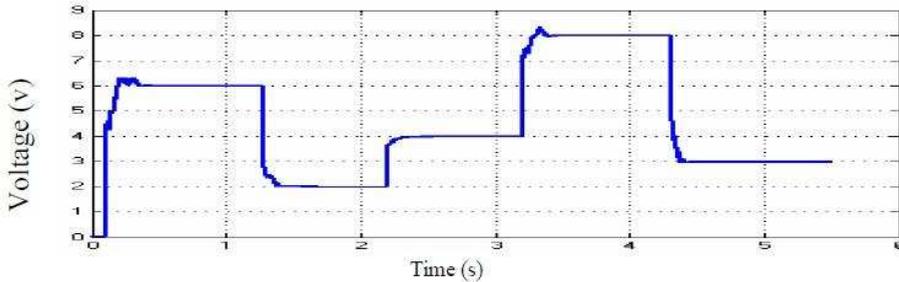


Figura 9.14: Respuesta con ANN y carga de 2Ω (Fuente: [250]).

La próxima prueba es variar la tensión de entrada V_g y la consigna. Se empieza con $V_g = 12 V$ y se cambia a $6 V$, $18 V$, $15 V$ y $7 V$ en los instantes $0.5 s$, $0.7 s$, $1.1 s$ y $1.3 s$, respectivamente. La referencia cambia de $0 V$ a $7 V$ a los $0.1 s$, de $7 V$ a $2 V$ a los $0.3 s$, de $2 V$ a $8 V$ a los $0.9 s$ y de $8 V$ a $3 V$ a los $1.1 s$. Como se muestra en la figura 9.17, la tensión de salida sigue a la consigna. Con esto se demuestra que el sistema es también estable y robusto frente a cambios de V_g .

Por último, se verá el rechazo a las perturbaciones de carga. En la figura 9.18 se observa la respuesta del convertidor cuando se producen cambios en

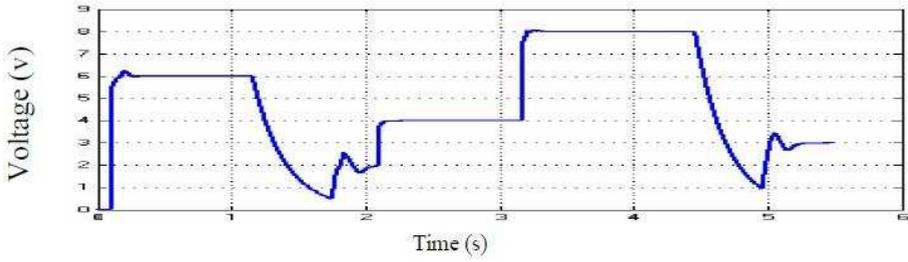


Figura 9.15: Respuesta con ANN y carga de 50Ω (Fuente: [250]).

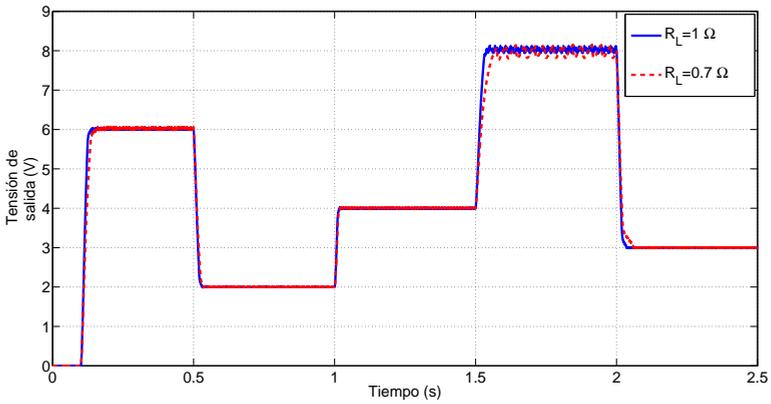


Figura 9.16: Respuesta para $R_L = 1 \Omega$ y $R_L = 0.7 \Omega$.

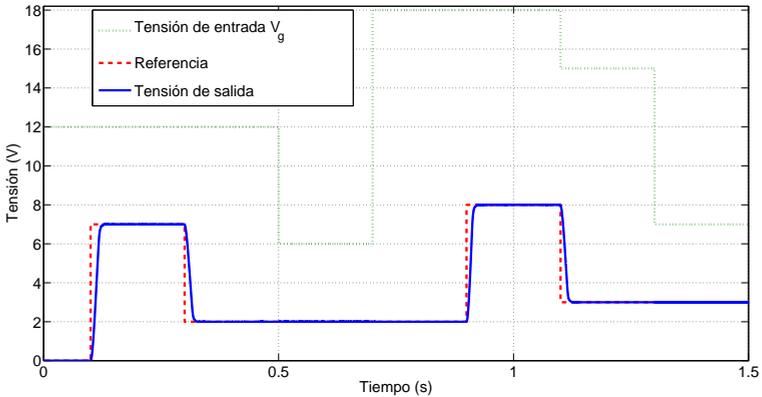


Figura 9.17: Respuesta para $R_L = 1 \Omega$ y $R_L = 0.7 \Omega$.

la tensión de referencia y en el valor de la resistencia de carga. Se comienza con una consigna de 0 V y una resistencia de carga de $4\ \Omega$. En el instante 0.1 ms la consigna cambia a 4 V , permaneciendo la carga en $4\ \Omega$ hasta que en el instante 1 ms la carga cambia a $2\ \Omega$ y en 1.5 ms cambia a $3\ \Omega$. Con la carga de $3\ \Omega$, la consigna baja a 2 V en el instante 2 ms . Por último, en el instante 3.5 ms la carga cambia a $1\ \Omega$, permaneciendo la consigna en 2 V . Como se muestra en la figura 9.18, la tensión de salida sigue a la señal de referencia y las perturbaciones de carga no afectan a la salida. Este comportamiento se ha obtenido para una tensión $V_g = 12\text{ V}$.

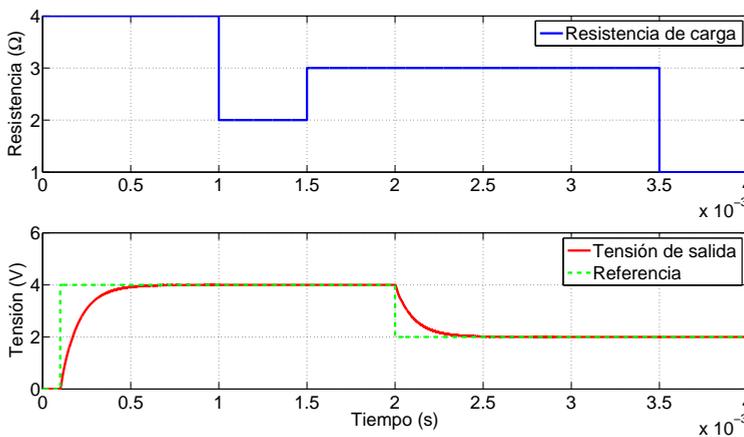


Figura 9.18: Respuesta para variaciones de referencia y de carga.

La figura 9.19 ilustra una ampliación de las transiciones de la figura 9.18 cuando la carga cambia. Tal como se observa, el rechazo a las perturbaciones de carga tiene una respuesta sin sobreimpulsos, tal como era de esperar y las transiciones producen una mínima variación en la tensión de salida. La corriente que circula por la bobina, para el caso presentado en la figura 9.18, se muestra en la figura 9.20. Tal como se puede apreciar, no presenta oscilaciones y las transiciones no tienen sobreimpulsos apreciables.

Sección 9.4

Aportaciones

Este capítulo se ha dedicado al control de un convertidor reductor DC/DC conmutado, siendo las aportaciones las que se resumen en los siguientes puntos:

1. Se han establecido los valores de los parámetros del algoritmo de agru-

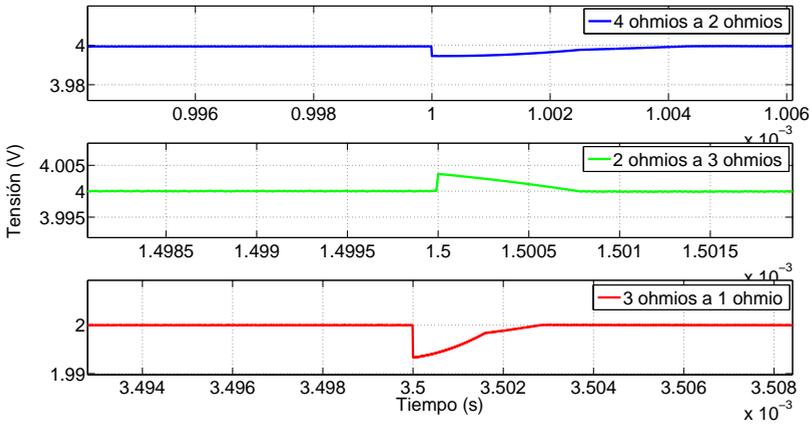


Figura 9.19: Rechazo a las perturbaciones de carga de la figura 9.18.

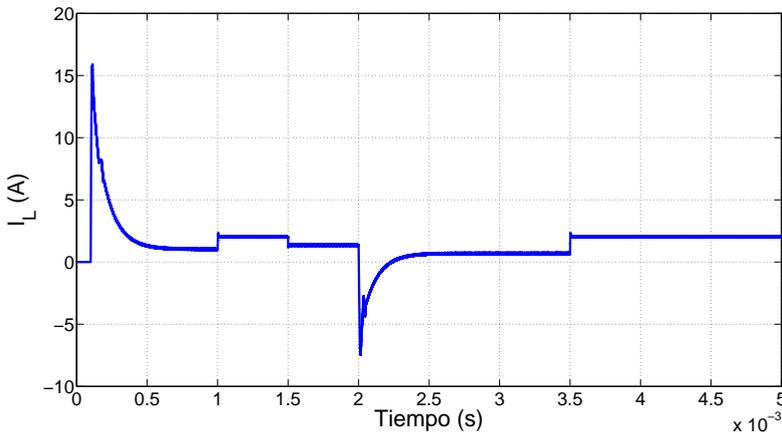


Figura 9.20: Corriente por la bobina I_L para el caso de la figura 9.18.

pamiento para la identificación de un convertidor cuando trabaja en el modo CCM.

2. El controlador inverso ha resultado de implementación muy sencilla ya que está compuesto de un única neurona.
3. El PI borroso ha rechazado las perturbaciones de carga y variaciones de parámetros del sistema de forma eficaz.
4. El sistema resultante ha resultado ser muy estable y robusto.

5. El comportamiento final del sistema controlado ofrece mejores prestaciones que otras técnicas de control más complejas.
6. Se ha planteado y demostrado la validez del sistema propuesto.

Parte del trabajo presentado en este capítulo está publicado en:

Betancor, C. S., Sosa-González, C. J., Montiel, J. A., Vega, A., “Gains Tuning of a PI-Fuzzy controller by genetic algorithms”, *Engineering Computations*, Emerald, Vol. 31, Issue 6, pp 1074-1097, 2014.

Conclusiones y líneas futuras

A lo largo de los diversos capítulos que se han presentado en esta memoria de tesis, se ha analizado la propuesta de diseño planteada, aplicándola a diferentes tipos de plantas con diferentes características, cubriendo una amplia variedad de procesos físicos. A continuación se exponen las conclusiones finales después de haber realizado este trabajo:

1. Se ha propuesto como sistema de control una arquitectura híbrida entre un control FF-FB, un sistema de control basado en modelo de referencia y un control inverso directo con redes neuronales. Los resultados obtenidos en las diferentes aplicaciones han demostrado su eficacia y su alta estabilidad y robustez.
2. El sistema propuesto puede ser aplicado a una amplia gama de plantas. Concretamente, en este trabajo se ha demostrado su eficacia para plantas:
 - Lineales de segundo orden.
 - Lineales de tercer orden que al ser discretizadas se convierten en plantas de fase no mínima.
 - Lineales de primer orden con la no linealidad de saturación.
 - Con no linealidades continuas y suaves.
 - Con no linealidades continuas y duras.
 - Con la no linealidad discontinua denominada zona muerta.
 - No lineales variantes en el tiempo que se puedan representar con un modelo promedio.
3. Debido a que un inverso perfecto no se puede conseguir en la práctica, se ha establecido una metodología con la cual se genera un controlador inverso aproximado, el cual realiza las funciones de controlador de prealimentación.

4. Aparte de conseguir un inverso aproximado, se analiza como conseguir señales de control que en la práctica no requieran gran esfuerzo a la planta.
5. Para estabilizar sistemas de control obtenidos mediante un modelo aproximado de la planta, se puede disminuir el periodo de muestreo del sistema final controlado con respecto al usado en la identificación.
6. En aplicaciones como regulador no es necesario que la salida siga al modelo de referencia, sino que el valor final sea alcanzado en el mínimo tiempo y sin sobreimpulsos. En estos casos, se puede operar con el modelo de referencia para optimizar la respuesta, por ejemplo, variando su tiempo de establecimiento.
7. En varias aplicaciones se mostró que se comportaba mejor un controlador diseñado a partir de un modelo borroso menos exacto que otro a partir de un modelo más exacto. Esto indica que en la identificación lo importante es el tipo de señales con que se excita a la planta y no la exactitud obtenida en la validación.
8. Se establecieron las pautas para diseñar un PI borroso que mitigue los efectos producidos por los errores de modelado y por las perturbaciones.
9. Un algoritmo genético ha optimizado, cuando ha sido necesario, los pesos de la red neuronal que implementa al controlador inverso y las ganancias del PI borroso.
10. Se ha creado una nueva función objetivo, para que el algoritmo genético entrene de forma adecuada al PI borroso, de forma que se rechace sin sobreimpulsos a las perturbaciones de carga.
11. Los resultados obtenidos demuestran una gran estabilidad y robustez del sistema de control obtenido frente a errores de modelado y perturbaciones.
12. Para algunas de las aplicaciones mostradas en los diferentes capítulos, únicamente se ha requerido del controlador inverso, e incluso sólo ha sido necesario una única neurona como controlador. Aún así, el sistema resultante ha presentado una gran estabilidad y robustez.

Tomando como base las conclusiones obtenidas en este trabajo, con el sistema de control diseñado, y continuando con el principio de un sistema de baja complejidad, se pueden citar como líneas futuras de investigación y desarrollo las siguientes:

- Demostrar formalmente que el coste computacional es reducido y que el sistema con la metodología propuesta es implementable en plataformas que no soporten complejidad en el hardware, pudiendo ser aplicable en tiempo real a plantas que posean dinámica rápida.
- Realizar una modificación mínima de la estructura del sistema para poder controlar plantas con retardo de transporte. Este proceso de desarrollo ya se ha iniciado, proponiéndose una modificación del predictor de Smith.
- Ampliar el diseño a otros tipos de no linealidades discontinuas, como por ejemplo, la histéresis.

Bibliografía

- [1] E. N. S. Camperos and A. Y. A. García, *Redes Neuronales. Conceptos Fundamentales y Aplicaciones al Control Automático*. Pearson, Prentice Hall, 2006.
- [2] P. I. Viñuela and I. M. G. León, *Redes Neuronales Artificiales. Un Enfoque Práctico*. Pearson, P. H. (Ed.), 2004.
- [3] B. M. del Brío and A. Sanz-Molina, *Redes Neuronales y Sistemas Borrosos*. Editorial Ra-Ma, 2001.
- [4] M. Nørgaard, O. Ravn, N. K. Poulsen, and L. K. Hansen, *Neural Networks for Modelling and Control of Dynamic Systems*. Springer, 2003.
- [5] O. Omidvar and D. L. Elliott, *Neural Systems for Control*. Academic Press, 1997.
- [6] L. A. Zadeh, “Fuzzy sets,” *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.
- [7] E. H. Mamdani, “Application of fuzzy algorithms for control of simple dynamics plant,” *Proceedings of the Institution of Electrical Engineers*, vol. 121, no. 12, pp. 1585–1588, 1974.
- [8] P. M. Larsen, “Industrial applications of fuzzy logic control,” *International Journal of Man-Machine Studies*, vol. 12, no. 1, pp. 3–10, 1980.
- [9] L. Ljung, *System Identification. Theory for the User*. Prentice Hall PTR, 1999.
- [10] A. A. Behar and M. M. Iranzo, *Identificación y Control Adaptativo*. Hall, P. (Ed.), 2003.
- [11] R. Isermann and M. Münchhof, *Identification of Dynamic Systems: An Introduction with Applications (Advanced Textbooks in Control)*. Springer, 2010.

- [12] D. Gustafson and W. Kessel, "Fuzzy clustering with a fuzzy covariance matrix," in *Proceedings of the IEEE Conference on Decision and Control*, pp. 761–766, January 1979.
- [13] R. Babuska, *Fuzzy Modeling for Control*. Kluwer Academic Publishers, 1998.
- [14] M. Sugeno and T. Yasukawa, "A fuzzy-logic-based approach to qualitative modeling," *IEEE Transactions on fuzzy systems*, vol. 1, no. 1, pp. 7–31, 1993.
- [15] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE transactions on systems, man, and cybernetics*, vol. SMC-15, no. 1, p. 116–132, 1985.
- [16] R. Rovatti, "Takagi-Sugeno models as approximators in Sobolev norms: the SISO case," in *Proceedings of the Fifth IEEE International Conference on Fuzzy Systems, FUZZ-IEEE'96*, pp. 1060–1066, September 1996.
- [17] P. C. Mahalanobis, "On the generalised distance in statistics," *Proceedings of the National Institute of Science of India*, vol. 12, pp. 49–55, 1936.
- [18] R. Babuska, P. J. V. D. Veen, and U. Kaymak, "Improved covariance estimation for Gustafson-Kessel clustering," in *Proceedings of the 2002 IEEE International Conference on Fuzzy Systems*, vol. 2, pp. 1081–1085, May 2002.
- [19] R. Babuska, *Fuzzy Modeling and Identification Toolbox*. Control Engineering Laboratory, Faculty of Information Technology and Systems, Delft University of Technology, Mekelweg 4, P.O. Box 5031, 2600 GA Delft, The Netherlands, August 2001.
- [20] C. Coello, "La importancia de la representación en los algoritmos genéticos (parte I)," *Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios*, vol. 7, no. 69, pp. 50–56, 1999.
- [21] C. Coello, "La importancia de la representación en los algoritmos genéticos (parte II)," *Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios*, vol. 7, no. 70, pp. 44–48, 1999.

-
- [22] Y. Saez, *Análisis de micropoblaciones para la mejora de la eficacia en computación evolutiva interactiva*. PhD thesis, Departamento de lenguajes, sistemas informáticos e ingeniería del software, Facultad de informática (UPM), 2005.
- [23] J. Holland, *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, Michigan, 1975.
- [24] J. E. Baker, “Adaptive selection methods for genetic algorithms,” in *Proceedings of the First International Conference on Genetic Algorithms*, pp. 101–111, 1985.
- [25] A. Wetzel, “Evaluation of the effectiveness of genetic algorithms in combinatorial optimization,” tech. rep., University of Pittsburgh, 1983.
- [26] D. E. Goldberg and K. Deb, “A comparative analysis of selection schemes used in genetic algorithms,” tech. rep., Department of General Engineering, University of Illinois at Urbana-Champaign, 1991.
- [27] T. Blickle and L. Thiele, “A comparison of selection schemes used in genetic algorithms,” tech. rep., Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH), Switzerland, 1995.
- [28] R. Sivaraj and T. Ravichandran, “A review of selection methods in genetic algorithm,” *International Journal of Engineering Science and Technology (IJEST)*, vol. 3, no. 5, p. 3792 – 3797, 2011.
- [29] A. K. D. Jong, *An Analysis of the Behavior of a Class of Genetic Adaptive Systems*. PhD thesis, University of Michigan, 1975.
- [30] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison Wesley Publishing Company, 1989.
- [31] A. H. Wright, “Genetic algorithms for real parameter optimization,” in *Foundations of Genetic Algorithms* (J. E. Rawlins, ed.), pp. 205–218, 1991.
- [32] C. Z. Janikow and Z. Michalewicz, “An experimental comparison of binary and floating point representations in genetic algorithms,” in *Proceedings of the 4th International Conference on Genetic Algorithms, San Diego, CA, USA*, 1991.

- [33] D. M. Tate and A. E. Smith, “Expected allele convergence and the role of mutation in genetic algorithms,” in *Proc. ICGA 5*, pp. 31–37, 1993.
- [34] X. Wang and D. Chen, “Output tracking control of nonminimum phase systems via causal inversion,” in *The 2002 45th Midwest Symposium on Circuits and Systems (MWSCAS), Tulsa, Oklahoma*, vol. 3, pp. 125–128, August 2002.
- [35] Q. Zou and S. Devasia, “Preview-based optimal inversion for output tracking: application to scanning tunneling microscopy,” *IEEE Transactions on Control Systems Technology*, vol. 12, no. 3, pp. 375–386, 2004.
- [36] R. Changhai, S. Lining, R. Weibin, and C. Liguó, “Adaptive inverse control for piezoelectric actuator with dominant hysteresis,” in *Proceedings of the 2004 IEEE International Conference on Control Applications*, vol. 2, pp. 973–976, September 2004.
- [37] W. Daosud, P. Thitiyasook, A. Arpornwichanop, P. Kittisupakorn, and M. A. Hussain, “Neural network inverse model-based controller for the control of a steel pickling process,” *Computers and Chemical Engineering*, vol. 29, no. 10, pp. 2110–2119, 2005.
- [38] R. Gimón and R. Lamanna, “Implantación de un esquema de control neuronal inverso en una planta de neutralización,” in *V Congreso de Automatización y Control (CAC), Caracas, Venezuela*, 2005.
- [39] B. Rigney, L. Pao, and D. Lawrence, “Model inversion architectures for settle time applications with uncertainty,” in *Proceedings of the 45th IEEE Conference on Decision and Control, San Diego, CA*, pp. 6518–6524, December 2006.
- [40] J. A. Butterworth, L. Y. Pao, and D. Y. Abramovitch, “Architectures for tracking control in atomic force microscopes,” in *Proceedings of the 17th IFAC World Congress, Seoul, Korea*, vol. 17, pp. 8236–8250, July 2008.
- [41] B. Rigney, L. Pao, and D. Lawrence, “Nonminimum phase dynamic inversion for settle time applications,” *IEEE Transactions on Control Systems Technology*, vol. 17, no. 5, pp. 989–1005, 2009.
- [42] B. Rigney, L. Pao, and D. Lawrence, “Adaptive inverse control for settling performance improvements,” in *American Control Conference (ACC), St. Louis, Missouri, USA*, pp. 190–197, June 2009.

-
- [43] S. Toha and M. Tokhi, "Inverse model based control for a twin rotor system," in *IEEE 9th International Conference on Cybernetic Intelligent Systems*, pp. 1–5, September 2010.
- [44] J. Skutová, "Adaptive control with inverse model of a compressed-air aggregate," in *12th International Carpathian Control Conference (ICCC), Velke Karlovice*, pp. 389–393, May 2011.
- [45] A. S. Puydupin-Jamin, M. Johnson, and T. Bretl, "A convex approach to inverse optimal control and its application to modeling human locomotion," in *IEEE International Conference on Robotics and Automation (ICRA), Saint Paul, MN*, pp. 531–536, May 2012.
- [46] B. S. Leon, A. Y. Alanis, E. N. Sanchez, F. Ornelas-Tellez, and E. Ruiz-Velazquez, "Neural inverse optimal control applied to type 1 diabetes mellitus patients," in *IEEE Third Latin American Symposium on Circuits and Systems (LASCAS), Playa del Carmen*, pp. 1–4, February-March 2012.
- [47] N. Aghasadeghi, A. Long, and T. Bretl, "Inverse optimal control for a hybrid dynamical system with impacts," in *IEEE International Conference on Robotics and Automation (ICRA), Saint Paul, MN*, pp. 4962–4967, May 2012.
- [48] A. Maghsoudi and M. Jahed, "Inverse dynamics control of needle in prostate brachytherapy," in *IEEE International Conference on Industrial Technology (ICIT), Athens*, pp. 510–515, March 2012.
- [49] J. Mamboundou and N. Langlois, "Application of feedforward model inverse control using MPC on a non-minimum phase system," in *IEEE International Conference on Industrial Technology (ICIT), Athens*, pp. 899–905, March 2012.
- [50] X. Cai, "Finite time inverse optimal control of affine nonlinear systems," in *24th Chinese Control and Decision Conference (CCDC), Taiyuan*, pp. 2328–2332, May 2012.
- [51] G. Ramsingh and B. B. Sharma, "Synchronization of chaotic system using U-model based adaptive inverse control," in *Students Conference on Engineering and Systems (SCES), Allahabad, Uttar Pradesh*, pp. 1–5, March 2012.

- [52] S. Das, K. Halder, I. Pan, S. Ghosh, and A. Gupta, "Inverse optimal control formulation for guaranteed dominant pole placement with PI/PID controllers," in *International Conference on Computer Communication and Informatics (ICCCI), Coimbatore*, pp. 1–6, January 2012.
- [53] K. V. Zmeu, B. S. Notkin, P. A. Dyachenko, and V. A. Kovalev, "Fast predictive inverse neurocontrol: Comparative simulation and experiment," in *International Joint Conference on Neural Networks (IJCNN), Brisbane, QLD*, pp. 1–7, June 2012.
- [54] F. Ornelas-Tellez, E. N. Sanchez, and A. G. Loukianov, "Discrete-time neural inverse optimal control for nonlinear systems via passivation," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 8, pp. 1327–1339, 2012.
- [55] B. Widrow and E. Walach, *Adaptive inverse control*. IEEE PRESS, WILEY-INTERSCIENCE, 2008.
- [56] S. Devasia, D. Chen, and B. Paden, "Nonlinear inversion-based output tracking," *IEEE Transactions on Automatic Control*, vol. 41, pp. 930–942, July 1996.
- [57] Q. Zou and S. Devasia, "Preview-based stable-inversion for output tracking," in *Proceedings of the 1999 American Control Conference, San Diego, CA*, vol. 5, pp. 3544–3548, June 1999.
- [58] M. Tomizuka, "Zero phase error tracking algorithm for digital control," *ASME Journal of Dynamic Systems Measurement and Control*, vol. 109, no. 1, pp. 65–68, 1987.
- [59] J. Wen and B. Potsaid, "An experimental study of a high performance motion control system," in *American Control Conference*, pp. 5158–5163, June 2004.
- [60] E. Gross and M. Tomizuka, "Experimental beam tip tracking control with a truncated series approximation to uncanceled inverse dynamics," *IEEE Transactions on Control Systems Technology*, vol. 2, pp. 382–391, December 1994.
- [61] C. Scali, M. Hvala, and D. R. Lewin, "Robustness issues in feedforward control," in *American Control Conference*, pp. 577–581, June 1989.
- [62] K. M. Tao, R. L. Kosut, and G. Aral, "Learning feedforward control," in *American Control Conference*, vol. 3, pp. 2575–2579, June-July 1994.

-
- [63] C. Brosilow and B. Joseph, *Techniques of Model-Based Control*. Prentice Hall PTR, 2002.
- [64] A. Fowler, “Feedback and feedforward as systemic frameworks for operations control,” *International Journal of Operations and Production Management*, vol. 19, no. 2, pp. 182–204, 1999.
- [65] D. Li, H. Wang, and W. Gai, “Application of a feedforward controller with a disturbance observer for UAV during missile launch,” in *2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC)*, pp. 2811–2815, August 2011.
- [66] N. Parikh, S. Patwardhan, S. Bandyopadhyay, and R. Gudi, “Control of a nuclear steam generator using feedback-feedforward LQG controller,” in *4th International Symposium on Advanced Control of Industrial Processes*, pp. 415–420, May 2011.
- [67] N. Wang and K. E. Johnson, “LIDAR-based FX-RLS feedforward control for wind turbine load mitigation,” in *American Control Conference*, pp. 1910–1915, June 2011.
- [68] D. Schlipf, L. Y. Pao, and P. W. Cheng, “Comparison of feedforward and model predictive control of wind turbines using LIDAR,” in *51st IEEE Conference on Decision and Control*, pp. 3050–3055, December 2012.
- [69] N. Wang, K. E. Johnson, and A. D. Wright, “FX-RLS-based feedforward control for LIDAR-enabled wind turbine load mitigation,” *IEEE Transactions on Control Systems Technology*, vol. 20, no. 5, pp. 1212–1222, 2012.
- [70] H. Zhong, L. Pao, and R. de Callafon, “Feedforward control for disturbance rejection: Model matching and other methods,” in *24th Chinese Control and Decision Conference (CCDC)*, pp. 3528–3533, May 2012.
- [71] M. Gulan, M. Salaj, and B. Rohal’-Ilkiv, “Real-time implementation of an adaptive feedback and feedforward generalized predictive controller,” in *International Conference on Process Control (PC), Strbske Pleso*, pp. 383–388, June 2013.
- [72] J. Elso, M. G. Martínez, and M. G. Sanz, “Quantitative feedback–feedforward control for model matching and disturbance rejection,” *IET Control Theory and Applications*, vol. 7, no. 6, pp. 894–900, 2013.

- [73] P. Hägg and B. Wahlberg, "On optimal input design for feed-forward control," in *52nd IEEE Conference on Decision and Control, Florence, Italy*, pp. 7174–7180, December 2013.
- [74] M. Veronesi and A. Visioli, "Automatic feedforward tuning for PID control loops," in *European Control Conference (ECC), Zurich*, pp. 3919–3924, July 2013.
- [75] B. Wang and H. Liu, "Control system of paper cutting machine based on motor feedforward and fuzzy PID strategy," in *Proceeding of the IEEE International Conference on Information and Automation, Yinchuan*, pp. 587–591, August 2013.
- [76] N. Wang, K. E. Johnson, and A. D. Wright, "Comparison of strategies for enhancing energy capture and reducing loads using LIDAR and feed-forward control," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 4, pp. 1129–1142, 2013.
- [77] H. Zhong and L. Y. Pao, "H2 model matching feedforward control for tape head positioning servo systems," in *American Control Conference (ACC)*, pp. 4504–4509, June 2013.
- [78] Y. Xu and K. Mohseni, "Bioinspired hydrodynamic force feedforward for autonomous underwater vehicle control," *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 4, pp. 1127–1137, 2014.
- [79] M. Heertjes and D. Bruijnen, "MIMO FIR feedforward design for zero error tracking control," in *American Control Conference (ACC)*, pp. 2166–2171, June 2014.
- [80] H. Ismail, N. Ishak, M. Tajjudin, M. H. F. Rahiman, and R. Adnan, "Feedforward control for high precision linear servo system," in *IEEE Control and System Graduate Research Colloquium*, pp. 119–123, June 2011.
- [81] H. Ismail, N. Ishak, M. Tajjudin, F. Rahiman, and R. Adnan, "Adaptive feedforward zero phase error tracking control with model reference for high precision X-Y table," in *4th International Conference on Intelligent and Advanced Systems (ICIAS2012)*, pp. 526–530, June 2012.
- [82] H. Ismail, R. Adnan, N. Ishak, M. Tajjudin, and M. H. F. Rahiman, "A study on feedforward tracking control for XY table by real-time experiments," in *SICE Annual Conference*, pp. 1205–1210, September 2013.

-
- [83] R. Adnan, H. Ismail, N. Ishak, M. Tajjudin, and F. Rahiman, "Adaptive feedforward zero phase error tracking control for minimum phase and non-minimum phase systems – XY table real-time application," in *IEEE Control and System Graduate Research Colloquium (ICSGRC 2012)*, pp. 359–363, July 2012.
- [84] B. J. Lin, G. Yu, S. H. Yang, S. Q. Kou, and J. H. Wang, "Application of a new type feedforward-feedback controller in assembly machine for assembled camshaft," *Assembly Automation*, vol. 32, no. 4, pp. 333–339, 2012.
- [85] M. K. Mohamed, W. P. Heath, and A. Lanzon, "Feedforward/feedback multivariable control design for high speed nanopositioning," in *European Control Conference (ECC)*, pp. 1939–1944, June 2014.
- [86] H. Butler, "Adaptive feedforward for a wafer stage in a lithographic tool," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 3, pp. 875–881, 2013.
- [87] F. S. Panni, H. Waschl, D. Alberer, and L. Zaccarian, "Position regulation of an EGR valve using reset control with adaptive feedforward," *IEEE Transactions on Control Systems Technology*, vol. 22, pp. 2424–2431, November 2014.
- [88] L. Zaccarian, D. Nesic, and A. R. Teel, "First order reset elements and the clegg integrator revisited," in *Proceedings of the American Control Conference*, vol. 1, pp. 563–568, June 2005.
- [89] J. Tsai, C. R. Koch, and M. Saif, "Cycle adaptive feedforward approach controllers for an electromagnetic valve actuator," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 3, pp. 738–746, 2012.
- [90] W. Lizhi, T. Si, C. Ping, and H. Guangjun, "Combined feedforward-feedback servo control scheme for a radar spy azimuth follow-up system," in *International Conference on Electronic and Mechanical Engineering and Information Technology*, pp. 4196–4198, August 2011.
- [91] M. Mailah, S. Abdullah, and T. H. Hing, "Tracking performance on feedforward model base active force control of mobile manipulator using Matlab and Adams," *Recent Advances in Electrical Engineering Series*, no. 11, pp. 83–88, 2013.

- [92] D. Tian, H. Shen, and M. Dai, "Improving the rapidity of nonlinear tracking differentiator via feedforward," *IEEE Transactions on Industrial Electronics*, vol. 61, pp. 3736–3743, July 2014.
- [93] Y. Altun, K. Gulez, and T. V. Mumcu, "Static LPV feedforward controller synthesis for linear parameter varying systems," in *9th Asian Control Conference (ASCC)*, pp. 1–4, June 2013.
- [94] Y. Altun, K. Gulez, T. V. Mumcu, and M. Kizilkaya, "A novel feedforward controller for PMSMs," in *3rd International Conference on Electric Power and Energy Conversion Systems*, pp. 1–5, October 2013.
- [95] Y. Lee, D. Shin, and C. C. Chung, "PID controller with feedforward low pass filters for permanent magnet stepper motors," in *12th International Conference on Control, Automation and Systems*, pp. 1597–1600, October 2012.
- [96] A. Watanabe, A. Ohya, and S. Yuta, "Feedforward current control system design for efficient drive method of small-sized brushless DC motor," in *37th Annual Conference on IEEE Industrial Electronics Society*, pp. 575–580, November 2011.
- [97] H. Tanaka and M. Hasegawa, "Improvements of speed adaptation response for IPMSM position sensorless vector control with parallel feedforward compensator," in *14th European Conference on Power Electronics and Applications (EPE 2011)*, pp. 1–10, August 2011.
- [98] X. Ling, Q. Li, T. Wang, J. Dong, Z. Tang, and Y. Ding, "Research and design of motion controller for CNC based on fuzzy PID algorithm with feedforward control," in *International Conference on Control, Automation and Systems Engineering (CASE)*, pp. 1–4, July 2011.
- [99] L. M. Grzesiak and T. Tarczewski, "PMSM servo-drive control system with a state feedback and a load torque feedforward compensation," *The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, vol. 32, no. 1, pp. 364 – 382, 2012.
- [100] M. Rakotondrabe, J. Agnus, and P. Lutz, "Feedforward and IMC-feedback control of a nonlinear 2-DOF piezoactuator dedicated to automated micropositioning tasks," in *IEEE International Conference on Automation Science and Engineering*, pp. 393–398, August 2011.

- [101] M. Rakotondrabe, K. Rabenorosoa, J. Agnus, and N. Chaillet, “Robust feedforward-feedback control of a nonlinear and oscillating 2-DOF piezo-cantilever,” *IEEE Transactions on Automation Science and Engineering*, vol. 8, pp. 506–519, July 2011.
- [102] Y. Liu and J. Shan, “Feedback/Feedforward control of hysteresis-compensated piezoactuators for highspeed scanning applications,” in *IEEE 23rd International Symposium on Industrial Electronics (ISIE)*, pp. 281–286, June 2014.
- [103] J.-W. Lian and H.-Y. Chen, “Feedforward and feedback control for piezoelectric- actuated systems using inverse Prandtl-Ishlinskii model and particle swarm optimization,” in *International Conference on Advanced Mechatronic Systems*, pp. 313–318, August 2014.
- [104] M. Mohammadzaheri, S. Grainger, M. K. Kopaei, and M. Bazghaleh, “IMC-based feedforward control of a piezoelectric tube actuator,” in *IEEE Eighth International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pp. 357–361, April 2013.
- [105] X. Wang, G. Alici, and X. Tan, “Modeling and inverse feedforward control for conducting polymer actuators with hysteresis,” *Smart Materials and Structures* *Email alert RSS feed*, vol. 23, no. 2, pp. 1–9, 2014.
- [106] J. T. Wen, S. Mishra, S. Mukherjee, N. Tantisujjatham, and M. Minakais, “Building temperature control with adaptive feedforward,” in *IEEE Conference on Decision and Control*, pp. 4827–4832, December 2013.
- [107] D. Fei and Z. Yongjie, “Research on a new feedforward controller of main steam temperature system,” in *IEEE International Conference on Information and Automation (ICIA)*, pp. 295–300, July 2014.
- [108] A. Mittal, P. Jain, T. K. Saxena, and A. Kapoor, “Genetic algorithm based neuro fuzzy tuning of PID controller for a nonlinear temperature water bath with feedforward control,” in *International Conference on Advanced Electronic Systems (ICAES)*, pp. 34–38, September 2013.
- [109] K. Kora, C. Z. Lu, and A. J. McDaid, “Automatic tuning with feedforward compensation of the HuREx rehabilitation system,” in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 1504–1509, July 2014.

- [110] C. Casellato, A. Pedrocchi, J. A. Garrido, N. R. Luque, G. Ferrigno, E. D'Angelo, and E. Ros, "An integrated motor control loop of a human-like robotic arm: feedforward, feedback and cerebellum-based learning," in *The Fourth IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechanics*, pp. 562–567, June 2012.
- [111] S. Hong, Y. Oh, D. Kim, and B.-J. You, "Real-time walking pattern generation method for humanoid robots by combining feedback and feedforward controller," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 1, pp. 355–364, 2014.
- [112] M. Plooij, M. de Vries, W. Wolfslag, and M. Wisse, "Optimization of feedforward controllers to minimize sensitivity to model inaccuracies," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3382–3389, November 2013.
- [113] N. O. Pérez-Arancibia, J. P. Whitney, and R. J. Wood, "Lift force control of flapping-wing microrobots using adaptive feedforward schemes," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 1, pp. 155–168, 2013.
- [114] K. Ebrahimi, A. Schramm, and C. R. Koch, "Feedforward/Feedback control of HCCI combustion timing," in *American Control Conference (ACC)*, pp. 831–836, June 2014.
- [115] F. Zhao, D. N. Assanis, T. N. Asmus, J. E. Dec, J. A. Eng, and P. M. Najt, *Homogeneous Charge Compression Ignition (HCCI) Engines*. SAE International, 2003.
- [116] Y. Qi, L. Peng, Z. Huang, M. Chen, and L. Sun, "Feedforward control of output current for three-phase voltage source inverter (VSI) with transformer," in *Applied Power Electronics Conference and Exposition (APEC), Twenty-Ninth Annual IEEE*, pp. 2318–2322, March 2014.
- [117] Y.-H. Liao, "A novel reduced switching loss bidirectional AC/DC converter PWM strategy with feedforward control for grid-tied microgrid systems," *IEEE Transactions on Power Electronics*, vol. 29, pp. 1500–1513, March 2014.
- [118] S. M. Park, Y. D. Lee, and S.-Y. Park, "Voltage sensorless feedforward control of a dual boost PFC converter for battery charger applications," in *IEEE Energy Conversion Congress and Exposition (ECCE)*, pp. 1376–1380, September 2011.

-
- [119] K. I. Hwu and Y. T. Yau, "Applying one-comparator counter-based PWM control strategy to DC-AC converter with voltage reference feed-forward control considered," in *Twenty-Sixth Annual IEEE Applied Power Electronics Conference and Exposition (APEC)*, pp. 1811–1815, March 2011.
- [120] B. Wang, P. Jia, T. Q. Zheng, and Y. Li, "Research on stability of buck converter with output-current-feedforward control," in *Twenty-Ninth Annual IEEE Applied Power Electronics Conference and Exposition (APEC)*, pp. 4188–4192, September 2014.
- [121] T. Watanabe, T. Nabeshima, T. Sato, and K. Nishijima, "Buck-boost converter controlled by hysteretic PWM method with input voltage feed-forward," in *IEEE 33rd International Telecommunications Energy Conference (INTELEC)*, pp. 1–6, October 2011.
- [122] C. Yao, X. Ruan, and X. Wang, "Automatic mode-shifting control strategy with input voltage feed-forward for full-bridge-boost DC-DC converter suitable for wide input voltage range," *IEEE Transactions on Power Electronics*, vol. 30, no. 3, pp. 1668–1682, 2015.
- [123] H. C. Foong, Y. Z. Y. K. Tan, and M. T. Tan, "Fast-transient integrated digital DC-DC converter with predictive and feedforward control," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 59, no. 7, pp. 1567–1576, 2012.
- [124] Y. Bing, L. Xinquan, and Z. Lingling, "Novel dynamic ramp circuit with input feedforward for voltage-mode DC-DC buck converter," in *IEEE International Symposium on Radio-Frequency Integration Technology*, pp. 1–3, August 2014.
- [125] B.-J. Park, K.-Y. Choi, and R.-Y. Kim, "An active feedforward compensation for a current harmonics reduction in three-phase grid-connected inverters," in *39th Annual Conference of the IEEE Industrial Electronics Society, IECON*, pp. 400–405, November 2013.
- [126] J. Wang, Y. Song, and A. Monti, "A study of feedforward control on stability of grid parallel inverter with various grid impedance," in *IEEE 5th International Symposium on Power Electronics for Distributed Generation Systems (PEDG)*, pp. 1–8, June 2014.
- [127] M. D. Pedroso, C. B. Nascimento, M. D. S. Kaster, and A. M. Tuset, "An optimal pole placement state feedback with feedforward digital control

- applied to a three-level NPC inverter implemented in FPGA,” in *IEEE 23rd International Symposium on Industrial Electronics (ISIE)*, pp. 52–57, June 2014.
- [128] J. C. S. Ganacim, M. B. Meier, A. A. Badin, R. Gules, E. F. R. Romaneli, and A. A. Assef, “Output power feedforward technique applied to a high power factor rectifier with high frequency transformer,” in *Brazilian Power Electronics Conference (COBEP)*, pp. 406–413, October 2013.
- [129] W. ZHuo, J. Yanyan, and W. SHichao, “The application of feedforward PID control in water level control system,” in *World Automation Congress*, pp. 1–3, June 2012.
- [130] M. Alma, J. J. Martinez, I. D. Landau, and G. Buche, “Design and tuning of reduced order H-infinity feedforward compensators for active vibration control,” *IEEE Transactions on Control Systems Technology*, vol. 20, pp. 554–561, March 2012.
- [131] S.-Y. Han, Y.-H. Chen, K. Ma, D. Wang, A. Abraham, and Z.-G. Liu, “Feedforward and feedback optimal vibration rejection for active suspension discrete-time systems under in-vehicle networks,” in *Sixth World Congress on Nature and Biologically Inspired Computing (NaBIC)*, pp. 139–144, July–August 2014.
- [132] D. Tran and Y. K. Tan, “Sensorless illumination control of a networked LED-lighting system using feedforward neural network,” *IEEE Transactions on Industrial Electronics*, vol. 61, no. 4, pp. 2113–2121, 2014.
- [133] M. Beschi, S. Dormido, J. Sanchez, A. Visioli, and L. J. Yebra, “Event-based PI plus feedforward control strategies for a distributed solar collector field,” *IEEE Transactions on Control Systems Technology*, vol. 22, no. 4, pp. 1615–1622, 2014.
- [134] J. Liang, D. F. Howard, J. A. Restrepo, and R. G. Harley, “Feedforward transient compensation control for DFIG wind turbines during both balanced and unbalanced grid disturbances,” *IEEE Transactions on Industry Applications*, vol. 49, no. 3, pp. 1452–1463, 2013.
- [135] A. Koerber and R. King, “Combined feedback–feedforward control of wind turbines using state-constrained model predictive control,” *IEEE Transactions on control Systems Technology*, vol. 21, no. 4, pp. 1117–1128, 2013.

-
- [136] Y. Rabhi, M. Mrabet, F. Fnaiech, and P. Gorce, "A feedforward neural network wheelchair driving joystick," in *International Conference on Electrical Engineering and Software Applications (ICEESA)*, pp. 1–6, March 2013.
- [137] P. S. Thakar, B. Bandyopadhyay, and P. S. Gandhi, "Sliding mode control for a class of underactuated systems using feedforward normal form: A slosh-container system," in *13th IEEE Workshop on Variable Structure Systems, VSS'14*, pp. 1–6, June-July 2014.
- [138] J. Shin, H. J. Kim, Y. Kim, and W. E. Dixon, "Autonomous flight of the rotorcraft-based UAV using RISE feedback and NN feedforward terms," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 5, pp. 1392–1399, 2012.
- [139] N. H. Shariati, A. Maleki, and A. Fallah, "Feedforward-feedback P-PID control of elbow joint angle for functional electrical stimulation: A simulation study," in *2nd International Conference on Control, Instrumentation and Automation (ICCIA)*, pp. 156–161, December 2011.
- [140] L. D. Novellis, A. Sorniotti, P. Gruber, and A. Pennycott, "Comparison of feedback control techniques for torque-vectoring control of fully electric vehicles," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 8, pp. 3612–3623, 2014.
- [141] Y. S. Neo, S. M. Idrus, M. F. Rahmat, S. E. Alavi, and I. S. Amiri, "Adaptive control for laser transmitter feedforward linearization system," *IEEE Photonics Journal*, vol. 6, no. 4, 2014.
- [142] T. Nakazaki, N. Sakimura, T. Ohashi, K. Ohishi, and T. Miyazaki, "Minimum trajectory control of focusing and tracking of optical disk systems by two-dimensional feedforward compensator," in *39th Annual Conference of the IEEE Industrial Electronics Society*, pp. 6490–6495, November 2013.
- [143] G. Li, J. Na, D. P. Stoten, and X. Ren, "Adaptive neural network feedforward control for dynamically substructured systems," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 3, pp. 944–954, 2014.
- [144] J. M. Lemos, L. F. Pinto, L. M. Rato, and M. Rijo, "Distributed LQG control of a water delivery canal with feedforward from measured consumptions," in *20th Mediterranean Conference on Control and Automation (MED)*, pp. 722–727, July 2012.

- [145] T. Kikuchi, H. Taga, and R. Takagi, “Fuzzy feed-forward charge/discharge control of stationary energy storage systems for DC electric railways,” in *The 2014 International Power Electronics Conference*, pp. 1328–1333, May 2014.
- [146] S. John, “Artificial intelligent-based feedforward optimized PID wheel slip controller,” in *AFRICON*, pp. 1–6, September 2013.
- [147] A. T. Elgammal, A. A. Abouelsoud, and S. F. M. Assal, “Fuzzy logic-based gain scheduling of exact feedforward linearization controller for magnetic ball levitation system,” in *UKACC International Conference on Control*, pp. 531–536, July 2014.
- [148] Y. Bai, X. Gong, Z. Hou, and Y. Tian, “Stability control of quad-rotor based on explicit model following with inverse model feedforward method,” in *International Conference on Mechatronics and Automation*, pp. 2189–2194, August 2011.
- [149] P. Osburn, H. Whitaker, and A. Kezer, “New developments in the design of adaptive control systems,” in *IAS 29th Annual Meeting*, no. 61-39, January 1961.
- [150] Z. Haishui, W. Dahu, Z. Tong, and H. Keming, “Design on a DC motor speed control,” in *International Conference on Intelligent Computation Technology and Automation*, vol. 2, pp. 59–63, May 2010.
- [151] A. Rajasekhar, A. Abraham, P. Kunathi, and M. Pant, “Fractal order speed control of DC motor using levy mutated artificial bee colony algorithm,” in *World Congress on Information and Communication Technologies (WICT)*, pp. 7–13, December 2011.
- [152] K. Ogata, *Sistemas de Control en Tiempo Discreto*. Pearson Educación, 1996.
- [153] K. Ogata, *Ingeniería de Control Moderna*. Pearson, Prentice Hall, 2003.
- [154] T. Liu and F. Gao, “New insight into internal model control filter design for load disturbance rejection,” *Control Theory and Applications, IET*, vol. 4, no. 3, pp. 448–460, 2010.
- [155] A. Visioli, “Improving the load disturbance rejection performances of IMC-tuned PID controllers,” in *15th IFAC Triennial World Congress on Automatic Control, Barcelona, Spain*, vol. 15, July 2002.

-
- [156] V. Zlosnikas and A. Baskys, "PID controller with enhanced disturbance rejection," *Electronics and Electrical Engineering*, vol. 5, pp. 65–68, 2008.
- [157] J. Jantzen, "Tuning of fuzzy PID controllers," Tech. Rep. 98-H 871, Technical Univ. of Denmark: Dept. of Automation, 1998.
- [158] H. Ying, "A general technique for deriving analytical structure of fuzzy controllers using arbitrary trapezoidal input fuzzy sets and zadeh AND operator," *Automatica*, vol. 39, pp. 1171–1184, July 2003.
- [159] X. Liu, R. Kuai, P. Guan, X. Ye, and Z. Wu, "Fuzzy-PID control for ARC furnace electrode regulator system based on genetic algorithm," in *International Conference on Machine Learning and Cybernetics*, vol. 2, pp. 683–689, July 2009.
- [160] I. Saaki, P. Babu, C. Rao, and D. Prasad, "Integral square error minimization technique for linear multi input and multi output systems," in *International Conference on Power and Energy Systems (ICPS)*, pp. 1–5, December 2011.
- [161] V. Shekher, P. Rai, and O. Prakash, "Design and evaluation of classic PID, gain and phase margin based controller and intelligent controller design for a ceramic infrared heater," *ARPJN Journal of Science and Technology*, vol. 3, no. 3, pp. 248–256, 2012.
- [162] M. Antonelli, P. Ducange, B. Lazzerini, and F. Marcelloni, "Exploiting a three-objective evolutionary algorithm for generating mamdani fuzzy rule-based systems," in *2010 IEEE International Conference on Fuzzy Systems (FUZZ)*, Barcelona, pp. 1–8, July 2010.
- [163] S. Chopra, R. Mitra, and V. Kumar, "Auto tuning of fuzzy PI type controller using fuzzy logic," *International Journal of Computational Cognition*, vol. 6, no. 1, pp. 12–18, 2008.
- [164] L. Gan, Y. Pang, Q. Liao, and N. Xiao, "Fuzzy criticality assessment of FMECA for the SADA based on modified FWGM algorithm & centroid defuzzification," in *2011 International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering (ICQR2MSE)*, Xi'an, pp. 195–202, June 2011.
- [165] A. Chipperfield, P. Fleming, H. Pohlheim, and C. Fonseca, *Genetic Algorithm Toolbox*. Department of Automatic Control and Systems Engineering, University of Sheffield, 1995.

- [166] H. Muhlenbein, “The breeder genetic algorithm – a provable optimal search algorithm and its application,” in *IEE Colloquium on Applications of Genetic Algorithms*, pp. 5/1–5/3, March 1994.
- [167] S. Hamamci and M. Koksak, “Robust control of a DC motor by coefficient diagram method,” in *9th Mediterranean Conference on Control and Automation (MED’01), Dubrovnik, Croatia*, vol. 148, pp. 85–90, June 2001.
- [168] S. Manabe, “Coefficient diagram method,” in *14th IFAC Symposium on Automatic Control in Aerospace, Seoul, Korea*, pp. 199–210, August 1998.
- [169] S. Manabe and Y. C. Kim, “Recent development of coefficient diagram method,” in *3th Asian Control Conference*, pp. 2055–2060, July 2000.
- [170] J. F. M. Amaral, M. M. Vellasco, R. Tanscheit, and M. A. C. Pacheco, “A neuro-fuzzy-genetic system for automatic setting of control strategies,” in *Proceedings of the International Conference on IFSA World Congress and 20th NAFIPS*, vol. 3, pp. 1553–1558, July 2001.
- [171] H. Sira-Ramírez, *Control de sistemas no lineales*. Pearson Educación, 2005.
- [172] G. Fedele, “A new method to estimate a first-order plus time delay model from step response,” *Journal of the Franklin Institute*, vol. 346, no. 1, pp. 1–9, 2009.
- [173] J. E. Normey-Rico, *Control of Dead-time Processes*. Ed. Springer, 2007.
- [174] P. Aravind, M. Valluvan, and S. Ranganathan, “Modelling and simulation of non linear tank,” *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 2, no. 2, pp. 842–849, 2013.
- [175] M. A. Johnson and M. H. Moradi, *PID Control: New Identification and Design Methods*. Ed. Springer-Verlag, London Limited, 2005.
- [176] D. Chen and D. E. Seborg, “PI/PID controller design based on direct synthesis and disturbance rejection,” *Industrial and Engineering Chemistry Research*, vol. 41, no. 19, pp. 4807–4822, 2002.
- [177] S. Skogestad, “Probably the best simple PID tuning rules in the world,” *Journal of Process Control*, vol. 13, no. 4, pp. 291–309, 2003.

-
- [178] S. Anand, V. Aswin, and S. R. Kumar, "Simple tuned adaptive PI controller for conical tank process," in *International Conference on Recent Advancements in Electrical, Electronics and Control Engineering, Sivakasi*, pp. 263–267, December 2011.
- [179] K. J. Astrom and B. Wittenmark, *Adaptive Control*. Dover Publications, second ed., 2008.
- [180] J. G. Ziegler and N. B. Nichols, "Optimum settings for automatic controllers," *Transactions of the American Society of Mechanical Engineers*, vol. 64, pp. 759–768, 1942.
- [181] P. Sowmya, N. Srivignesh, N. Sivakumaran, and G. Balasubramanian, "A fuzzy control scheme for nonlinear process," in *IEEE International Conference on Advances in Engineering, Science and Management (ICAESM-2012), Nagapattinam, Tamil Nadu*, vol. 3, pp. 683–688, March 2012.
- [182] T. L. Grigorie, *Fuzzy Controllers, Theory and Applications*. Ed. Lucian Grigorie, Publisher InTech, 2011.
- [183] I. L. Chien and P. S. Fruehauf, "Consider IMC tuning to improve controller performance," *Chemical Engineering Progress*, vol. 86, pp. 33–41, 1990.
- [184] R. Anandanatarajan, M. Chidambaram, and T. Jayasingh, "Improved design of FLC for a first order nonlinear processes with dead time," in *Proceedings of International Conference on Intelligent Sensing and Information Processing*, pp. 466 – 471, 2004.
- [185] P. Aravind and M. Valluvan, "Analysis of fuzzy based level controller for non-linear process tank," *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 3, no. 1, pp. 7015–7019, 2014.
- [186] A. Ganesh and S. Abraham, "Fuzzy adaptive PI controller for single input single output non-linear system," *ARPJ Journal of Engineering and Applied Sciences*, vol. 7, no. 10, pp. 1273–1280, 2012.
- [187] M. Esfandyari, M. Fanaei, and H. Zohreie, "Adaptive fuzzy tuning of PID controllers," *Neural Computing and Applications, Supplement*, vol. 23, no. 1, pp. 19–28, 2013.

- [188] A. Ganesh and S. Abraham, "A model reference-based fuzzy adaptive PI controller for non-linear level process system," *International Journal of Research and Reviews in Applied Sciences (IJRRAS)*, vol. 14, no. 2, pp. 477–486, 2013.
- [189] H. Butler, *Model Reference Adaptive Control: From Theory to Practice*. Prentice Hall International Series in Systems and Control Engineering, Prentice Hall College Div, 1992.
- [190] R. Valarmathi, P. R. Theerthagiri, and S. Rakeshkumar, "Design and analysis of genetic algorithm based controllers for non linear liquid tank system," in *IEEE-International Conference On Advances In Engineering, Science And Management (ICAESM -2012)*, pp. 616–620, 2012.
- [191] P. Aravind and S. M. GirirajKumar, "Performance optimization of PI controller in non linear process using genetic algorithm," *International Journal of Current Engineering and Technology, International Press Corporation*, vol. 3, no. 5, pp. 1968–1972, 2013.
- [192] D. J. Leith and W. E. Leithead, "Survey of gain-scheduling analysis & design," *International Journal of Control*, vol. 73, no. 11, pp. 1001–1025, 2000.
- [193] R. Sukanya and V. Sivanandam, "Design of controllers based on MPC for a conical tank system," in *International Conference On Advances In Engineering, Science And Management (ICAESM -2012)*, pp. 309–313, March 2012.
- [194] N. Venkatesan and N. Anantharaman, "Controller design based on model predictive control for a nonlinear process," in *8th International Symposium on Mechatronics and its Applications (ISMA)*, Sharjah, pp. 1–6, April 2012.
- [195] R. Shridhar and D. Cooper, "A tuning strategy for unconstrained multi-variable model predictive control," *Industrial and Engineering Chemistry Research*, vol. 37, pp. 4003–4016, 1998.
- [196] T. K. Madhubala, M. Boopathy, J. S. Chandra, and T. K. Radhukrishnan, "Development and tuning of fuzzy controller for a conical level system," in *Proceedings of International Conference on Intelligent Sensing and Information Processing*, pp. 450–455, 2004.

-
- [197] J. H. Lilly, *Fuzzy Control with Mamdani Systems*. Ed. John Wiley & Sons, Inc., 2010.
- [198] D. K. Chaturvedi, *Soft Computing: Techniques and Its Applications in Electrical Engineering*. Ed. Springer-Verlag Berlin Heidelberg, 2008.
- [199] S. Nithya, N. Sivakumaran, T. K. Radhakrishnan, and N. Anantharaman, "Soft computing based controllers implementation for non-linear process in real time," in *Proceedings of the World Congress on Engineering and Computer Science, San Francisco, USA*, vol. 2, October 2010.
- [200] C. Priya and P. Lakshmi, "Fractional order controller design and particle swarm optimization applied to a nonlinear system," in *IEEE International Conference on Recent Trends in Information Technology (ICRTIT), Chennai, Tamil Nadu*, pp. 959–964, June 2011.
- [201] A. E. Olsson, *Particle Swarm Optimization: Theory, Techniques and Applications*. Ed. Nova Science Publishers, Incorporated, 2011.
- [202] N. Anantharaman, V. Dharmalingam, S. M. Giriraj, T. K. Radhakrishnan, and R. Sivasankar, "Particle swarm optimization technique based design of PI controller for a real-time non-linear process," *Instrumentation Science and Technology*, vol. 36, no. 5, pp. 525–542, 2008.
- [203] C. Monje, Y. Chen, B. Vinagre, D. Xue, and V. Feliu, *Fractional-order Systems and Controls*. Ed. Springer: Advances in Industrial Control, 2010.
- [204] P. Aravind and P. GirirajKumar, "Optimal tuning of PI controller using swarm intelligence for a nonlinear process," *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 2, no. 12, pp. 5949–5958, 2013.
- [205] D. W. Clarke, C. Mohtadi, and P. S. Tuffs, "Generalized predictive control - part I," *Automatica*, vol. 23, no. 2, pp. 137–148, 1987.
- [206] D. W. Clarke, C. Mohtadi, and P. S. Tuffs, "Generalized predictive control - part II," *Automatica*, vol. 23, no. 2, pp. 149–160, 1987.
- [207] B. Yegnanarayana, *Artificial Neural Networks*. Ed. Prentice-Hall of India, 2006.

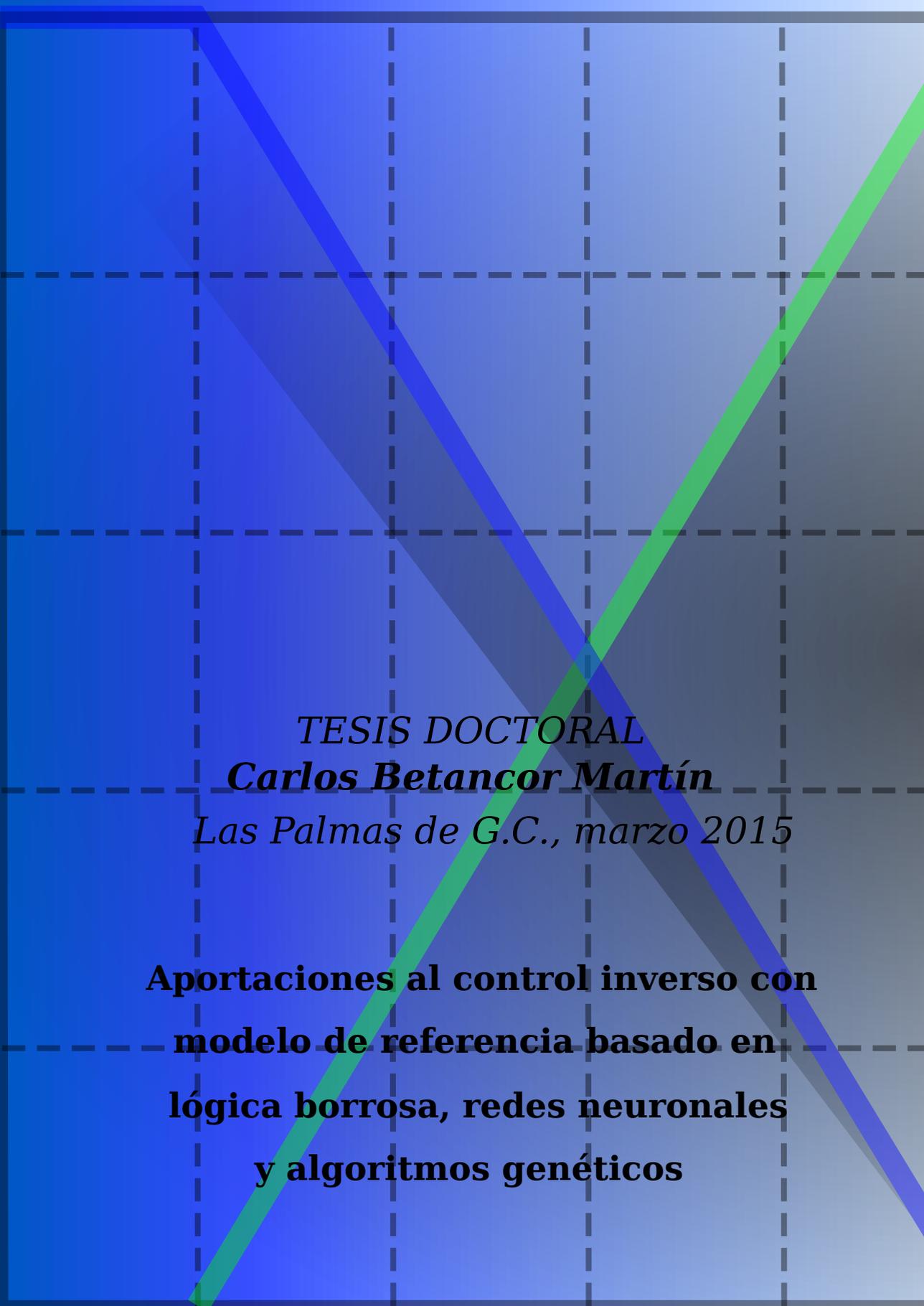
- [208] N. S. Bhuvaneshwari, G. Uma, and T. R. Rangaswamy, "Adaptive and optimal control of a non-linear process using intelligent controllers," *Applied Soft Computing*, vol. 9, p. 182–190, January 2009.
- [209] D. E. Lirk, *Optimal Control Theory: An Introduction*. Ed. Dover Books on Electrical Engineering, 2004.
- [210] A. Kumar, *Concepts of Biophysics*. APH Publishing Corporation, 2005.
- [211] J. Moré, "The Levenberg-Marquardt algorithm: Implementation and theory," *Numerical Analysis, Lecture Notes in Mathematics*, vol. 630, pp. 105–116, 1978.
- [212] M. Nørgaard, *Neural Network Based System Identification Toolbox*. Department of Automation, Department of Mathematical Modelling, Technical University of Denmark, 2000.
- [213] M. Nørgaard, *Neural Network Based Control System Design Toolkit*. Department of Automation, Department of Mathematical Modelling, Technical University of Denmark, 2000.
- [214] X. He and H. Asada, "A new method for identifying orders of input-output models for nonlinear dynamic systems," in *American Control Conference, San Francisco, CA, USA*, pp. 2520–2523, June 1993.
- [215] J. D. Fortgang, L. E. George, and W. J. Book, "Practical implementation of a dead zone inverse on a hydraulic wrist," in *ASME 2002 International Mechanical Engineering Congress and Exposition, New Orleans, Louisiana, USA*, pp. 149–155, November 2002.
- [216] J. Rubio, Z. Zamudio, J. Pacheco, and D. M. Vargas, "Proportional derivative control with inverse dead-zone for pendulum systems," *Mathematical Problems in Engineering*, vol. 2013, 2013.
- [217] R. R. Selmic and F. L. Lewis, "Deadzone compensation in motion control systems using neural networks," *IEEE Transactions on Automatic Control*, vol. 45, no. 4, pp. 602–613, 2000.
- [218] L. Liu, Y. J. Liu, and C. L. P. Chen, "Adaptive neural network control for a DC motor system with dead-zone," *Nonlinear Dynamics*, vol. 72, no. 1-2, pp. 141–147, 2013.

-
- [219] J. H. P. Cruz, "Identification and control of class of non-linear systems with non-symmetric deadzone using recurrent neural networks," *Control Theory and Applications, IET*, vol. 8, no. 3, pp. 183–192, 2014.
- [220] F. R. Gutierrez and Y. F. Makableh, "Efficient position control of DC servomotor using backpropagation neural network," in *Seventh International Conference on Natural Computation (ICNC), Shanghai*, pp. 653–657, July 2011.
- [221] Z. Wang, Y. Zhang, and H. Fang, "Neural adaptive control for a class of nonlinear systems with unknown deadzone," *Neural Computing and Applications*, vol. 17, no. 4, pp. 339–345, 2008.
- [222] J. O. Jang, "A deadzone compensator of a DC motor system using fuzzy logic control," *IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews*, vol. 31, no. 1, pp. 42–48, 2001.
- [223] W. M. Bessa, M. S. Dutra, and E. Kreuzer, "An adaptive fuzzy dead-zone compensation scheme and its application to electro-hydraulic systems," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 32, no. 1, pp. 1–7, 2010.
- [224] F. L. Lewis, W. K. Tim, L. Z. Wang, and Z. X. Li, "Deadzone compensation in motion control systems using adaptive fuzzy logic control," *Control Systems Technology, IEEE Transactions on*, vol. 7, pp. 731–742, Nov 1999.
- [225] J. O. Jang, H. T. Chung, and G. J. Jeon, "Saturation and deadzone compensation of systems using neural networks and fuzzy logic," in *Proceedings of the 2005 American Control Conference*, pp. 1715–1720, June 2005.
- [226] G. Tao and F. L. Lewis, *Adaptive Control of Nonsmooth Dynamic Systems*. Springer, 2001.
- [227] D. A. Recker, P. V. Kokotovic, D. Rhode, and J. Winkelman, "Adaptive nonlinear control of systems containing a deadzone," in *Proceedings of the 30th IEEE Conference on Decision and Control, Brighton, England*, pp. 2111–2115, December 1991.
- [228] N. J. Ahmad, H. K. Ebraheem, M. J. Alnaser, and J. M. Alostath, "Adaptive control of a DC motor with uncertain deadzone nonlinearity at the input," in *Control and Decision Conference (CCDC), Chinese*, pp. 4295–4299, May 2011.

- [229] X. S. Wang, H. Hong, and C. Y. Su, "Model reference adaptive control of continuous-time systems with an unknown input dead-zone," *IEE Proceedings on Control Theory and Applications*, vol. 150, no. 3, p. 261–266, 2003.
- [230] W. Zhonghua, Y. Bo, C. Lin, and Z. Shusheng, "Robust adaptive dead-zone compensation of DC servo system," *IEE Proceedings on Control Theory and Applications*, vol. 153, no. 6, pp. 709–713, 2006.
- [231] X. Wang, C. Su, and H. Hong, "Robust adaptive control of a class of nonlinear systems with unknown dead-zone," *Automatica*, vol. 40, no. 3, pp. 407–413, 2004.
- [232] M. L. Corradini and G. Orlando, "Robust stabilization of nonlinear uncertain plants with backlash or dead zone in the actuator," *IEEE Transactions on Control Systems Technology*, vol. 10, no. 1, pp. 158–166, 2002.
- [233] M. C. Turner, "Actuator deadzone compensation: theoretical verification of an intuitive control strategy," *Control Theory and Applications, IEE Proceedings*, vol. 153, no. 1, pp. 59–68, 2006.
- [234] E. F. Camacho and C. Bordons, *Model Predictive Control*. Springer, 2004.
- [235] D. Bao-Cang, *Modern Predictive Control*. Taylor & Francis Inc, 2009.
- [236] R. Erickson and D. Maksimović, *Fundamentals of power electronics*. Kluwer Academic Publishers, Boulder, 2004.
- [237] S. Vijayalakshmi and K. Muthukumar, "Design and simulation for sliding-mode control in DC-DC converter," in *IEEE International Conference On Advances In Engineering, Science And Management (ICAESM), Nagapattinam, Tamil Nadu*, pp. 662–665, March 2012.
- [238] M. Qin and J. Xue, "Improved pulse regulation control technique for switching DC-DC converters operating in DCM," *IEEE Transactions on Industrial Electronics*, vol. 60, pp. 1819–1830, May 2013.
- [239] T. Hashemi, A. Farnam, R. M. Esfanjani, and H. M. Kojabadi, "A new approach to design switching strategy for the buck converters," in *4th Power Electronics, Drive Systems and Technologies Conference (PEDSTC), Tehran*, pp. 301–305, February 2013.

-
- [240] X. Li, M. Chen, and Y. Tsutomu, "A method of searching PID controller's optimized coefficients for buck converter using particle swarm optimization," in *IEEE 10th International Conference on Power Electronics and Drive Systems (PEDS), Kitakyushu*, pp. 238–243, April 2013.
- [241] M. Mahdavian, M. B. Poudeh, and N. Wattanapongsakorn, "Improving the efficiency of power supplies with an adaptive control method," in *10th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTICON)*, pp. 1–4, May 2013.
- [242] M. D. Pedroso, C. B. Nascimento, A. M. Tuset, and S. Kaster, "Performance comparison between nonlinear and linear controllers with weighted adaptive control applied to a buck converter using poles placement design," in *IEEE International Symposium on Industrial Electronics (ISIE)*, pp. 1–6, May 2013.
- [243] J. Zhao, Y. Liu, K. Qu, and H. Geng, "Adaptive hysteresis band control for DC-DC buck converter," in *IEEE ECCE Asia Downunder, Melbourne, VIC*, pp. 804–809, June 2013.
- [244] P. Gayathiridevi, S. Vijayalakshmi, and K. Vairamani, "Discrete controller for high frequency buck converter," in *International Conference on Circuits, Power and Computing Technologies (ICCPCT), Nagercoil*, pp. 605–610, March 2013.
- [245] G. Martínez, A. García, F. R. Trejo, and F. Coyolt, "Design and implementation of a FLC for CD-CD converters using a microcontroller for PV systems," in *International Conference on Electronics, Communications and Computing (CONIELECOMP), Cholula*, pp. 158–161, March 2013.
- [246] W. Janke, "Averaged models of pulse-modulated DC-DC power converters. part I. discussion of standard methods," *Archives of Electrical Engineering*, vol. 61, pp. 609–631, November 2012.
- [247] W. Janke, "Averaged models of pulse-modulated DC-DC power converters. part II. models based on the separation of variables," *Archives of Electrical Engineering*, vol. 61, pp. 633–654, December 2012.
- [248] S. Laali and H. M. Mahery, "Buck DC-DC converter: Mathematical modeling and transient state analyzes," in *3rd IEEE International Sympo-*

- sium on Power Electronics for Distributed Generation Systems (PEDG)*, Aalborg, pp. 661–667, June 2012.
- [249] E. Babaei and H. M. Mahery, “Mathematical modelling and analysis of transient and steady states of buck DC-DC converter in DCM,” *COMPEL - The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, vol. 32, no. 1, pp. 337–363, 2013.
- [250] N. L. Diaz and J. J. Soriano, “Study of two control strategies based in fuzzy logic and artificial neural network compared with an optimal control strategy applied to a buck converter,” in *Annual Meeting of the North American Fuzzy Information Processing Society (NAFIPS’07)*, San Diego, CA, pp. 313–318, June 2007.



TESIS DOCTORAL
Carlos Betancor Martín
Las Palmas de G.C., marzo 2015

**Aportaciones al control inverso con
modelo de referencia basado en
lógica borrosa, redes neuronales
y algoritmos genéticos**